POLITEXT

Antoni Guasch - Miquel Àngel Piera
Josep Casanovas - Jaume Figueras

Modelado y simulación

Aplicación a procesos logísticos de fabricación y servicios

EDICIONS UPC

POLITEXT 118

Modelado y simulación

Aplicación a procesos logísticos de fabricación y servicios

POLITEXT

Antoni Guasch - Miquel Àngel Piera
Josep Casanovas - Jaume Figueras

Modelado y simulación

Aplicación a procesos logísticos de fabricación y servicios

EDICIONS UPC

La presente obra fue galardonada en el séptimo concurso
"Ajut a l'elaboració de material docent" convocado por la UPC.

Primera edición: febrero de 2002

Diseño de la cubierta: Manuel Andreu

© Los autores, 2002

© Edicions UPC, 2002
Edicions de la Universitat Politècnica de Catalunya, SL
Jordi Girona Salgado 31, 08034 Barcelona
Tel.: 934 016 883 Fax: 934 015 885
Edicions Virtuals: www.edicionsupc.es
E-mail: edicions-upc@upc.es

Producción: CPET (Centre de Publicacions del Campus Nord)
La Cup. Gran Capità s/n, 08034 Barcelona

Depósito legal: B-8987-2002
ISBN: 84-8301-577-3

Quedan rigurosamente prohibidas, sin la autorización escrita de los titulares del copyright, bajo las sanciones establecidas en las leyes, la reproducción total o parcial de esta obra por cualquier medio o procedimiento, comprendidos la reprografía y el tratamiento informático, y la distribución de ejemplares de ella mediante alquiler o préstamo públicos.

A Maria, Alba, Andreu i Guillem pel vostre amor i recolzament.
Per a vosaltres, pares, que m'heu ensenyat el camí a seguir.
A. Guasch

A Gemma, que sense entendre res, ho ha entés tot.
A Alba i Judit, les seves riatlles també estan presents en aquestes planes.
A vosaltres pares, per tot el que m'heu donat.
M.A. Piera

A Rosa, Ada i Laura, i als meus pares, Josep i Pilar.
A les persones estimades. A Jaume Barceló per la seva amistat,
la nostra passió comuna per la muntanya i per haver-me introduït a la simulació.
J.Casanovas

Vull agrair el suport incondicional de les persones estimades,
a tu Maite, pel teu amor i les hores passades al meu costat,
i a vosaltres pares, per l'esforç i sacrifici esmerçat en el vostre fill.
J. Figueras

© Los autores, 2002; © Edicions UPC, 2002.

Prólogo

Siempre es difícil hacer la introducción a un libro, máxime cuando es académico ya que... ¿qué se puede decir que no esté dicho? Y si se expone alguna idea nueva, el lector se pregunta ¿por qué no lo explican los autores? ¿No pretenderán los autores del prólogo robar protagonismo a los autores legítimos? Así que empezaremos con aspectos generales sobre el tema del libro que se disponen a leer. El mundo de la simulación de procesos productivos, logísticos o de servicios ha sufrido un importante avance desde su inicio. En la actualidad la simulación de procesos estocásticos se utiliza cada vez más en el estudio detallado de sistemas crecientemente complejos, desde factorías industriales a terminales portuarias.

Está apareciendo una gran diversidad de programas que facilita la aproximación a problemas con un grado elevado de realismo y un mayor rigor metodológico. También existe un mayor interés por no dejar nada al azar en el diseño o explotación de cualquier proceso. Se intenta que los modelos de simulación puedan integrarse o sean útiles para apoyar su funcionamiento diario. Así, se espera que existan modelos que de forma virtual coloquen contenedores, estas cajas que vienen cargadas de sorpresas del ancho mundo, en una posición óptima y adecuada dentro de una terminal portuaria, mientras el buque que los transporta todavía está surcando el océano (y no se espera su llegada a puerto hasta la mañana siguiente).

Cabe destacar el enfoque práctico que los autores han sabido darle, buscando casos reales que sin duda ellos mismos han vivido. No es este el lugar de resaltar el bagaje profesional de los autores, sólo recordar una frase de Toni: *tenim els dits pelat*s. Es decir, no sólo tienen las pestañas quemadas, como ocurre sin duda a menudo en el mundo académico, sino también las puntas de los dedos. Yo me atrevería a decir que hasta los zapatos los tienen gastados, cerrando así un círculo trinitario: estudiar, hacer, comprobar. Para estudiar y trabajar un tema es necesario, en efecto, gastar zapatos y hablar con los expertos allí donde se hallen.

Hay quien dice que para desarrollar un modelo las mejores horas son las nocturnas, cuando todo está oscuro y es posible, con la simple ayuda de un teclado y un ratón, crear un mundo ficticio que toma cuerpo entre las sombras. Se trata de un mundo artificial, mecánico, donde todo ocurre por un motivo. A medida que se hojean los diferentes capítulos del libro, en algún momento parece envolvernos el aroma del café, el mismo café que sin duda ha mantenido a los autores en vigilia durante largas horas, peleándose con el ordenador. A veces debe verse la relación con el ordenador como una lucha. Como dice siempre Miquel Àngel “hay que vencerlo” y que al final haga lo que estaba previsto. El sacrificio

© Los autores, 2002; © Edicions UPC, 2002.

nocturno de los autores, y de sus familias, no ha sido en vano. Gracias a su trabajo es ahora posible repasar tranquilamente y a la luz del día, los principios de la simulación de procesos estocásticos. Parece que ha sido sólo un sacrificio en horas, que no en malos ratos. En esta lucha con el ordenador las heridas no duelen, los distintos obstáculos sólo escuecen en el amor propio hasta que finalmente son superados. En ese momento todo encaja y el modelo de simulación toma vida propia. Es posible, en este sentido, establecer un paralelismo con la creación del doctor Frankestein, aunque en este caso nadie corre peligro.

Desde una óptica más profesional, pensando en los expertos en simulación y a todos aquellos que buscarán en ella solución a sus problemas, éste es un texto idóneo para introducirse en el extenso mundo de la simulación. Estas afirmaciones no se presentan al vuelo ni gratuitamente, sino apoyadas en la experiencia y pagadas con el sudor de la frente. Sin duda, nuestros esfuerzos en el desarrollo de modelos de simulación orientados a eventos discretos habrían caído en un pozo profundo sin el apoyo moral, académico y profesional de los autores del libro. La mayor parte del trabajo se destina a evaluar el comportamiento de los sistemas: esto es la razón de ser del modelo y significa largas horas estudiando su evolución. De aquí nace otra máxima de los autores: el desarrollo de un modelo de simulación precisa huir de la superficialidad, y requiere reflexionar, investigar, entender y adquirir un compromiso con el rigor y la responsabilidad.

Para terminar agradecer, felicitar y saludar a los autores del libro.

David Vergés
Josep Canudas
Barcelona, enero 2002

© Los autores, 2002; © Edicions UPC, 2002.

Agradecimientos

Deseamos destacar en primer lugar a la Universitat Politècnica de Catalunya (UPC) por el impulso y el apoyo recibido. El premio concedido para la elaboración de material docente ha permitido aglutinar y dar cuerpo a las energías e ilusiones de los autores que, desde hace tiempo, deseaban hacer una aportación al apasionante mundo de la simulación.

Nuestra gratitud a los profesores Jaume Barceló y Rafael M. Huber (Univ. Politècnica de Catalunya) y César de Prada (Univ. de Valladolid) por la formación y tutelaje recibidos, lo que nos permitió introducirnos y consolidar nuestra formación en esta área. Han sido muy valiosas las aportaciones y sugerencias de los profesores M. Santos (Univ. Complutense de Madrid), M. Berenguel (Univ. de Almería), A. Sanfeliu (Univ. Politècnica de Catalunya) y J.L. Villarroel (Univ. de Zaragoza). A los profesores J.J. Ramos, R. Moreno, R. Vilanova, J.M. Surís y muy especialmente al profesor I. Serra (U.A.B.) por haber sabido compartir tanto las ilusiones como los malos momentos que nos ha tocado pasar.

Queremos destacar a las personas y empresas que han confiado en la simulación y en nosotros para resolver sus problemas o avanzar en su proceso de formación. La lista no es exhaustiva, resaltamos aquellas personas que nos han forzado a alcanzar y superar el límite de nuestros conocimientos al plantearnos problemas de notable dificultad científica y práctica: E. Echeverria, J.A. Bautista, J. Dueñas y J. Gómez (Sidmed -Aceralia/Unisor-); S. Ametller (Sener); J. Moreno (Autoritat Portuaria de Barcelona); J. Benet, A. Benito, J. Pages y J. Conrado (Grupo Damm); X. Gesé (Puertos del Estado); A. Casanovas (Almirall-Prodesfarma); X. Ferran (Indra) y J. Rivera (Consorcio Hospitalario Parc Tauli).

Han sido muy valiosas las aportaciones y el apoyo de Marc Bacardit, Josep Vila, Oscar Hernández, Víctor Huerta, Susana Francisco, Jordi Montero, Pau Fonseca, Lluïsa Romeu y Rosa Anglès. También deseamos agradecer el apoyo de J.A. Muñoz de Drilco S.A. para que nuestros siempre exigentes estudiantes puedan disponer de licencias educaciones del software de simulación Arena y su interés para resolver nuestras, a veces no fáciles, dudas. Por último, esperamos que este libro satisfaga a nuestros lectores. Serán bienvenidos los comentarios o sugerencias que nos permita mejorar sus futuras versiones.

Antoni Guasch,
Miquel Angel Piera,
Josep Casanovas,
Jaume Figueras.

Barcelona, enero de 2002

© Los autores, 2002; © Edicions UPC, 2002.

Índice

PRÓLOGO ..... **III**

AGRADECIMIENTOS ..... **V**

ÍNDICE ..... **VII**

1 INTRODUCCIÓN A LAS TÉCNICAS DE SIMULACIÓN DIGITAL **1**

1.1 TIPOS DE SISTEMAS 2
1.2 TIPOS DE MODELOS 3
1.3 MODELOS DE SIMULACIÓN DE EVENTOS DISCRETOS 6
1.4 ALTERNATIVAS EN LA SIMULACIÓN DE MODELOS DE EVENTOS DISCRETOS 7
1.4.1 Simulación del modelo estático *8*
1.4.2 Simulación a mano *8*
1.4.3 Simulación digital mediante un lenguaje de propósito general *12*
1.4.4 Experimentación mediante un entorno de simulación *13*
1.5 VENTAJAS Y DESVENTAJAS DE LA SIMULACIÓN 14
1.6 CAMPOS DE APLICACIÓN DE LA SIMULACIÓN ORIENTADA A EVENTOS DISCRETOS 15
1.7 CICLO DE VIDA DE UN PROYECTO DE SIMULACIÓN 16
1.8 CRITERIOS PARA LA EVALUACIÓN DE LAS HERRAMIENTAS DE SIMULACIÓN 20
1.9 EJERCICIOS 21

2 MODELADO DE SISTEMAS ORIENTADOS A EVENTOS DISCRETOS **25**

2.1 FORMALIZACIÓN DE MODELOS CONCEPTUALES 27
2.2 MODELOS DE FLUJOS 29
2.3 REDES DE COLAS 30
2.4 REDES DE PETRI 32
2.4.1 Definiciones *33*
2.4.2 Programación Bottom-Up *39*
2.4.3 Análisis de las redes de Petri *44*
2.4.4 Algunas consideraciones sobre la formalización de modelos conceptuales en RdP *56*
2.5 LAS REDES DE PETRI COLOREADAS 57
2.5.1 Definiciones *59*
2.5.2 Ventajas del formalismo de RdPC *72*

© Los autores, 2002; © Edicions UPC, 2002.

2.6 OTRAS APROXIMACIONES FORMALES PARA EL MODELADO DE SISTEMAS DE EVENTOS DISCRETOS 72
2.7 EJERCICIOS 74

3 MODELOS ESTADÍSTICOS EN SIMULACIÓN 87

3.1 DEFINICIONES 89
3.2 DESCRIPCIÓN DE UN SISTEMA CON CARACTERÍSTICAS ESTOCÁSTICAS 91
3.3 ADQUISICIÓN Y ANÁLISIS DE DATOS 91
3.3.1 Técnicas para evaluar la independencia de los valores de una muestra 94
3.3.2 Técnicas para evaluar la estabilidad de la distribución 96
3.4 DISTRIBUCIÓN DE PROBABILIDAD (HISTOGRAMA Y DIAGRAMA DE BARRAS) 98
3.5 FUNCIÓN DE DENSIDAD DE PROBABILIDAD 100
3.6 FUNCIÓN DE DISTRIBUCIÓN ACUMULATIVA 102
3.7 DISTRIBUCIONES EMPÍRICAS PARA VARIABLES ALEATORIAS CONTINUAS 104
3.8 ANÁLISIS DEL AJUSTE DE UNA DISTRIBUCIÓN 105
3.8.1 Test de hipótesis 106
3.8.2 Técnicas heurísticas para determinar la calidad del ajuste 112
3.8.3 Qué hacer si no hay datos disponibles 113
3.9 GENERACIÓN DE DATOS ALEATORIOS A PARTIR DE UN MODELO ESTADÍSTICO 114
3.9.1 Requerimientos y utilización en simulación 115
3.9.2 Métodos históricos 117
3.9.3 Métodos actuales 117
3.9.4 Tests sobre la calidad en la generación de números aleatorios 121
3.9.5 Métodos de generación de variables aleatorias: enumeración y principios básicos .. 123
3.10 FUNCIONES DE DISTRIBUCIÓN MÁS UTILIZADAS 125
3.10.1 Funciones de distribución continuas 125
3.10.2 Funciones de distribución discretas 134
3.11 EJERCICIOS 139

4 SIMULACIÓN DE SISTEMAS ORIENTADOS A EVENTOS DISCRETOS 143

4.1 ELEMENTOS DE UN MODELO DE SIMULACIÓN 145
4.1.1 Entidades, entidades temporales y recursos 145
4.1.2 Atributos 146
4.1.3 Actividades 146
4.1.4 Eventos 147
4.1.5 Colas 148
4.2 SIMULACIÓN MANUAL Y MEDIDAS DE COMPORTAMIENTO 150
4.2.1 Retardo promedio en la cola 154
4.2.2 Tamaño promedio de la cola 154
4.2.3 Grado de utilización de los recursos 156
4.2.4 Otras estadísticas de interés 157
4.3 ELEMENTOS DE UN SIMULADOR 158
4.4 ESTRATEGIAS DE SIMULACIÓN 160
4.4.1 Programación de eventos (Event Scheduling) 160

© Los autores, 2002; © Edicions UPC, 2002.

4.4.2 Interacción de procesos (Process Interaction) ... *165*
4.4.3 Exploración de actividades (Activity Scanning) ... *168*
4.5 *SOFTWARE* DE SIMULACIÓN ... 176
4.5.1 Criterios para la evaluación de las herramientas informáticas ... *177*
4.5.2 Entornos de simulación de eventos discretos ... *179*
4.6 EJERCICIOS ... 180

5 DESARROLLO DE EXPERIMENTOS DE SIMULACIÓN ... 185

5.1 VERIFICACIÓN, VALIDACIÓN Y ACREDITACIÓN EN SIMULACIÓN ... 191
5.1.1 Principios para la validación, verificación y acreditación en simulación ... *194*
5.1.2 Técnicas de VV&T ... *197*
5.1.3 Validación ... *199*
5.2 DISEÑO DE EXPERIMENTOS ... 208
5.2.1 Introducción ... *208*
5.2.2 Diseños clásicos: simples, factoriales completos y factoriales fraccionales ... *220*
5.2.3 Criba de factores ... *224*
5.3 ANÁLISIS DE RESULTADOS ... 225
5.3.1 Simulaciones terminales ... *226*
5.3.2 Simulación en el estado estacionario ... *228*
5.3.3 Longitud de las ejecuciones de la simulación ... *228*
5.3.4 Determinación de los intervalos de confianza ... *233*
5.3.5 Técnicas para la reducción de la variancia ... *239*
5.3.6 Análisis de sensibilidad ... *241*
5.3.7 Optimización en simulación ... *242*
5.3.8 Análisis de resultados multivariantes ... *244*
5.4 EJERCICIOS ... 247

6 APLICACIONES ... 253

6.1 FABRICACIÓN Y SISTEMAS DE MANIPULACIÓN DE MATERIALES ... 254
6.1.1 Medidas de comportamiento ... *254*
6.1.2 Componentes de un proceso productivo ... *255*
6.1.3 Fuentes de azar ... *255*
6.1.4 Sistemas de manipulación de materiales ... *257*
6.2 TRANSPORTE ... 258
6.2.1 Puertos ... *259*
6.2.2 Ferrocarriles ... *262*
6.2.3 Gestión de aeropuertos ... *263*
6.3 REDES DE DISTRIBUCIÓN (*SUPPLY CHAIN*) ... 267
6.3.1 Componentes de una red de distribución ... *267*
6.3.2 Tipos de modelos ... *268*
6.3.3 Modelos de simulación ... *268*
6.4 OTRAS APLICACIONES ... 269
6.4.1 Sanidad ... *269*
6.4.2 Planes de emergencia ... *270*

© Los autores, 2002; © Edicions UPC, 2002.

6.4.3 Telecomunicaciones *270*
6.4.4 Procesos administrativos *270*
6.4.5 Actividades logísticas diversas *271*
6.5 EJEMPLOS 271
6.5.1 Drive-Through simulation *271*
6.5.2 Sistema de servicios a clientes *274*
6.5.3 Modelado de la asignación de barcos a atraques *283*
6.5.4 Célula flexible de fabricación *288*
6.6 EJERCICIOS 301

A INTRODUCCIÓN A LA TEORÍA DE COLAS 315

A.1 NOTACIÓN DE KENDALL 315
A.2 MEDIDAS DE COMPORTAMIENTO 316

B MODELADO DE SISTEMAS CON ARENA 319

B.1 MODELO DE UN RECURSO LIMITADO 319
B.2 PROCESO ROBOTIZADO SIMPLE 323
B.3 LÍNEA DE PRODUCCIÓN 325
B.4 DOS MÁQUINAS Y DOS CINTAS 327
B.5 CONTROL DE VEHÍCULOS 329
B.6 SISTEMA DE VEHÍCULOS AUTOGUIADOS (AGV) 333
B.7 METODOLOGÍA DE PROGRAMACIÓN 336

C TABLAS 345

C.1 FUNCIÓN DE DISTRIBUCIÓN NORMAL ESTÁNDAR 345
C.2 PUNTOS CRÍTICOS DE LA DISTRIBUCIÓN DE CHI-CUADRADO 346
C.3 PUNTOS CRÍTICOS DE LA DISTRIBUCIÓN T DE STUDENT 347

ÍNDICE DE CONTENIDOS 349

ACRÓNIMOS 353

BIBLIOGRAFÍA 355

© Los autores, 2002; © Edicions UPC, 2002.

Índice de Contenidos

A

Abrazo Mortal ..... *Véase* Bloqueo
Acreditación 191
Actividad 7, 146, 149
Activity Scanning ..... *Véase* Exploración de Actividades
Aeropuertos 263
Agrupación 35
AGV 258
Alcanzabilidad 45
Análisis 19
Análisis de Resultados 225
Análisis Multivariantes 244
Árbol de Alcance 44
  construcción 45
  ejemplo 45, 47, 49
Árbol de Cobertura 50
  ejemplo 51, 54, 63
Arena 14
Atributo 146

B

Basic 27
Bloqueo 52
  ejemplo 54

C

C 27
C++ 12
Célula de Fabricación Flexible 288
Cintas Transportadoras 258
Cola 148, 149
  evolución 155
  retardo promedio 154
  tamaño promedio 154
Comparación de Alternativas 236
Concurrencia 34
Conflictos 34
Construcción 16
  árbol de alcance 45
  modelo 19
Criba de Factores 224
CSL 27

D

Datos
  recogida de, ... 19
  tratamiento 19
  validez 201
Deadlock ..... *Véase* Bloqueo
Diagrama de Flujo 9
Diagrama de Simulación Manual 10
Diseño de Experimentos 208
Diseño Factorial 220
  completo 221
  fraccional 224
  simple 220
Documentación 20, 205

E

Ejecución Secuencial 34
Elementos de un Simulador 158
Emergencias 16
Entidad 2, 7, 145, 149
  permanente 7
  temporal 7

© Los autores, 2002; © Edicions UPC, 2002.

Entorno de Simulación 13
Entornos
  aeroportuarios 263
  ferroviarios 262
  peligrosos 270
  portuarios 259, 283
  productivos 288
  sanitarios 269
  servicios 271, 274
Entornos de Simulación 179
Especificación 18
Estado Estacionario 228
Estrategias de un Simulador 160
Event Scheduling ..... *Véase* Programación de Eventos
Evento 3, 7, 147, 149
  clasificación 147
Evolución de una Cola 155
Experimentos
  diseño 208
Exploración de Actividades 168
  ejemplo 171

F

Fabricación 16, 254, 288
Factor
  criba 224
Ferrocarriles 262
FIFO 35
Flujos
  modelo de, ... 29–30
Formalismo 27
  modelo de flujos 30
  red de petri 56, 32–57
  red de petri coloreada 57–72
  redes de colas 32
Fortran 27

G

GPSS 12, 27

H

Herramienta
  analítica 15
Herramientas 20, 26
  metodológicas 26

I

Implementación 12, 20
  lenguaje de propósito general 12
Interacción de Procesos 165
  ejemplo 166
Intervalos de Confianza 233
  media 234

K

Kendall 31

L

LeanSIM 14
Logística 16, 271
Longitud de Ejecución 228

M

Manipulación 254
Medidas de Comportamiento 150
Metamodelos 242
Método del Calendario 256
Método del Tiempo de Trabajo 256
Método Regenerativo 232
Metodología 27
Modelo 4
  conceptual 18, 26
  construcción de, ... 19
  continuo 6
  determinista 5
  discreto 6
  estático 8
  estocástico 5
  eventos discretos 6
  flujos 29–30
  formalismo 27
  formalización 27
  metodología 27
  otras aproximaciones 72
  redes de colas 30–32
Modularidad 28

© Los autores, 2002; © Edicions UPC, 2002.

N

Negocios 16
Nodo Transición 33
  activación 34
Notación de Colas 31
Notación de Kendall 31
Números Aleatorios 240

O

Optimización 242

P

Pascal 12
Planes de Emergencia 270
Proceso Productivo 255
Procesos Administrativos 270
Process Interaction ..... *Véase* Interacción de Procesos
Programación *Bottom-Up* 39, 56
  ejemplo 39
Programación de Eventos 160
  ejemplo 162
Programación *Top-Down* 56
Proyecto 17
  ciclo de vida 16–20
Prueba de Media Nula 235
Puertos 259, 283

R

Reachability ..... *Véase* Alcanzabilidad
Recurso 145
Recursos 7
Red de Petri 32–57
  acotada 48
  activación de transiciones 34
  análisis 44
  árbol de alcance 44
  árbol de cobertura 50
  bloqueo 52
  comportamiento 34
  definición 33
  ejemplo 35, 37, 38, 39, 45, 47, 49, 51, 52, 54
  no acotada 52
  transición 33
Red de Petri Coloreada 57–72
  arco, expresiones de 60
  definición 59
  ejemplo 57, 61, 63, 66
  guardas 61
  inicialización, expresiones de 60
  nodo lugar 61
  ventajas y desventajas 72
Redes de Colas 30–32
Redes de Distribución 267
Reducción de la Variancia 239
Rendimiento
  medidas 217
Repeticiones 229
Replicaciones ..... *Véase* Repeticiones

S

Sanidad 16, 269
Sensibilidad 241
Servicios 16, 271, 274
Siman 12
Simscript 12, 27
Simulación 1
  alternativas 7
  campos de aplicación 15–16
  elementos 158
  entorno de, ... 21
  entornos 179
  estado estacionario 228
  estática 8
  estrategias 160
  exploración de actividades ..... *Véase* Interacción de Procesos 165
  manual 10–12
  medidas de comportamiento 150
  programación de eventos 160
  proyectos de 16–20
  software 176
  terminal 226
  ventajas y desventajas 14–15
Simulador 20
  elementos 158
  estrategias 160
  exploración de actividades ..... *Véase*

© Los autores, 2002; © Edicions UPC, 2002.

Interacción de Procesos 165
programación de eventos 160
Sincronismo 56
Sincronización 34
Sistema 2
contiuo 2
discreto 2
estado 2, 33
eventos discretos 3, 25
mixto 3
Software
criterios evaluación 177
eficiencia 179
facilidad 178
fiabilidad 179
funcionalidad 177
Supply Chain .....*Véase* Redes de Distribución

T

Tabla de Simulación Manual 11
Telecomunicaciones 270
Transporte 16

V

Validación 19, 191, 199
proceso 193, 200
Variables Antitéticas 240
Variables de Control 241
Vehículos de Transporte 258
Verificación 19, 191
proceso 193
VV&T
costes 206
técnicas 197

W

Witness 14

© Los autores, 2002; © Edicions UPC, 2002.

Acrónimos

AdR: Análisis de resultados

AdS: Análisis de Sensibilidad

AR: Análisis de Regresión

ANOVA: Análisis de la variancia

BS: Bifurcación Secuencial

CRN: *Common random numbers* (números aleatorios comunes)

DdE: Diseño de Experimentos

DoD: Departamento de Defensa de USA

GNA: Generador de Números Aleatórios

GVA: Generador de Variables Aleatórias

IC: Intervalo de Confianza

IEEE: *Institute of Electrical and Electronic Engineering*

MSR: Métodos de Superficie de Respuesta

M&S: Modelado y Simulación

RdP: Red de Petri

RdPC: Red de Petri Coloreada

RV: Realidad Virtual

SCS: *Society for Computer Simulation*

TRV's: Técnicas de Reducción de la Variancia

V&V: Verificación y Validación

© Los autores, 2002; © Edicions UPC, 2002.

VV&A: Verificación, Validación y Acreditación

V&VI: Verificación y Validación Independiente

VV&T: Verificación, Validación y Testeo o pruebas

WSC: Winter Simulation Conference, organizada anualmente por la SCS

© Los autores, 2002; © Edicions UPC, 2002.

Bibliografía

[ALEX00] ALEXOPOULOS, C. y SEILA, A.F. "Output analysis for simulations". *Proceedings of the Winter Simulation Conference*, SCS, 2000.

[ANDR98] ANDRADOTTIR, S. "A review of simulation optimization techniques". *Proceedings of the Winter Simulation Conference*, SCS, 1998.

[ARTH00] ARTHUR,J.D. y NANCE, R.E. "Verification and validation without independence: a recipe for failure". *Proceedings of the Winter Simulation Conference*, SCS, 2000.

[BABI86] BABICH, W.A. *Software Configuration Management*. Addison-Wesley, Massachusetts. 1986.

[BALC84] BALCI, O.,SARGENT, R. "A bibliography on the credibility assesment and validation of simulation and mathematical models". *Simuletter*, 15,3. 1984.

[BALC81] BALCI, O.,SARGENT, R. "A methodology for risk analysis in the statistical validation of simulation models". *Communications of ACM*, 24. 1981.

[BALC00a] BALCI, O., ORSMBY, W.F. "Well-defined intended uses: an explicit requirement for accreditation of modeling and simulation applications". *Proceedings of the Winter Simulation Conference*, SCS, 2000

[BALC00b] BALCI, O., ORSMBY, W.F., CARR III, J.T., SAADI, S.D. "Planning for verification, validation and accreditation of modeling and simulation applications". Proceedings of the Winter Simulation Conference, SCS, 2000.

[BANK99] BANKS, J. "What does industry need from simulation vendors un Y2K and after? A panel discussion". *Proceedings of the 1999 Winter Simulation Conference*, 1999.

[BANK96] BANKS, J., CARSON, J.S. y NELSON, B.L. *Discrete-Event System Simulation*. Prentice-Hall, 1996.

[BARC96] BARCELO, J. *Simulación de sistemas siscretos.* ISDEFE, 1996.

[BARC00] BARCELO, J. "Putting the rush back in rush hour". *OR/MS Today*. April 2000.

[BARN00] BARNARD, C.J., HAAFTEN, D.H.V. "Simulation model for shipment of waste to the waste isolation pilot plant". *Proceedings of the 2000 Winter Simulation Conference*, 2000.

© Los autores, 2002; © Edicions UPC, 2002.

[BAUS93] BAUSE, F. “QN + PN = QPN Combining Queueing Networks and Petri Nets”. *Forschungsbericht* n. 461, des Fachbereichs Informatik der Universität Dortmund (Germany), 1993.

[BETT97] BETTONVIL, B. y KLEIJNEN, J.P.C. “Searching for important factors in simulation models with many factors: sequential bifurcation”. *EJOR*, vol. 96, 1997.

[BILE96] BILES, W.E. “Discrete-Event Simulation”. *System Modeling and Computer Simulation*, Kheir N.A., Dekker, 1996.

[BOWM01] BOWMAN, R.A. y SCHMEE, J. “Pricing and managing a maintenance contract for a fleet of aircrat engines”. Simulation, vol. 76, num. 2, 2001.

[BRAD00] BRADE, D. “Enhancing modeling and simulation accreditation by structuring verification and validation results”. *Proceedings of the Winter Simulation Conference*, SCS,2000

[BRAT87] BRATLEY, P., FOX, B., SCHRAGE, L. *A guide to simulation*. Springer-Verlag, 1987.

[BRUZ99] BRUZZONE, A., GIRIBONE, P., REVETRIA, R. “Operative requirements and advances for the new generation simulators in multimodal container terminals”. *Proceedings of the Winter Simulation Conference*, SCS,1999.

[CARR92] CARRIE, A. *Simulation of Manufacturing Systems*. Wiley, 1992.

[CARS86] CARSON, J.S. “Convincing Users of Model’s Validity is a Challenging Aspect of Modeler’s Job”. *Industrial Engineering*, junio 1986.

[CASA99] CASANOVAS, J., PÉREZ, W., FONSECA, P., MONTERO, J. “Simulation of reception, expedition and picking areas of a pharmaceutical products plan”. *Proceedings ETFA’99*. IEEE, 1999.

[CASS99] CASSANDRAS, C.G. y LAFORTUNE, S. *Introduction to Discrete Event Systems*, Kluwer Academic Publishers, 1999.

[CENT98] CENTENO, M., REYES, M.F. “So you have your model: what to do next. A tutorial on simulation output analysis”. *Proceedings of the Winter Simulation Conference*, SCS, 1998.

[CENT00] CENTENO, M., ALBACETE, C., TERZANO, D.O., CARRILLO, M. y OGAZON, T. “A simulation study of the radiology department at JMH”. *Proceedings of the 2000 Winter Simulation Conference*, SCS, 2000.

[CHAR93] CHARNES, J.M., KELTON, W.D. “Multivariate autoregressive techniques for constructing confidence regions of the mean vector”.Mgt. Science, 39,1993.

[CHAR95] CHARNES, J.M. “Analyzing multivariate output”. *Proceedings of the Winter Simulation Conference*, SCS, 1995.

[CHWI00] CHWIF, L., RIBEIRO, M. y PAUL, R.J. “On Simulation Model Complexity”. *Proceedings of the 2000 Winter Simulation Conference*, 2000.

[COLE93] COLEMAN, D.E., MONTGOMERY, D.C. “A systematic approach to planning for a designed industrial experiment”. *Technometrics*, vol. 35-1, 1993.

[CRAN74] CRANE, M.A., IGLEHART, D.L. “Simulating stable stochastic systems, I y II. *Journal of ACM*, vol. 21, 1974.

© Los autores, 2002; © Edicions UPC, 2002.

[CRAN75] CRANE, M.A., IGLEHART, D.L. “Simulating stable stochastic systems, III: regenerative methods and discrete-event simulations”. *Operations Research*, vol. 23, 1975.

[DALA01] DALAL, M.A. “Simulation modeling at Union Pacific Railroad”. *Proceedings of the 2001 Winter Simulation Conference*, 2001.

[DIJK71] DIJKSTRA, E.W. “Hierarchical Ordering of Sequential Processes”. *Acta Informatica* 1, pp. 115-138, 1971.

[DONO95] DONOHUE, J.M. “The use of variance reduction techniques in the estimation of simulation metamodels”. *Proceedings of the Winter Simulation Conference*. SCS, 1995.

[DOUG97] DOUG, B. “A different shade of blue: how IBM transformed itself using the powerof networked information”. *Industry Week*, julio 1997.

[DUBI00] KELTON, W.S., SADOWSKI, R.P. y SADOWSKI D.A. *Monte Carlo Applications in Systems Engineering*, John Wiley & Sons, 2000.

[FORG83] FORGIONNE, G.A. “Corporate Management Science Activities”. *Interfaces*, vol. 13, num. 3, pp. 20-23, 1983.

[FRIE88] FRIEDMAN, L.W., PRESSMAN, I. “The metamodel in simulation analysis: can it be trusted?”. *Journal of the Oper. Res. Soc*. 39, 1988.

[FRIE96] FRIEDMAN, L.W. “The simulation metamodel”. *Kluwer Academic Publishers*, 1996.

[HARP89] HARPELL, J.L., LANE, M.S. y MANSOUR, A.H. “Operations Research in Practice: a Longitudinal Study”. *Interfaces*, vol. 19, num. 3, pp. 65-74, 1989.

[HARR91] HARRINGTON, H.J. *Business Process Improvement*. McGraw-Hill, 1991.

[HELL98] HELLEKALEK, P. “Good random number generators are (not so) easy to find”. *Mathematics and Computers in Simulation*, 46, 1998.

[HILL91] HILLIER, F.S. y LIEBERMAN, G.J. *Introducción a la investigación de operaciones*, McGraw-Hill, 1991.

[HUTC01] HUTCHISON,. D.W., HILL, S.D. “Simulation optimization of airline delay with constraints”. *Proceedings of the 2001 Winter Simulation Conference*, 2001.

[INGL99] INGALLS, R.G. “CSCAT: The COMPAQ supply chain analysis tool”. *Proceedings of the 1999 Winter Simulation Conference*, 1999.

[IOAN99] IOANNOU, P.G. y MARTINEZ, J.C. “Who servers whom? Dynamic resource matching in an activity-scanning simulation system”. *Proceedings of the 1999 Winter Simulation Conference*, 1999.

[JAIN01] JAIN, R. “The art of computer systems performance analysis” *Wiley*, 1991

[JAIN01] JAIN, S., ERVIN, E.C., LATHROP, A.P., WORKMAN, R.W. y COLLINS, L.M. “Analyzing the supply chain for a large logistics operation using simulation”. *Proceedings of the 2000 Winter Simulation Conference*, 2000.

[JENG97] JENG, M.D. “A Petri Net Synthesis Theory for Modeling Flexible Manufacturing Systems”. *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 27, n. 2, abril 1997.

© Los autores, 2002; © Edicions UPC, 2002.

[JENS97] JENSEN, K. *Coloured Petri Nets.* vol. 1, Springer-Verlag, 1997.

[JONG97] JONG, S.J., LEVARY, R.R. y FERRIS, M.E. "Planning preventive maintenance for a fleet of police vehicles". Simulation, vol. 68, num. 2, 1997.

[JOUS01] JOUSTRA, P.E. "Simulation of check-in at airports". *Proceedings of the Winter Simulation Conference*, SCS,2001.

[KELT95] KELTON, W.D. "A tutorial on design and analysis of simulation experiments". *Procedings of the Winter Simulation Conference*. SCS, 1995.

[KELT98] KELTON, W.S., SADOWSKI, R.P. y SADOWSKI, D.A. *Simulation With Arena*. McGraw-Hill, 1998.

[KEND53] KENDALL, G. G. "Stochastic Processes Occurring in the Theory of Queues and Their Analysis by the Method of Imbedded Markov Chains". *Annals of Mathematical Statistics* 24, pp. 338-354, 1953.

[KLEI92] KLEIJNEN J.P.C. y ANNINK, B. "Pseudorandom number generators for supercomputers and classical computers: A practical introduction". European Journal of Operations Research 63. 1992.

[KLEI92] KLEIJNEN, J.P.C. *Simulation: a statistical perspective*. John Wiley, 1992.

[KLEI95a] KLEIJNEN, J.P.C. "Case study: statistical validation of simulation models". *EJOR*, 87-1, 1995.

[KLEI95b] KLEIJNEN, J.P.C. "Sensivity analysis an optimization in simulation: design of experiments and case studies" *Procedings of the Winter Simulation Conference*. SCS. 1995.

[KLEIN98] KLEINDORFER, G.B., O'NEILL, L., GANESHAN, R. "Validation in Simulation: Various Positions in the Philosophy of Science". *Management Science*, vol.44. n.8, 1998.

[KNUT97] KNUTH, D.E. *The art of computing programming. Vol.2: Seminumerical algorithms*. Addison-Wesley. (3ª. Ed.) 1997.

[KREU86] KREUTZER, W. *System Simulation: Programming Styles and Languages,* Addison-Wesley, 1986.

[KUEI94] KUEI, C., MADU, C.N. "Polynomial metamodeling and Taguchi design in simulation with application to the maintenance float system", *EJOR*, 72, 1994.

[KWON94] KWON,C., TEW, J.D. "Strategies for combining Antithetic Variates and Control Variates in Designed Simulation Experiments". *Management Science*, vol. 40, n.8, 1994.

[LANE93] LANE, M.S. MANSOUR, A.H. y HARPELL, J.L., "Operations Research Techniques". *Interfaces*, vol. 23, num. 2, pp. 63-68, 1993.

[LAW99] LAW, A.M., KELTON, W.D. *Simulation Modeling and Analysis*, McGraw-Hill, 2000.

[LECU94] L'ECUYER, P. "Uniform random number generation". *Annals of Operations Research*, 53. 1994.

[LEBE98] LEBEDEV, A. y STAPLEST, P. "Simulation of materials handling systems in the mines: two case studies". Simulation, vol. 70, num. 3, 1998.

© Los autores, 2002; © Edicions UPC, 2002.

[LEHM51] LEHMER, D.H. “Mathematical methods in large-scale mathematical units”. Symp. on Large-Scale Digital Cal. Mach. Hardvard Univ. Press. 1951.

[LEWI89] LEWIS, P.A.W. y ORAV, E.J. *Simulation methodology forstaticians, operations analysts and engineers*. Pacific Grove Wandsworth and Brooks /Cole, 1989

[LITT61] LITTLE, J.D.C. “A Proof of the Queueing Formula L=λ W”. *Operations Research*, 9, pp. 383-387, 1961.

[LIU01] LIU, J.S. *Monte Carlo Strategies in Scientific Computing*, Springer-Verlag, 2001.

[MADU92] MADU, C., KUEI, C-H. “Group screening and Taguchi design in the optimization of multi-echelon maintenance float simulation metamodels”. *Computers and operations research*, vol. 19-2.1992.

[MARK81] MARKOWITZ, H.M. “Barriers to the practical use of simulation analysis”. *Proceedings of the 1981 Winter Simulation Conference*, 1981.

[MARS68] MARSAGLIA, G. “Randon numbers fall mainly in the planes”. *Proc. Natl. Acad. Sci. US*, 1968

[MARS91] MARSAGLIA G., ZAMAN A. “A new class of random number generators”. *The Annals of Applied Probability*. Vol.1, n.3. 1991

[MATS94] MATSUNOMU,M. y KURITA,Y. “Twisted GFSR generators II”. *ACM Trans. Modeling and Computer Simulation*, 4, 1994.

[MCGR00] McGRAW, R.M., y McDONALD; R.M. “Abstract Modeling for Engineering and Engagement Level Simulations”. *Proceedings of the 2000 Winter Simulation Conference*, 2000.

[MONT91] MONTGOMERY, D.C. *Design and Analysis of Experiments*. Wiley & Sons. 1991.

[MIDD01] MIDDELKOP, D. y BOUWMAN, M. “Simone: large scale network simulation”. *Proceedings of the 2001 Winter Simulation Conference*, 2001.

[NANC99] NANCE, R.E. “Redundancy in Model Specification for Discrete Event Simulation”. *ACM Transactions on Modeling and Computer Simulation*, vol. 9, n. 3, 1999.

[NELS93] NELSON, B.L.,HSU, J.C. “Control-variate models of Common Random Numbers for Multiple Comparisons with the Best”. *Management Science*, vol. 39, n. 8, 1993.

[NIED92] NIEDERREITER, H. “Random Number Generation and Quasi-Monte Carlo Methods”. *SIAM*, 1992.

[NIKO98] NOKOUKARAN, J. HLUPIC, V. y PAUL, R.J. “Criteria for simulation software evaluation”. *Proceedings of the 1998 Winter Simulation Conference*, 1998.

[ORMS99] OR/MS Today. “Simulation Survey”, febrero 1999.

[ORTN99] ORTNER, W. y STARY, C. “Virtualization of Organizations: Consequences for Workflow Modeling”. *Proceedings of the 32nd Hawaii International Conference on System Sciences*, 1999.

[OVER82] OVERSTREET,C.M. *Model Specification and Analysis for Discrete Event Simulation*. Ph. D. Dissertation. Department of Computer Science, Virginia Tech, Blacksburg. VA. 1982.

© Los autores, 2002; © Edicions UPC, 2002.

[ÖZD96] ÖZDEMIREL, N.E., GAZANTER, Y., GÜLSER, K. "Computer-aided planning and design of manufacturing simulation experiments". *Simulation*, SCS, sep. 1996.

[PARK88] PARK S.K., MILLER K.W. "Random Number Generators: good ones are hard to find". *Communications of ACM*, vol.31. núm.10, 1988.

[PARS01] PARSONS, D.J. y SIPRELLE, A.J. "A supply chain case study of a food manufacturing merger". *Proceedings of the 2001 Winter Simulation Conference*, 2001.

[PAUL93] PAUL, R.J. "Activity Cycle Diagrams and the Three Phase Approach". *Proceedings of the 1993 Winter Simulation Conference*, 1993.

[PEDG95] PEDGEN C.D., SHANNON, R.E. y SADOWSKI, R.P. *Introduction to Simulation Using SIMAN*. McGraw-Hill, 1995.

[PETRI66] PETRI, C.A. *Kommunikation mit Automaten*, Bonn: Institut für Instrumentelle Mathematik, Schriften des IIM Nr. 2, 1962, Second Edition:, New York: Griffiss Air Force Base, Technical Report RADC-TR-65-377, vol. 1, 1966

[PIDD98] PIDD, M. Y CASTRO, R.B. "Hierarchical Modular Modelling in Discrete Simulation". *Proceedings of the 1998 Winter Simulation Conference*, 1998.

[PIER02] PIERA, M.A., GUASCH, A. y CASANOVAS, J. *Simulación de sistemas orientados a eventos discretos*. Monografías de CEA-IFAC, 2002.

[POOC00] POOCH, U.W. y WALL, J. *Discrete Event Simulation: A practical approach*. CRC Press, 2000.

[PRAT97] PRAT, A., TORT-MARTORELL, X., GRIMA, P. y POZUETA,L. *Métodos estadísticos.. Control y mejora de la calidad*. Edicions UPC, 1997.

[QMIP93] QMIPS GROUP in UTO+ZAR. "The Timed (Coloured) Petri Net Formalism: Position Paper". Formalisms: Principles and State-of-the-Art, *Proceedings of the Erlangen Workshop*, 1993.

[RAAT93] RAATIKAINEN, K.E. "A sequential procedure for simultaneous estimation of several means".*ACM Transactions on Modeling and Computer Simulation*, 3, 1993.

[RAMI01] RAMIS, F.J., PALMA, J.L. y BAESLER, F.F. "The use of simulation for process improvement at an ambulatory surgery center". *Proceedings of the 2001 Winter Simulation Conference*, SCS, 2001.

[ROHR98] ROHRER, M.W. "Simulation of Manufacturing and Material Handling Systems". *Handbook of Simulation*. Ed. Banks, J. Wiley Interscience, 1998.

[RIOS97] RIOS, D., RIOS, S. y MARTIN, J. *Simulación. Métodos y aplicaciones*. Ra-Ma, 1997.

[RIPL87] RIPLEY, B.D. *Stochastic Simulation*. J. Wiley, 1997.

[ROBI95] ROBINSON, S. "A heuristic technique for selecting the run-length of non-terminating steady-state simulations". *Simulation* 65:3, 1995.

[ROSS99] ROSETTI, M.D., TRZCINSKI, G.F. y SYVERUD, S.A. "Emergency department simulation and determination of optimal attending physician staffing schedules". *Proceedings of the 1999 Winter Simulation Conference*, SCS, 1999.

[RUBI81] RUBINSTEIN, R.Y. *Simulation and the Monte Carlo Method.* Wiley, 1981

© Los autores, 2002; © Edicions UPC, 2002.

[RUNC99] RUNCIMAN, N., VAGENAS, N. y BAIDEN, T. “Evaluation of underground development mining systems using discrete-event simulation”. Simulation, vol. 72, num. 1, 1999.

[SANC99] SANCHEZ, S.M. “ABC’s of Output Analysis”. Proceedings of the Winter Simulation Conference, SCS, 1999.

[SARG99] SARGENT, R.G. “Validation and Verification of Simulation Models” *Proc. of 1999 Winter Simulation Conference*,SCS, 1999.

[SHAP01] SHAPIRO, J.F. *Modeling the Supply Chain*, Duxbury Press, 2001.

[SCHR96] SCHRIBER, T.J. “Comparison 8: Canal-and-lock-system”, *EUROSIM – Simulation News Europe*, 16, pp. 29-43. ASIM, Viena (Austria), 1996.

[SCHR96] SCHRIBER, T.J. “Comparison 8: Canal-and-lock-system”, *EUROSIM – Simulation News Europe*, 16, pp. 29-43. ASIM, Vienna (Austria), 1996.

[SCHR00] SCHRIBER, T.J. y BRUNNER, D.T. “Inside discrete-event simulation software: how it works and why it matters”. *Proceedings of the 2000 Winter Simulation Conference*, 2000.

[SCHR83] SCHRUBEN, L. “Simulation Modeling with Event Graphs”. *Communications of the ACM*, 26 (11), pp. 957-963, Noviembre 1983.

[SILV89] SILVA, M., VALETTE R. *Advances in Petri Nets*, Springer-Verlag, pp. 374-417, 1989.

[SIST98] SISTI, A. y FARR, S. “Model Abstraction Techniques: An Intuitive Overview”. *Proceedings of the 1998 Winter Simulation Conference*, 1998.

[SMIR48] SMIRNOV, N. “Table for estimating the goodness of fit of empirical distributions”. *Annuals of Mathematical Statistics*, vol. 19, 1948.

[STUR26] STURGES, H. “The choice of a class-interval”. *Journal of American Statistics*, vol. 21, pp. 65-66, 1926.

[TARA00] TARANTO S.A., HARPER, A.M., EDWARDS, E.B. “Developing a National allocation model for cadaveric kidneys”. *Proceedings of the 2000 Winter Simulation Conference*, 2000.

[THOM79] THOMAS, G. y DaCOSTA, J. “A Sample Survey of Corporate Operations Research”. *Interfaces*, vol. 9, num. 4, pp. 102-111, 1979.ç

[THOM95] THOMPSON P.A., MARCHANT, E.W. “Computer Model for the Evaluation of Large Building Populations”. Fire safety Journal v 24 n 2, 1995.

[TRON00] TRONE, J., GUERIN, A. y CLAY, A.D. “Simulation of waste processing, transportation and disposal operations”. *Proceedings of the 2000 Winter Simulation Conference*, 2000.

[ULAM91] ULM, A.M. *Adventures of a Mathematician*, University of California Press, 1991.

[VENK94] VENKATESH, S.J., SMITH, B.L., DEUERMEYER, B.L. y CURRY, G.L. “Deadlock Properties in Discrete Simulation Systems”. *Proceedings of the 1994 IEEE International Conference on Robotics and Automation* , San Diego, CA, 1994.

[VERN87] VERNON, M., ZAHOEJAN, J. y LAZOWSKA, E.D. “A comparison of performance Petri Nets and Queuing Network Models”. *Proceedings on the International Workshop on Modelling Techniques and Performance Evaluation*, París (Francia), 1987.

© Los autores, 2002; © Edicions UPC, 2002.

[VINC98] VINCENT, S. "Input Data Analysis". *Handbook of Simulation*. Ed. Banks, J., John Wiley & Sons, 1998.

[VISW92] VISWANADHAM, N., NARAHARI, Y. *Performance Modeling of Automated Manufacturing Systems.* Prentice-Hall, New Jersey, 1992.

[WILL01] WILLIAMS, E.J. y ZOTTOLO, M. "Upgrading Drive-Through Services of a Credit Union Via Simulation". *The International Workshop on Harbour, Maritime and Multimodal Logistics Modelling and Simulation,* Marsella 2001.

[ZEIG00] ZEIGLER, B.P., PRAEHOFER, H. y KIM, T.G. *Theory of Modeling and Simulation: Integrating Discrete Event and Continuous Complex Dynamic Systems*. Academic Press, 2000.

[ZHOU93] ZHOU, M., McDERMOTT, K. y PATEL, P.A. "Petri Net Synthesis and Analysis of a Flexible Manufacturing System Cell". *IEEE Transactions on Systems, Man, and Cybernetics,* vol. 23, n. 2, Marzo 1993.

© Los autores, 2002; © Edicions UPC, 2002.

1 Introducción a las técnicas de simulación digital

La simulación digital es una técnica que permite imitar (o *simular*) en un ordenador el comportamiento de un sistema físico o teórico según ciertas condiciones particulares de operación. Para analizar, estudiar y mejorar el comportamiento de un sistema mediante las técnicas de simulación digital es necesario primero desarrollar un modelo conceptual que describa las dinámicas de interés, y luego implementarlo en un simulador para poder analizar los resultados.

Aunque la simulación digital es una técnica relativamente reciente y en constante evolución, el uso de la simulación como metodología de trabajo es una actividad muy antigua, y podría decirse que inherente al proceso de aprendizaje del ser humano. Es el caso, por ejemplo, de un niño pequeño jugando con unos objetos que no son más que representaciones, a escala, de objetos reales. Para poder comprender la realidad y toda la complejidad que un sistema puede conllevar, ha sido necesario construir artificialmente objetos y experimentar con ellos dinámicamente antes de interactuar con el sistema real. La simulación digital puede verse como el equivalente electrónico a este tipo de experimentación. Para ello será necesario construir objetos (modelos) que representen la realidad de tal modo que puedan ser interpretados por un ordenador.

A partir de esta primera definición -muy genérica- se desprende que el uso de las técnicas de simulación digital para la solución de problemas es un campo interdisciplinario muy amplio, tanto por la variedad de sistemas que pueden ser considerados, como por la diversidad de contextos que pueden describirse.

Estas técnicas han adquirido recientemente una importancia cada vez más relevante en la resolución de diferentes tipos de problemas prácticos; es usual encontrar aplicaciones en campos tales como ingeniería, economía, medicina, ciencias de la vida, así como ecología y ciencias sociales. De hecho, la formación en el desarrollo de modelos matemáticos y su ejecución en simuladores digitales está presente en muchas titulaciones universitarias.

Aunque existen diferentes enfoques posibles en la formación en el campo de la simulación digital, la orientación que pretenden dar los autores en el presente libro consiste en introducir una metodología de modelado de sistemas orientada a eventos discretos, así como los mecanismos que permiten a los simuladores ejecutar los modelos. Estos conocimientos son, en opinión de los autores, imprescindibles para un correcto uso de las técnicas de simulación.

© Los autores, 2002; © Edicions UPC, 2002.

1.1 Tipos de sistemas

Para poder introducir el concepto de modelo de un sistema, así como presentar los tipos de modelos de simulación y sus principales características, es imprescindible definir previamente qué se entiende por sistema.

Un sistema puede definirse como una colección de objetos o entidades que interactúan entre sí para alcanzar un cierto objetivo. Nótese que la colección de objetos de un sistema puede ser tan solo un subconjunto de los objetos que deberían ser considerados en un sistema si el objetivo que se pretendiera alcanzar fuera otro.

Si se considera, por ejemplo, el estudio sobre el número de cajeros de que debe disponer un supermercado para ofrecer un buen servicio a sus clientes, los objetos del sistema consistirían principalmente en los clientes en espera de ser atendidos, y en los cajeros/as que realizan dicho servicio. En cambio, si se desea estudiar la atención a las necesidades de los clientes, el sistema debería contemplar adicionalmente entidades tales como los productos y su consumo.

El estado de un sistema puede ser definido como el conjunto mínimo de variables necesarias para caracterizar o describir todos aquellos aspectos de interés del sistema en un cierto instante de tiempo. A estas variables las denominamos *variables de estado*. Cabe reiterar que el número y tipo de éstas vienen condicionados por los objetivos de nuestro estudio. Así pues, en el sistema ejemplo descrito, las variables de estado podrían ser el estado de cada uno de los cajeros (en este caso, disponible u ocupado), y el número total de clientes en el supermercado.

Sin perder generalidad y considerando como finalidad de los experimentos tan solo el estudio del comportamiento de un sistema en el dominio temporal, se pueden clasificar los sistemas en: *continuos, discretos, orientados a eventos discretos* y *combinados*, atendiendo tan sólo a la relación entre la evolución de las propiedades de interés y la variable independiente tiempo.

- *Sistemas continuos*: las variables del estado del sistema evolucionan de modo continuo a lo largo del tiempo. Un ejemplo de este tipo de sistemas es la evolución de la temperatura en una habitación durante cualquier intervalo de tiempo, o bien la evolución del nivel de líquido en un tanque (figura 1.1).

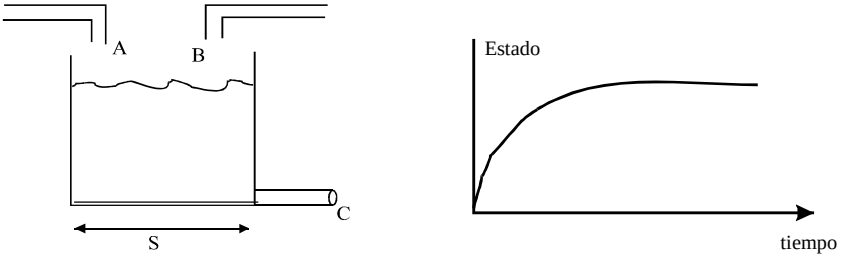


Figura 1.1 Evolución de un sistema continuo

- *Sistemas discretos*: se caracterizan porque las propiedades de interés del sistema cambian únicamente en un cierto instante o secuencia de instantes, y permanecen constantes el resto del tiempo. La

© Los autores, 2002; © Edicions UPC, 2002.

secuencia de instantes en los cuales el estado del sistema puede presentar un cambio, obedece normalmente a un patrón periódico.

- *Sistemas orientados a eventos discretos*: al igual que los sistemas discretos, se caracterizan porque las propiedades de interés del sistema cambian únicamente en una secuencia de instantes de tiempo y, podemos considerar que permanecen constantes el resto del tiempo. La secuencia de instantes en los cuales el estado del sistema puede presentar un cambio, obedece a un patrón aleatorio.
- *Sistemas combinados*: aquellos que combinan subsistemas que siguen filosofías continuas o discretas, respectivamente. Es el caso de los sistemas que poseen componentes que deben ser necesariamente modelados según alguno de dichos enfoques específicos.

EJEMPLO 1.1 Peaje de una autopista

Un proceso muy conocido que ilustra el comportamiento de un sistema orientado a eventos discretos, es el peaje de una autopista, al cual llegan coches de modo aleatorio y en el cual se desea prever el tiempo de espera de los coches en las colas en función del número de taquillas abiertas. Considerando, por ejemplo, los cambios de estado de una taquilla (libre, ocupado), puede observarse que los instantes de tiempo en los cuales aparecerá un cambio, obedecen a un patrón aleatorio. En la figura 1.2 (izquierda) se ha representado el estado de una de las taquillas de un peaje, indicando con el valor 1 que la taquilla se encuentra libre y con valor 0 que la taquilla se encuentra ocupada. En contraposición, en la misma figura (derecha) también se ha representado la evolución de la señal de control generada por un ordenador con un periodo de muestreo constante (sistema discreto).

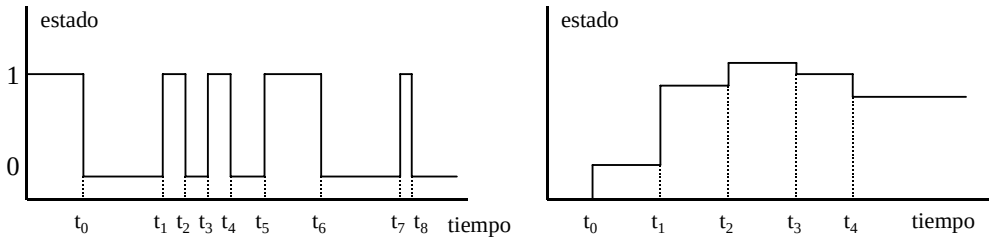


Figura 1.2 Evolución de un sistema discreto

1.2 Tipos de modelos

Existen otras alternativas a las técnicas de simulación digital para imitar (simular) el comportamiento de un sistema. Alternativas típicas son la construcción de un prototipo a escala del sistema real (plantas piloto, procesos en miniatura,...), la representación analógica del sistema mediante circuitos eléctricos, la descripción cualitativa del sistema, o la analogía con otros sistemas físicos o biológicos, como podría ser la experimentación de drogas en animales para prever sus efectos en las personas.

Todas estas técnicas tienen en común que, para imitar el comportamiento de un sistema, requieren (de algún modo) la descripción de las características internas (mecanismos) del sistema en cuestión para

© Los autores, 2002; © Edicions UPC, 2002.

prever su respuesta. La descripción de las características de interés de un sistema se conoce como *modelo del sistema*, y el proceso de abstracción para obtener esta descripción se conoce como *modelado***.**

Existen muchos tipos de modelos (modelos físicos, modelos mentales, modelos simbólicos) para representar los sistemas en estudio, pero puesto que uno de los objetivos para los cuales se van a desarrollar los modelos es su uso en entornos de simulación digital, es necesario que los modelos formalicen el conocimiento que se tiene del sistema de modo conciso, sin ambigüedades (interpretación única), y que puedan ser procesados por un ordenador. Estas características determinan el uso de *modelos simbólicos matemáticos* como herramienta para representar las dinámicas de interés de cualquier sistema en un entorno de simulación digital. Los modelos simbólicos matemáticos mapean las relaciones existentes entre las propiedades físicas del sistema que se pretende modelar en las correspondientes estructuras matemáticas. El tipo de formalización matemática que se utilice va a depender de las características intrínsecas de las dinámicas de interés que se quieran representar.

La descripción en términos matemáticos de un problema real no es una metodología de trabajo propia de la simulación digital, sino que es inherente a la mayoría de las técnicas que se utilizan para solventar cualquier tipo de problema, las cuales suelen seguir unas pautas que, de modo general, se han resumido en la figura siguiente:

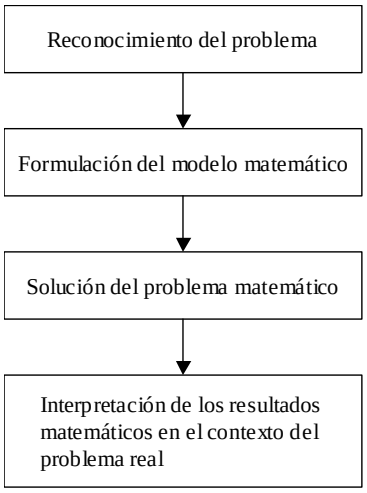


Figura 1.3 Etapas en la explotación de modelos matemáticos

Aunque existe una gran diversidad de metodologías para el desarrollo de modelos matemáticos de sistemas físicos, existen un conjunto de consideraciones que se deben tener en cuenta para garantizar una representación eficiente del sistema real:

- Un modelo se desarrolla siempre a partir de una serie de aproximaciones e hipótesis y, consecuentemente, representa tan sólo parcialmente la realidad.
- Un modelo se construye para una finalidad específica y debe ser formulado para que sea útil a dicho fin.

© Los autores, 2002; © Edicions UPC, 2002.

- Un modelo tiene que ser por necesidad un compromiso entre la simplicidad y la necesidad de recoger todos los aspectos esenciales del sistema en estudio.

Así pues, un buen modelo debe preservar ambas propiedades:

- Representar adecuadamente aquellas características del sistema que son de nuestro interés.
- Ser una representación abstracta de la realidad lo suficientemente sencilla como para facilitar su mantenimiento, adaptación y reutilización.

Atendiendo a las características que debe poseer un buen modelo, así como los objetivos del estudio de simulación, los modelos de simulación suelen clasificarse de diferentes formas:

Modelos estáticos respecto a *modelos dinámicos*

Los *Modelos Estáticos* suelen utilizarse para representar el sistema en un cierto instante de tiempo; y en su formulación no se considera el avance del tiempo. A título ilustrativo, considérese un modelo matemático simple de cantidad de material en un almacén de una fábrica:

Stock = Stock *inicial + Material entrada - Material consumido por la fábrica*

Este tipo de modelos es muy útil cuando el sistema se encuentra en equilibrio (no evoluciona respecto al tiempo). Si se cambia el punto de equilibrio alterando uno o más de los valores del sistema, el modelo permite deducir el resto de los valores, pero no muestra la manera en que cambiaron.

En contraposición a los modelos estáticos, los *modelos dinámicos* permiten deducir cómo las variables de interés del sistema en estudio evolucionan respecto al tiempo. Un ejemplo de modelo dinámico es la evolución de material en un *stock*, que depende de los flujos de entrada y salida, cada uno de los cuales conlleva implícito la evolución del tiempo.

Evolución del stock = *Flujo de entrada – Flujo de salida*

Las ecuaciones siguientes describen matemáticamente la evolución de un stock suponiendo que las variables de interés evolucionan de manera continua o discreta respectivamente, donde F_i y F_o representan los flujos de entrada y salida del almacén.

$$\frac{dS}{dt} = F_i(t) - F_o(t)$$

$$S(k+1) = S(k) + F_i(k) - F_o(k)$$

Modelos deterministas respecto a *modelos estocásticos*

Un modelo se denomina *determinista* si su nuevo estado puede ser completamente definido a partir del estado previo y de sus entradas. Es decir, ofrece un único conjunto de valores de salida para un conjunto de entradas conocidas.

© Los autores, 2002; © Edicions UPC, 2002.

Los *modelos estocásticos* requieren de una o más variables aleatorias para formalizar las dinámicas de interés. En consecuencia, el modelo no genera un único conjunto de salida cuando es utilizado para realizar un experimento, sino que los resultados generados son utilizados para estimar el comportamiento real del sistema.

Modelos continuos respecto a *modelos discretos*

Los *modelos continuos* se caracterizan por representar la evolución de las variables de interés de forma continua. En general suelen utilizarse ecuaciones diferenciales ordinarias si se considera simplemente la evolución de una propiedad respecto al tiempo, o bien ecuaciones en derivadas parciales si se considera también la evolución respecto al espacio.

De modo análogo a la definición de modelos continuos, los *modelos discretos* se caracterizan por representar la *evolución* de las variables de interés de forma discreta.

Es importante notar a partir de la clasificación de modelos realizada, que es posible describir un sistema continuo mediante un modelo discreto y, al revés, también es posible describir un sistema discreto mediante un modelo continuo. La decisión de utilizar un modelo continuo o un modelo discreto depende de los objetivos particulares de cada estudio y no tanto de las características del sistema. Así pues, por ejemplo, es posible encontrar modelos continuos de flujos de coches en una autopista donde se ha escogido una formulación continua cuando los objetivos del estudio se centran pongamos por caso en evaluar la evolución de los flujos ante un accidente o bien cuando se recorta un carril, en que el movimiento de un coche carece de importancia.

Antes de finalizar la presentación del concepto de modelo de un sistema, se introducen a continuación otras definiciones igualmente válidas en el contexto de este libro, así como algunas de las características que debe cumplir un buen modelo:

- Un modelo es un objeto o concepto que utilizamos para representar cualquier otra entidad (un sistema). Así pues, mediante un proceso de abstracción, se muestran en un formato adecuado las características de interés de un objeto (sistema) real o hipotético.
- Un modelo es una representación simplificada de un sistema que nos facilitará explicar, comprender, cambiar, preservar, prever y posiblemente controlar el comportamiento de un sistema.
- Un modelo es el sustituto de un sistema físico concreto.
- Un modelo debe representar el conocimiento que se tiene de un sistema de modo que facilite su interpretación, formalizando tan sólo los factores que son importantes para los objetivos de modelado.
- Un modelo debe ser tan sencillo como sea posible, porque el desarrollo de modelos universales es impracticable y poco económico, siempre y cuando represente adecuadamente los aspectos de interés.

1.3 Modelos de simulación de eventos discretos

Los Modelos de eventos discretos son *modelos dinámicos, estocásticos* y *discretos* en los que las variables de estado cambian de valor en instantes no periódicos del tiempo. Estos instantes de tiempo

© Los autores, 2002; © Edicions UPC, 2002.

se corresponden con la ocurrencia de un evento. Por tanto, *un evento se define como una acción instantánea que puede cambiar el estado del modelo*.

Otros elementos de interés de los modelos de eventos discretos son:

- *Actividades*: son las tareas o acciones que tienen lugar en el sistema. Están encapsuladas entre dos eventos. Por ejemplo, la reparación de una máquina, el procesado de una pieza o el transporte de un cliente. Las actividades generalmente tienen duración temporal y, normalmente, precisan del uso de recursos.
- *Entidades*: son el conjunto de objetos que constituyen o fluyen por el sistema. Pueden ser temporales o permanentes. En la bibliografía a menudo se emplea el concepto de *entidad* para aludir sólo a las entidades temporales, aunque en este libro el concepto es más amplio dado que también incluye a los recursos que nos permiten mantener la coherencia con los múltiples formalismos empleados en el proceso de abstracción del sistema.
- *Entidades temporales*: son los objetos que se procesan en el sistema, como por ejemplo las piezas, los clientes o los documentos. Entidades diferentes pueden tener características diferentes que denominaremos atributos: por ejemplo, precio, prioridad, estado o tamaño. Las entidades temporales son los objetos que llegan, se procesan y salen del sistema.
- *Recursos* o *entidades permanentes*: son los medios gracias a los cuales se pueden ejecutar las actividades. Los recursos definen quién o qué ejecuta la actividad. Los recursos pueden tener características como capacidad, velocidad, averías y reparaciones o tiempo de ciclo. Ejemplos de recursos son las máquinas, los elementos de transporte o las personas.

EJEMPLO 1.2

En el modelo del sistema del ejemplo 1.1 descrito anteriormente, los eventos que pueden ocurrir son:

- La llegada de un vehículo a la cola.
- Inicio del pago del peaje.
- Fin del pago y salida del sistema.

Las entidades temporales son los vehículos que pasan por el peaje. La actividad única del proceso es el pago en la taquilla y el recurso de interés es la taquilla o taquillas habilitadas.

1.4 Alternativas en la simulación de modelos de eventos discretos

Resumiendo algunos de los conceptos introducidos en las secciones anteriores, el modelo de un sistema representa las dinámicas de interés del sistema en un estudio particular, y la simulación consiste en realizar un experimento sobre el modelo del sistema de tal modo que los resultados generados por el simulador permitan prever con cierta exactitud los resultados que se obtendrían realizando el mismo experimento sobre el sistema real.

Con el objetivo de ilustrar diferentes alternativas para experimentar con el modelo de simulación de un sistema orientado a eventos discretos, considérese el sistema de procesado de órdenes ilustrado en

© Los autores, 2002; © Edicions UPC, 2002.

la figura 1.4, donde las tres actividades más significativas del sistema en estudio son: la recepción, el procesado y finalmente la expedición de las órdenes recibidas. Se considera que hay cuatro trabajadores por cada turno diario de 8 horas y que la franja horaria normal de trabajo es desde las 9 de la mañana hasta las 5 de la tarde. No obstante, la jornada laboral se alarga si no ha sido posible expedir todas las órdenes recibidas a lo largo del día.

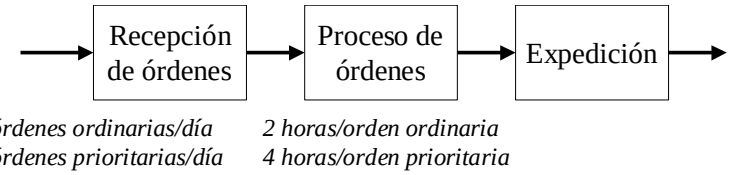


Figura 1.4 Esquema del procesado de órdenes

Los parámetros más significativos del sistema son:

- En promedio, se reciben 10 órdenes cada día.
- Hay dos tipos de órdenes, las ordinarias y las prioritarias: el 40% de las órdenes son ordinarias y el 60% restante son prioritarias.
- En promedio, una orden ordinaria requiere 2 horas de proceso y una orden prioritaria 4 horas.
- Solo se aceptan órdenes hasta la 1 del mediodía.

1.4.1 Simulación del modelo estático

A partir de una formulación estática del sistema en la que no interviene directamente el tiempo (basada en valores promedio), es posible analizar el comportamiento del sistema llegando a conclusiones erróneas, como que no existe ningún retraso en las órdenes y que los recursos humanos están aprovechados en un 100%:

órdenes ordinarias = 4 órdenes/día * 2 horas/orden = 8 horas/día

órdenes prioritarias = 6 órdenes/día * 4 horas/orden = 24 horas/día

capacidad necesaria = 8 + 24 = 32 horas/día

capacidad disponible = 4 trabajadores/día * 8 horas/día = 32 horas/día

porcentaje de utilización = capacidad necesaria/cap. disponible * 100=100%

1.4.2 Simulación a mano

Para realizar una simulación del modelo orientado a eventos discretos (modelo de simulación estocástico y discreto) es necesario formalizar los distintos eventos que afectan a las variables de interés del estudio, los instantes de tiempo en que pueden aparecer dichos eventos, así como las actividades que deben realizarse como consecuencia de la aparición de dichos eventos.

En la figura 1.5 se muestra en forma de organigrama las acciones asociadas al evento de llegada de una orden. Si todos los trabajadores están ocupados, la orden se espera en la cola. En caso contrario, se asigna un trabajador a la orden y se inicia su proceso.

© Los autores, 2002; © Edicions UPC, 2002.

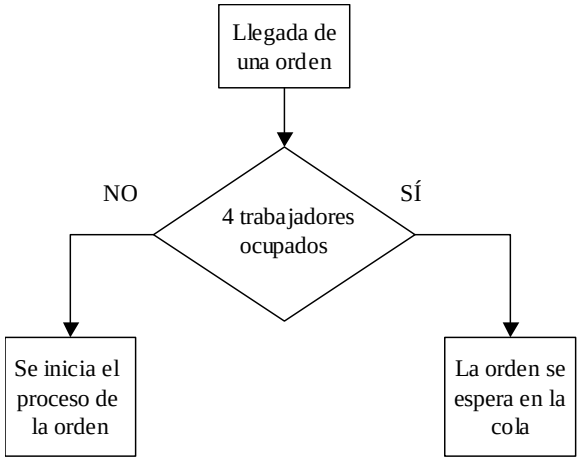


Figura 1.5 Diagrama de flujo del evento de llegada de una orden

En la figura 1.6 se representan las acciones ligadas al evento de expedición de una orden. Al finalizar el proceso de una orden, uno de los trabajadores queda libre. Si hay órdenes en cola, el trabajador selecciona la orden más prioritaria e inicia su procesado. En caso contrario, el trabajador permanece a la espera de la llegada de nuevas órdenes.

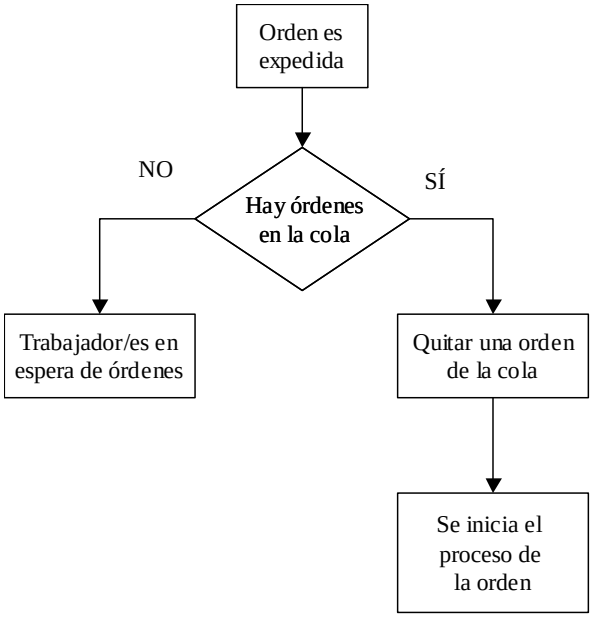


Figura 1.6 Diagrama de flujo del evento de expedición de una orden

© Los autores, 2002; © Edicions UPC, 2002.

Para analizar el comportamiento del sistema se simula la dinámica del proceso en el cual la llegada de órdenes es aleatoria. La figura siguiente muestra para cada una de las órdenes la hora de llegada, el tiempo que está en cola, el tiempo de procesado y la hora de expedición.

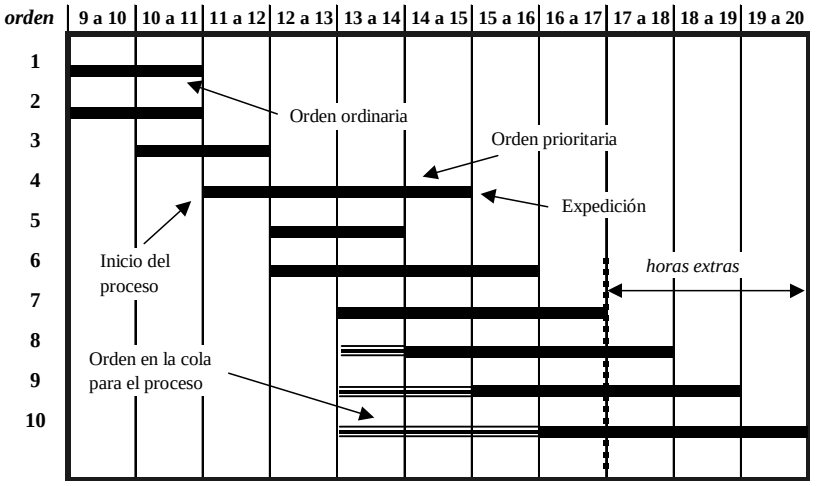


Figura 1.7 Diagrama de la simulación a mano

Las primeras dos órdenes llegan a las 9 de la mañana. La tercera orden llega una hora más tarde, a las 10 de la mañana. Dado que estas primeras órdenes son ordinarias y hay trabajadores para procesarlas, las tres órdenes son expedidas al cabo de dos horas. Más adelante, a las 13 horas, llega la orden número 8. En este instante, los cuatro trabajadores están ocupados. En consecuencia, no se puede iniciar su proceso hasta una hora más tarde después de ser expedida la orden número 5. Para completar las órdenes 8, 9 y 10 es preciso que 3 trabajadores alarguen su jornada de trabajo más allá de las 5 de la tarde.

A continuación se muestra también de forma tabular los resultados de la simulación a mano. La primera columna muestra en qué instante ocurre cada evento. En la cuarta columna se muestra el tipo de evento, evento de llegada de una orden o evento de expedición de una orden. En la columna número dos se muestra la orden que se recibe o expide y en la siguiente columna, la tercera, se muestra el tipo de orden, ordinaria o prioritaria. En la quinta columna se muestra en número de órdenes en la cola. En la sexta columna se muestra el número de órdenes en el sistema (desde que llega la orden hasta que ésta es expedida). La penúltima columna muestra el tiempo que está cada orden en la cola a la espera de ser procesada. La última columna muestra, en el evento de expedición, el tiempo total que ha estado la orden en el sistema, o sea, la suma del tiempo en la cola más el tiempo en proceso.

Las últimas cuatro columnas muestran estadísticas importantes para analizar el comportamiento del sistema. Estas estadísticas son recogidas cada vez que hay un evento. Por ejemplo, el número de órdenes en la cola alcanza su valor máximo a las 13 horas. Las últimas tres órdenes están en la cola una, dos y tres horas respectivamente.

© Los autores, 2002; © Edicions UPC, 2002.

Tiempo (hora)	*Número de orden*	*Tipo de orden*	*Tipo de evento*	*Número en cola*	*Número en sistema*	*Tiempo en cola*	*Tiempo en sistema*
9	1	Ordinaria	Llegada	0	1	-	-
9	2	Ordinaria	Llegada	0	2	-	-
10	3	Ordinaria	Llegada	0	3	-	-
11	1	Ordinaria	Expedición	0	2	-	2 h
11	2	Ordinaria	Expedición	0	1	-	2 h
11	4	Prioritaria	Llegada	0	2	-	-
12	3	Ordinaria	Expedición	0	1	-	2 h
12	5	Ordinaria	Llegada	0	2	-	-
12	6	Prioritaria	Llegada	0	3	-	-
13	7	Prioritaria	Llegada	0	4	-	-
13	8	Prioritaria	Llegada	1	5	-	-
13	9	Prioritaria	Llegada	2	6	-	-
13	10	Prioritaria	Llegada	3	7	-	-
14	5	Ordinaria	Expedición	2	6	-	2 h
15	4	Prioritaria	Expedición	1	5	-	4 h
16	6	Prioritaria	Expedición	0	4	-	4 h
17	7	Prioritaria	Expedición	0	3	-	4 h
19	8	Prioritaria	Expedición	0	2	1 h	5 h
19	9	Prioritaria	Expedición	0	1	2 h	6 h
20	10	Prioritaria	Expedición	0	0	3 h	7 h

Tabla 1.1 Tabla de la simulación manual del proceso de órdenes

A continuación se resume el comportamiento del proceso observado mediante la simulación manual:

tiempo de ciclo promedio para órdenes ordinarias = 8 h/4 órdenes = 2 h/orden

tiempo de ciclo promedio para las prioritarias = 30 h/6 órdenes = 5 h/orden

tiempo promedio en la cola = 6 h/10 órdenes = 0.6 h/orden

tiempo máximo en la cola = 3 h

nivel de servicio promedio = 7 órdenes a tiempo / 10 órdenes = 70%

3 trabajadores han tenido que trabajar un total de 6 horas extras para completar las órdenes

Aunque inicialmente parecía que el diseño del proceso era el adecuado, las llegadas aleatorias provocan colas y retardos posteriores en el procesado de las órdenes. En este ejemplo se ha observado que las llegadas aleatorias provocan desviaciones significativas respecto al comportamiento esperado inicialmente.

Para efectuar la simulación anterior, se ha supuesto conocido el número de órdenes que llegan en cada franja horaria. Sin embargo, en la práctica diaria, esta secuencia precisa de tiempos puede no ser conocida dado que:

- el proceso simulado todavía no existe en la realidad,
- o se desea simular el comportamiento de un sistema real según condiciones diferentes de operación.

© Los autores, 2002; © Edicions UPC, 2002.

Aunque no se disponga de la información anterior, sí que, en general, se dispone de información sobre la distribución del número de órdenes que llegan cada hora. En este caso, la información pudo ser obtenida de forma precisa a partir de una muestra real o de forma aproximada a partir del conocimiento de que se disponía del sistema.

Se supone conocida la probabilidad de que en una hora determinada lleguen 1, 2, 3 o 4 órdenes. La distribución de probabilidades se muestra en la tabla siguiente

Órdenes / hora	Probabilidad (%)	Prob. acumulada
1	40	40
2	30	70
3	20	90
4	10	100

Tabla 1.2 Distribución de probabilidades

Una forma de simular cuál será el tiempo de proceso de la siguiente orden de trabajo consiste en introducir 100 bolas en un saco, numeradas de la 01 hasta la 100, y sacar una al azar. Si el valor extraído está comprendido entre el 01 y el 40, el número de órdenes será 1. Si está comprendido entre el 41 y el 70, llegarán 2 órdenes, si está comprendido entre el 71 y el 90, llegarán 3, y si está comprendido entre el 91 y el 100, llegarán 4 órdenes. Para proseguir, se vuelve a colocar la bola extraída en el saco y se procede de forma exactamente igual para obtener el número de órdenes de la hora siguiente. Nótese que, dado que del total de bolas, 40 están comprendidas entre los valores 01 y 40, la probabilidad de que el número de órdenes sea 1 será de 0.4 (40%).

Hora	Número al azar	Número de órdenes
9	54	2
10	12	1
11	36	1
12	60	2
13	90	4

Tabla 1.3 Extracción de valores aleatorios

1.4.3 Simulación digital mediante un lenguaje de propósito general

Los lenguajes que pueden utilizarse para la simulación de un sistema orientado a eventos discretos pueden ser tanto lenguajes de programación de propósito general (por ejemplo, el Pascal o el C++), como lenguajes de simulación de propósito general (como el GPSS, Simscript, Siman, Slamsystem), los cuales aparecieron a partir de 1960. La principal característica de los lenguajes de programación es que ofrecen una librería de instrucciones suficientemente completa como para permitir programar cualquier tipo de modelo, por complejo que éste sea. Cualquier programador con cierta experiencia en el uso de la variable tiempo para coordinar, sincronizar o bloquear procesos, podría con cierta facilidad codificar el modelo de simulación descrito en los organigramas anteriores (figura 1.5 y figura 1.6), considerando adicionalmente la necesidad de generar números aleatorios que representen los tiempos entre 2 llegadas de órdenes sucesivas, y números aleatorios que representen el tiempo invertido en el procesamiento de una orden.

© Los autores, 2002; © Edicions UPC, 2002.

Aunque en el ejemplo presentado la codificación del modelo de simulación en un lenguaje de propósito general es una tarea relativamente sencilla, en general, la codificación del modelo de un sistema orientado a eventos discretos requiere una inversión considerable de tiempo y personal especializado para el desarrollo del modelo de simulación, con lo que el mantenimiento del modelo para futuros experimentos suele presentar grandes limitaciones si no se ha formalizado correctamente.

1.4.4 Experimentación mediante un entorno de simulación

El principal objetivo para el que se desarrollan entornos de simulación es facilitar el uso de la simulación en la fase de programación del modelo, a costa de restringir los campos de aplicación. Los primeros entornos de simulación orientados a eventos discretos aparecieron a principios de los 80, y pueden ser definidos como una *toolbox* con un número limitado de herramientas de programación de alto nivel. Aunque estos entornos permiten reducir considerablemente las tareas de programación, las directivas de programación con las que se trabaja no ofrecen aún la flexibilidad que ofrecen los lenguajes de programación de propósito general.

En la figura 8 se muestra una interfaz de desarrollo de modelos de un simulador (Arena), en la que puede observarse (parte izquierda) las herramientas de programación gráfica de alto nivel, que permiten el desarrollo del modelo de simulación a partir de la interconexión de los módulos predefinidos.

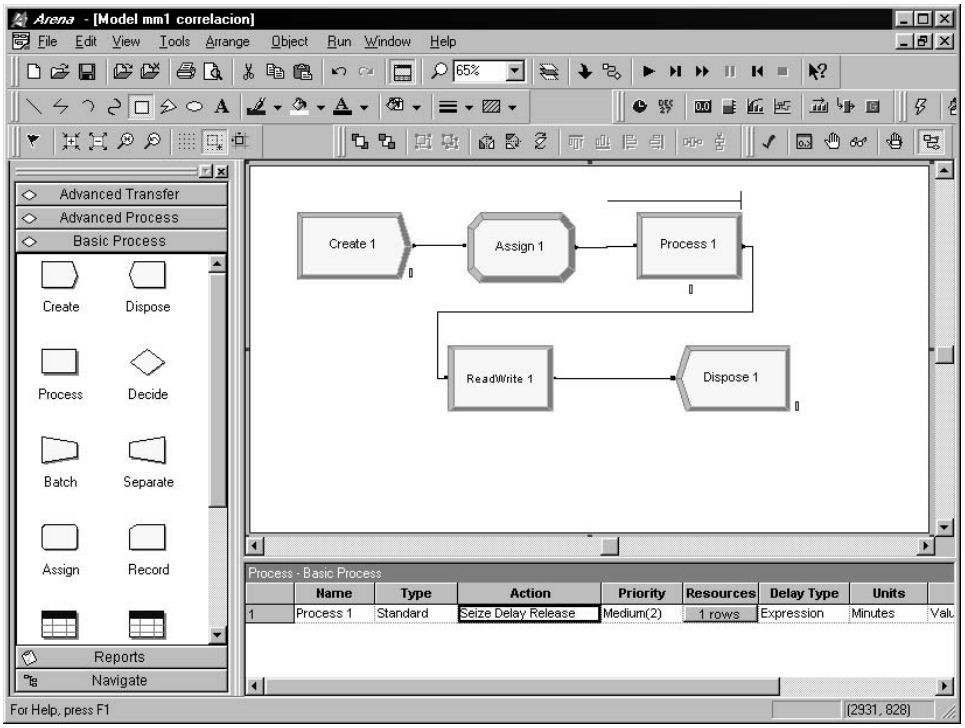


Figura 1.8 Interfaz gráfico de modelado de un simulador

© Los autores, 2002; © Edicions UPC, 2002.

En opinión de los autores, los entornos de simulación actuales (como por ejemplo Arena, Witness o LeanSIM) ofrecen la misma flexibilidad que los lenguajes de programación, puesto que incorporan bloques que pueden ser programados desde lenguajes de alto nivel como C y Visual Basic. La diferencia en cuanto a prestaciones, capacidades y limitaciones entre simuladores y lenguajes de simulación está prácticamente desapareciendo.

1.5 Ventajas y desventajas de la simulación

La disponibilidad en el mercado de entornos de simulación orientados a campos específicos, con librerías preprogramadas y validadas de componentes típicos, junto con la capacidad de programación gráfica y de visualización, han hecho posible que la simulación sea una técnica cada vez más utilizada en el análisis y la mejora de sistemas. Thomas y DaCosta [THOM79] dieron a los analistas de 137 grandes firmas una lista de herramientas y les preguntaron cuales de ellas usaban. El *análisis estadístico* resultó clasificado en primer lugar, el 93% de las firmas dijeron utilizarlo, seguido por la *simulación* con el 84%. La simulación se situó delante de técnicas clásicas de *investigación operativa*: programación lineal, teoría de colas, PERT/CPM, teoría de inventarios o programación no lineal. Estudios posteriores [FORG83, HARP89, LANE93] confirman que la simulación es la segunda técnica de interés (por detrás de la estadística) en las grandes empresas.

Algunos de los objetivos para los cuales pueden utilizarse muy satisfactoriamente las técnicas de simulación digital de sistemas orientados a eventos discretos son:

- El análisis y estudio de la incidencia sobre el rendimiento global del sistema de pequeños cambios realizados sobre alguno de sus componentes. En la actualidad, los sistemas logísticos son considerados sistemas complejos, entre otros motivos, porque un pequeño cambio o una mala sincronización entre los elementos que intervienen en los mismos pueden derivar fácilmente en un comportamiento no deseado, lo que frecuentemente implica una importante repercusión económica.
- Cambios en la organización de una empresa, así como en la gestión de la información pueden ser fácilmente simulados, y los efectos sobre el sistema real pueden ser analizados a partir de la experimentación con el modelo.
- El conocimiento que se obtiene en el desarrollo de un modelo de simulación es de gran interés para poder sugerir posibles mejoras en su rendimiento, más allá del estricto encargo del estudio de simulación propiamente dicho.
- La observación de los resultados que se obtienen de un simulador a partir de experimentar con ciertos parámetros del modelo, así como los cambios en las entradas, permiten inferir aspectos relativos a la sensibilidad del sistema y qué variables son las que más pueden beneficiar al rendimiento del mismo.
- La simulación suele ser utilizada también con una perspectiva pedagógica para ilustrar y facilitar la comprensión de los resultados que se obtienen mediante técnicas analíticas.
- Experimentar con condiciones de operación que podrían ser peligrosas o de elevado coste económico en el sistema real.

© Los autores, 2002; © Edicions UPC, 2002.

En resumen, las técnicas de simulación pueden generalmente ser utilizadas como una metodología de trabajo barata y segura que,

- Permite responder muy satisfactoriamente a preguntas del tipo “qué ocurriría si realizamos este cambio en...”
- Contribuye a la reducción del riesgo inherente a la toma de decisiones.

Por otro lado, cabe notar que la toma de decisiones basada únicamente en un estudio realizado mediante la simulación, conlleva un elevado riesgo si el modelo en el cual se basa el estudio no ha sido validado y las fases del proyecto convenientemente verificadas. Nótese que la popularización de las técnicas de simulación ha ido paralela al incremento de potencia y flexibilidad de las nuevas herramientas. Su facilidad de manejo y sus entornos de visualización/animación han facilitado la incorporación de nuevos usuarios menos “especializados”. Si bien se ha popularizado su utilización, *también se ha incrementado su mal uso*. Un modelo de simulación no puede ser mejor que los datos y las técnicas empleadas en su construcción.

En este libro se pretende presentar las pautas que deberían seguirse en el desarrollo de un proyecto de simulación, tanto para poder garantizar la fiabilidad de los resultados, como para facilitar la implementación del modelo.

1.6 Campos de aplicación de la simulación orientada a eventos discretos

En la actualidad, la mayor parte de las organizaciones, empresas y procesos productivos han crecido en complejidad y han visto aumentados los requerimientos de competitividad, flexibilidad y calidad en sus actividades; por ello han debido innovar y adaptarse a los constantes cambios provocados por su pertenencia a un mercado cada vez más global. Posiblemente influenciadas por las nuevas tendencias en el sector, muchas empresas se han visto involucradas en un proceso de reingeniería de su proceso productivo.

La falta de herramientas analíticas que ayuden y faciliten la toma de decisiones es uno de los principales problemas con los que se encuentran los directivos, que ven la necesidad de mejorar el rendimiento de aquellos sistemas cuyo comportamiento depende de un número elevado de variables de decisión. Determinar cambios en los procedimientos, en las reglas que utilizan los distintos departamentos en la toma de decisiones, en los flujos de información, así como cambios en la organización y en las políticas de funcionamiento tradicionales de la empresa son algunos ejemplos de tareas que deben ser realizadas para reaccionar ante perturbaciones y adaptarse a los cambios del mercado a los que se ven constantemente sometidos.

La dificultad que comportan estos procesos se hace evidente en una gran diversidad de campos en los que, prácticamente, se desconoce la influencia de determinados cambios en algunas variables de decisión, sobre el rendimiento global de la empresa. Considérese, a modo de ejemplo, todos aquellos sistemas que no pueden procesar de modo inmediato las peticiones de trabajo y que, en consecuencia, son retenidas y priorizadas en colas de espera asociadas a los distintos recursos.

El problema es todavía más complicado cuando se pretende optimizar (o en su caso mejorar) el rendimiento de un sistema que presenta un comportamiento estocástico (predecible desde un punto de

© Los autores, 2002; © Edicions UPC, 2002.

vista estadístico) a partir de cambios en la logística del sistema. Ésta suele contemplar no sólo las fluctuaciones estadísticas, sino también las distintas situaciones que se derivan de la dependencia entre sucesos.

Las herramientas de simulación orientadas a eventos discretos ofrecen una plataforma que permite abordar con éxito un proceso de mejora continua de sistemas complejos para los cuales las técnicas analíticas clásicas basadas en el uso de cálculo diferencial, teoría de probabilidades y métodos algebraicos, no pueden ser utilizadas para formalizar de modo sencillo la complejidad de los procesos.

Una consulta rápida a los artículos publicados en la *Winter Simulation Conference* de los últimos años (http://www.informs-cs.org) nos puede dar una buena perspectiva de los diferentes campos de aplicación de las técnicas de simulación orientadas a eventos discretos:

- *Fabricación de procesos*: fue una de las primera áreas beneficiadas por estas técnicas. La simulación se emplea tanto para el diseño como para la ayuda a la toma de decisiones operacionales.
- *Logística*: la simulación contribuye de forma significativa a la mejora de los procesos logísticos en general. Dentro de esta área, se incluye tanto una cadena completa de suministros (s*upply chain*) como la gestión de inventarios de un almacén.
- *Transporte*: ya sea por ferrocarril, por carretera o por vía aérea y ligado o no a decisiones logísticas. Dentro de esta área, la intermodalidad entendida como interconexión de diferentes medios de transporte, ha merecido especial atención en los últimos años. También se ha introducido con fuerza la simulación visual de modelos de tránsito, tanto en lo que se refiere a modelos macroscópicos como microscópicos.
- *Sanidad*: se emplea para la mejora tanto de un departamento hospitalario, como en la logística asociada a los transplantes o a la coordinación médica en una región.
- *Negocios (*Business Processing*)*: simulación de los procesos administrativos y de negocio de una empresa.
- *Construcción*: por ejemplo, la planificación de la construcción en obra civil o la gestión de recursos.
- *Emergencias*: la simulación se emplea para la gestión de la emergencia; por ejemplo, diseño de planes de evacuación en edificios o recintos públicos.
- *Servicios en general:* servicios públicos, gestión de restaurantes, banca, empresas de seguros,...

1.7 Ciclo de vida de un proyecto de simulación

Existe el consenso, entre las personas involucradas en el desarrollo y mantenimiento de modelos de simulación, de que los modelos simples son preferibles a los modelos complejos. A pesar de ello, en muchos proyectos, los modelos de simulación suelen ser largos y complejos. Una posible explicación a la falta de dedicación o de interés en la simplificación de modelos radicaría en la potencia de cálculo de los ordenadores actuales, suficiente para soportar la ejecución de grandes modelos de simulación.

© Los autores, 2002; © Edicions UPC, 2002.

A pesar de ello, cabe destacar que el exceso de complejidad en los modelos no tan sólo repercute en el rendimiento computacional, y también afecta a otros cuestiones, como el tiempo invertido en el desarrollo del modelo, su mantenimiento, verificación, validación, etc.

Aunque parezca un concepto muy intuitivo, no existe ninguna definición o medida de complejidad que haya sido aceptada como estándar por la comunidad científica. Algunos autores relacionan, por ejemplo, la complejidad del modelo con “el nivel de detalle” y otros con “la envergadura del sistema”.

Citamos, a continuación, algunas de las ventajas de trabajar con modelos sencillos:

- Son más fáciles de implementar, validar y analizar.
- Es más fácil y, en cierto modo, menos doloroso, desechar un modelo simple cuando se ha cometido, por ejemplo, un error de diseño que si se trata de un modelo complejo en el cual se han invertido un número de horas y de personal experto considerable.
- Es más fácil cambiar un modelo simple que un modelo complejo si las condiciones o hipótesis de operación sobre el sistema real cambian.
- El tiempo que conlleva un proyecto de simulación se ve considerablemente reducido y pueden presentarse resultados correctamente validados antes de las fechas de entrega si se trabaja con modelos sencillos.

Un proyecto de simulación es dinámico por naturaleza. Los resultados que se van obteniendo a medida que se va desarrollando, ponen a su vez de manifiesto nuevos problemas, así como limitaciones inherentes al sistema en estudio, y pueden forzar a reconsiderar la orientación tomada en un principio. Es más, la motivación del cliente puede también variar a lo largo del mismo, como consecuencia de los resultados obtenidos o por factores externos al propio proyecto. Para tener éxito en un entorno tan dinámico, es necesario emplear una aproximación metodológica correcta. Los elementos que se presentan a continuación tienen como objetivo servir de guía para abordar con éxito un proyecto de simulación.

En la práctica, el éxito en el uso de una tecnología tan potente como la simulación no depende tan sólo de los aspectos puramente técnicos, sino también de otros aspectos complementarios como son:

- La necesidad de tener o ganarse el apoyo de los órganos con capacidad de decisión de las empresas. Este apoyo es absolutamente necesario para poder superar las barreras que suelen aparecer.
- Es importarte inspirar confianza. El cliente o usuario final responsable de la toma de decisiones no tiene, en general, ni el tiempo ni la capacidad suficientes para fiscalizar todo el trabajo efectuado. En consecuencia, se tomarán decisiones sobre la base de los resultados de un estudio de simulación si el grupo que lo ha realizado inspira confianza. En general ésta hay que ganársela a lo largo del estudio efectuando un trabajo serio y riguroso.

La tabla siguiente muestra el conjunto de etapas de un proyecto de simulación. Aunque puede parecer que el desarrollo de un proceso de simulación es un proceso secuencial, en la práctica no es así. Por ejemplo, si el modelo de simulación obtenido no supera la etapa de validación (etapa 5), es posible que sea necesario modificar tanto el modelo conceptual como el de simulación:

© Los autores, 2002; © Edicions UPC, 2002.

Etapa	Descripción
Formulación del problema	Define el problema que se pretende estudiar. Incluye por escrito sus objetivos.
Diseño del modelo conceptual	Especificación del modelo a partir de las características de los elementos del sistema que se quiere estudiar y sus interacciones teniendo en cuenta los objetivos del problema.
Recogida de datos	Identificar, recoger y analizar los datos necesarios para el estudio.
Construcción del modelo	Construcción del modelo de simulación partiendo del modelo conceptual y de los datos.
Verificación y validación	Comprobar que el modelo se comporta como es de esperar y que existe la correspondencia adecuada entre el sistema real y el modelo.
Análisis	Analizar los resultados de la simulación con la finalidad de detectar problemas y recomendar mejoras o soluciones.
Documentación	Proporcionar documentación sobre el trabajo efectuado.
Implementación	Poner en práctica las decisiones efectuadas con el apoyo del estudio de simulación.

Tabla 1.4 Etapas de un proyecto de simulación

- *Formulación del problema:* la especificación de objetivos es una de las tareas más importantes de un proyecto de simulación. Si los objetivos no están claros o son poco concretos, existe el peligro de no abordar correctamente el problema para el cual se ha solicitado el proyecto de simulación y de ser incapaz de responder a las expectativas generadas. En consecuencia, es necesario, en la fase inicial de cualquier proyecto de simulación, saber identificar los objetivos para los cuales se ha optado por el uso de las técnicas de simulación, y formalizarlos de forma que sean precisos, razonables, comprensibles y medibles. Estos objetivos servirán de guía a lo largo del proyecto.

- *Diseño del modelo conceptual:* una vez conocidos los objetivos del proyecto, puede existir la tentación de iniciar la construcción del modelo de simulación de forma inmediata. Esta opción generalmente conduce a la obtención de modelos de simulación con múltiples lagunas y de difícil mantenimiento. Es por ello conveniente formular o especificar el modelo de simulación empleando un nivel de abstracción (modelo conceptual) superior al del propio código. El modelo conceptual especifica las relaciones estructurales más importantes del sistema que se intenta simular y, en consecuencia, constituye un medio de diálogo y de coordinación entre los distintos departamentos o grupos involucrados. El capítulo 2 de este libro aborda el diseño de modelos conceptuales empleando el formalismo de las redes de Petri.

  También corresponde a esta etapa especificar qué resultados o estadísticas esperamos obtener del modelo de simulación para así responder a las preguntas formuladas en la definición de objetivos. Al especificar los resultados, se detecta a menudo qué partes del modelo pueden ser simplificadas o eliminadas dado que no contribuyen a responder a dichas preguntas.

© Los autores, 2002; © Edicions UPC, 2002.

- *Recogida y tratamiento de los datos:* en general, se recomienda cuestionar siempre toda la información y los datos disponibles. ¿Cuál es la fuente?, ¿cuándo se obtuvo?, ¿cómo fue recogida?, ¿tiene ésta sentido?, ¿tenemos insuficientes datos o son excesivos?. Para obtener buenos resultados es condición indispensable disponer de unos buenos datos. Desgraciadamente, en muchos casos no se dispone de ellos. Aun así, se requiere una respuesta a las preguntas planteadas y es necesario efectuar hipótesis razonables en colaboración con el usuario final. Si los datos son limitados o su calidad es dudosa es conveniente ser prudente a la hora de extraer conclusiones sobre la base de los resultados de la simulación. Aun en los casos en los que hay problemas con los datos, el conocimiento adquirido y los resultados obtenidos en el estudio de simulación aportan información valiosa para la toma de decisiones. En el capítulo 3 de este libro se introducen las nociones estadísticas básicas necesarias para el tratamiento y análisis de los datos necesarios para parametrizar el modelo de simulación.

- *Construcción del modelo:* en numerosos proyectos de simulación el esfuerzo se concentra más en la construcción del modelo que en la resolución del problema. En esta misma línea, la obtención de un modelo ejecutable se convierte erróneamente en un objetivo prioritario. La motivación dominante debería ser la comprensión del problema y la obtención de soluciones. Para avanzar más rápidamente en la consecución de estos objetivos, es recomendable construir en primer lugar uno o varios modelos simplificados que caractericen las partes más esenciales del sistema de interés. En el apéndice B se muestran de manera informal técnicas para construir el modelo de simulación a partir de las redes de Petri.

- *Verificación y validación:* en los procesos judiciales se presupone que los encausados son inocentes hasta que no se demuestra su culpabilidad. Bien al contrario, en el campo de la simulación, la experiencia recomienda suponer que todo modelo es incorrecto excepto que se demuestre lo contrario. Los esfuerzos en dotar a los actuales simuladores de potentes herramientas para facilitar su uso curiosamente han contribuido a uno de los principales peligros de la simulación: “Olvidarse del mundo real y aceptar sin reparos los resultados del modelo”. Para tener una garantía razonable de que el modelo de simulación representa la realidad, y como consecuencia, tomar decisiones estratégicas u operacionales basándose en los resultados, es absolutamente necesario verificar y validar antes el modelo de simulación.

  La verificación consiste en comprobar que el modelo se ejecuta correctamente y según las especificaciones (modelo conceptual). La validación consiste en comprobar que las teorías, hipótesis de trabajo así como suposiciones, son correctas. Si el proceso todavía no existe, es necesario contrastar los resultados con expertos del proceso de interés para comprobar si el modelo se comporta tal como ellos esperan. Los conceptos presentados en los capítulos 4 y 5, así como el resto de la información presentada en este libro, pueden facilitar el proceso de verificación y validación del modelo de simulación.

- *Análisis:* consiste en experimentar con el modelo con el objetivo de efectuar inferencias que permitan tomar decisiones con mayor seguridad. En esta etapa se emplean a menudo técnicas analíticas como la reducción de la variancia o el diseño de experimentos.

  En general, el valor añadido más importante de un estudio de simulación no son los resultados finales obtenidos con el modelo. El resultado más valioso es el conocimiento adquirido en el

© Los autores, 2002; © Edicions UPC, 2002.

proceso de análisis que permite aportar argumentos cualitativos y/o cuantitativos justificados a favor o en contra de las diferentes opciones de diseño planteadas. En el capítulo 5 se muestran las técnicas básicas para facilitar la experimentación y análisis de resultados.

- *Documentación:* es importante mantener un documento al día que refleje el estado del proyecto. Por tanto, el documento evolucionará y se enriquecerá en paralelo con el proyecto de simulación. Los objetivos perseguidos con la documentación son:

  1. Reflejar el estado del proyecto en un momento dado. De esta forma, todo el personal técnico o directivo que está relacionado con el proyecto tiene información al día sobre su progreso.
  2. Informar sobre todo el proyecto (documento final).
  3. Facilitar la reutilización del modelo en los casos en los que se prevé un posible interés en su uso futuro.

  Se recomienda recoger en los informes la siguiente información: introducción, objetivos, hipótesis, descripción física del sistema, descripción del modelo, análisis de los experimentos efectuados y conclusiones.

  - *Implementación:* se entiende por implementación el tomar decisiones como consecuencia del estudio de simulación. Un proyecto de simulación no tiene éxito si ha justificado técnica y económicamente una mejora o cambio que no ha sido posteriormente implementado.

  Para que las recomendaciones sean tenidas en cuenta es necesario que el modelo sea creíble. Para ello, no es suficiente que el modelo sea válido. Es también imprescindible que los responsables de la toma de decisiones estén convencidos de su validez.

1.8 Criterios para la evaluación de las herramientas de simulación

La popularidad que ha estado adquiriendo la simulación orientada a eventos discretos en los últimos años se ha visto acompañada de un incremento notable de las herramientas de simulación comerciales. Las prestaciones de éstas mejoran continuamente con el objetivo de facilitar su utilización por parte de usuarios no expertos.

Los mecanismos internos (por ejemplo, la gestión de la cola de eventos o la generación de números aleatorios) son esencialmente los mismos para todas las herramientas. Las principales diferencias entre ellas residen en las funciones y metodología de modelado, y en las interfaces de usuario. La mayoría de las herramientas tienen interfaces gráficas, requieren poca programación y el conjunto de opciones se puede seleccionar a través de menús desplegables. Los mecanismos para la animación están cada vez más integrados con las funciones de modelado. De esta forma se dispone directamente de un nivel mínimo de animación. En algunos casos, los mecanismos de animación pueden ser muy sofisticados.

A menudo, las herramientas de programación son evaluadas sólo atendiendo a dos o tres criterios, tales como precio, facilidad de uso o calidad de las demostraciones comerciales. Aunque todos los entornos han sido desarrollados para cubrir las necesidades de los usuarios finales mediante interfaces amigables y fáciles de utilizar, una mala elección de las herramientas informáticas suele acabar repercutiendo tanto en un incremento considerable de las horas dedicadas a la programación por parte

© Los autores, 2002; © Edicions UPC, 2002.

de personal especialista, como en la necesidad de adquisición de librerías adicionales que no se suministraban con el entorno original.

Al igual que en el campo de la programación no existe el concepto de *mejor lenguaje de programación*, puesto que cada lenguaje ofrece herramientas muy potentes y depuradas, útiles para satisfacer ciertos objetivos y totalmente inservibles para abordar otro tipo de problemas, tampoco existe el concepto *el mejor simulador* en el campo de la simulación de modelos de eventos discretos. Aunque algunos entornos de simulación ofrecen más funciones que otros, no existe ningún paquete que satisfaga de manera óptima los diversos aspectos del desarrollo de un proyecto de simulación en todos los campos de aplicación [BANK99].

A pesar de que cualquier lenguaje de programación podría ser utilizado para simular el comportamiento de un sistema orientado a eventos discretos, existe un conjunto de características inherentes, tanto a los proyectos de simulación, como a la propia evolución de los sistemas a simular, que aconsejan el uso de herramientas específicas del campo de la simulación:

- Los entornos de simulación ofrecen las herramientas necesarias para programar un modelo de simulación. Entre ellas podemos destacar la generación de números aleatorios, la generación de variables aleatorias que responden a ciertas funciones de distribución, la gestión automatizada de los proyectos y escenarios, del tiempo de simulación, de las rutinas de tratamiento de los eventos que determinan el comportamiento del sistema, los métodos de diseño experimental y de análisis de los resultados, y la generación de informes entre otros. Todas estas funcionalidades redundan en una reducción considerable del tiempo necesario para la programación del modelo.
- El mantenimiento de un modelo de simulación se ve enormemente simplificado cuando éste ha sido programado en un entorno especializado de simulación. Nótese que una característica inherente a la metodología de modelado es la constante manipulación del modelo a partir de los resultados obtenidos del simulador.
- Las herramientas de simulación suelen ofrecer funciones específicas para la detección de errores, como es el caso de la aparición de bloqueos en los sistemas de transporte.

1.9 Ejercicios

1. Busca en Internet los servidores con contenidos relevantes en el campo de la simulación de eventos discretos (palabras clave: *discrete event simulation*). Clasifícalos en función de los contenidos.
2. Busca en la librería virtual http://www.amazon.com o similar, los libros publicados de simulación de eventos discretos.
3. Describe las fuentes más importantes de azar (variables aleatorias) en los componentes de los sistemas siguientes:
    - Un restaurante
    - El departamento de urgencias de un hospital
    - Una estación de ferrocarril

© Los autores, 2002; © Edicions UPC, 2002.

- Un supermercado
- Una fábrica de motores

4. Describe cuáles pueden ser los eventos, entidades temporales, actividades y recursos de los sistemas del ejercicio anterior.

5. Proporciona tres ejemplos de sistemas que se puedan modelar como sistemas continuos, tres que se puedan modelar como sistemas discretos y tres más que sean modelables como sistemas orientados a eventos discretos.

6. La taquilla de un teatro está formada por una ventanilla dedicada a la venta de entradas. Los compradores llegan a la taquilla para comprar las entradas. Si ya hay clientes en la taquilla los compradores hacen cola y esperan hasta que la ventanilla queda libre.

    - ¿Qué parámetros debemos conocer para poder modelar el sistema?
    - ¿Qué variables podemos estudiar del sistema?

7. Simula la llegada de clientes al teatro en los minutos 5, 9, 10, 12, 13, 16, 21, 22, 24 y 28, de manera manual, asociándoles el tiempo de servicio en taquilla que se distribuye según la siguiente tabla.

Tiempo de servicio	Probabilidad (%)
1 min.	20 %
2 min.	30 %
3 min.	40 %
4 min.	10 %

Tabla 1.5 Distribución de probabilidad

8. Simula manualmente el modelo del proceso de órdenes introducido en este capitulo para las 10 primeras órdenes. Genera los valores aleatorios necesarios con un ordenador (por ejemplo, empleando Excel) o una calculadora.

9. Es importante garantizar que un modelo de simulación es válido. Para ello es muy importante asegurar que los datos obtenidos de un sistema sean realmente representativos de lo que se desea modelar. Analiza los problemas potenciales de un estudio de simulación en el que se desea modelar la eficiencia de los trabajadores en una línea de fabricación. Para la modelización de este aspecto se observa directamente la eficiencia de una muestra de trabajadores y, a partir de estos datos, se obtiene una distribución representativa.

10. Una oficina de información turística está formada por un grupo de tres trabajadores. A esta oficina llegan pidiendo información una media de 100 personas/hora. De estas personas, un 70 % solamente necesita mapas o folletos informativos, con lo que el tiempo de atención por parte de los informadores es de 1 minuto. El 30 % restante pide información más detallada y se precisa una media de 3 minutos para atenderlos. Encuentra el grado de ocupación de los informadores desarrollando el modelo estático del sistema.

© Los autores, 2002; © Edicions UPC, 2002.

11. El servicio de información turístico del ejercicio anterior ha decidido contratar a estudiantes de turismo para que puedan realizar prácticas. La única diferencia entre los estudiantes y los otros trabajadores es que los primeros sólo trabajan durante 4 horas en lugar de una jornada laboral de 8. La oficina está compuesta por 1 trabajador y un cierto número de estudiantes. Si la jornada laboral de la oficina es de 8 horas/día, ¿cuántos estudiantes se necesitan?

© Los autores, 2002; © Edicions UPC, 2002.

2 Modelado de sistemas orientados a eventos discretos

Son muchos los sistemas o campos de aplicación en los que se desconocen las consecuencias de la ocurrencia de un cierto evento, así como la influencia sobre el rendimiento global del proceso de una variación en la secuencia de eventos que se producirían como consecuencia de un cierto evento situado en algún instante anterior en el tiempo. Considérese por ejemplo la siguiente situación:

> *Cada atardecer, cuando llega el autobús a la fábrica con los trabajadores del turno de noche, puede verse, tan pronto se abren las puertas del vehículo, cómo cada uno de los obreros, bien en solitario bien en grupo, se dirige hacia los distintos departamentos de la línea de producción; cada uno de ellos realiza un conjunto de tareas distintas (marcar hora de llegada, cambiar de ropa en los vestuarios, lectura del parte del turno anterior, reunión con el responsable del departamento, etc.) en diferentes lugares de la fábrica antes de incorporarse a su lugar de trabajo.*

En este caso, un evento (llegada del autobús) desencadena un nuevo conjunto de eventos, los cuales, a su vez, desencadenarán nuevos eventos, y así sucesivamente. Existen una gran diversidad de sistemas (desde sistemas logísticos puros en empresas de servicio, hasta líneas de producción completamente automatizadas) que, como en la situación descrita, deben ser considerados complejos por la dificultad que comporta formalizar las secuencias de actividades que puede desencadenar un determinado evento.

Nótese que la complejidad de un sistema no debería medirse en función del número de componentes -o subsistemas- que lo integran (existen sistemas electrónicos con un considerable número de componentes que se encuentran lejos de ser clasificados como sistemas complejos, por ejemplo, un televisor), ni tampoco en función del número de ecuaciones necesarias para describir su comportamiento. En el ámbito de este tema, la complejidad de un sistema no se entiende como una propiedad inherente al comportamiento del mismo, sino más bien como *una falta de metodología y de herramientas que permitan especificar y formalizar el conocimiento que se tiene del sistema con el objetivo de desarrollar un modelo que presente un comportamiento similar al del sistema real*. El conocimiento de mecanismos que permitan formalizar la dinámica de un proceso, así como una buena base en el uso de las herramientas matemáticas, hace que algunos sistemas, que en el pasado habían sido considerados complejos, ya no lo sean en la actualidad.

En los sistemas logísticos, de fabricación o de servicios, y en aquellos sistemas en los que entidades compiten por acceder a recursos compartidos y cuyo estado puede variar de forma asíncrona, se

© Los autores, 2002; © Edicions UPC, 2002.

requiere una buena metodología para describir el comportamiento del sistema, y que el modelador tenga un buen conocimiento de cómo, cuándo y por qué el estado del sistema sufre un cambio.

En este sentido, puede decirse que hoy en día todavía no existe una metodología reconocida y aceptada por los distintos sectores empresariales, industriales y académicos, que permita formalizar el conocimiento que se tiene de los sistemas logísticos.

En el punto 1.7 del capítulo anterior se han presentado las etapas principales de un proyecto de simulación. Posteriormente, en el mismo apartado se ha justificado la necesidad de obtener un modelo conceptual del proceso de interés previo a las etapas de recogida de datos y de construcción del modelo de simulación. Las ventajas de disponer de un modelo conceptual son:

- La mera especificación textual de un modelo de simulación es a menudo poco precisa y genera varias interpretaciones posibles.
- Los modelos conceptuales recogen de forma detallada y precisa las relaciones dinámicas entre los diferentes elementos del proceso de interés y, por tanto, constituyen en sí mismos una especificación del modelo del proceso que se pretende estudiar.
- El modelo conceptual facilita el diálogo y la coordinación entre todas las partes implicadas en el estudio.
- El modelo conceptual constituye una representación del modelo de simulación independiente de la herramienta de simulación escogida.

Los modelos conceptuales también pueden ser considerados modelos analíticos si pueden ser resueltos analítica o numéricamente. Si ello es posible, siempre es preferible obtener la solución empleando técnicas analíticas, dado que en general su coste de cálculo será menor. No obstante, en la mayoría de los procesos reales la solución analítica es inviable si se pretende construir el modelo con todo el nivel de detalle deseado. En este apartado, el objetivo principal de los modelos conceptuales (analíticos o no) es el de constituir una especificación formal del modelo de simulación.

A pesar de que algunos autores apuntan diversos métodos para reducir la complejidad de los modelos de simulación, dichos métodos suelen reducir el nivel de detalle del modelo (agrupando los eventos que deben ser modelados), o bien comportan una formalización a un nivel de abstracción superior. Aunque los modelos de simulación generados utilizando estas metodologías que simplifican la complejidad de los sistemas garantizan una fácil reutilización de los modelos codificados, existen muchas aplicaciones de la simulación donde no es posible garantizar que dichos modelos respondan a los diversos objetivos de experimentación. En [CHWI00] el lector puede encontrar una clasificación de los diversos motivos por los cuales hay una gran proliferación de modelos de simulación largos y complejos, de entre los cuales la mayor parte están relacionados con una errónea formalización del modelo conceptual.

Las herramientas metodológicas que ofrecen los actuales entornos de simulación comerciales, a pesar de poder ser consideradas eficientes, no son suficientemente potentes como para satisfacer las necesidades de mantenimiento, manipulación y reutilización de ciertos modelos. Esto es especialmente cierto en aquellas áreas de la ingeniería que utilizan la simulación como una

© Los autores, 2002; © Edicions UPC, 2002.

herramienta para mejorar el rendimiento del sistema a partir de aproximaciones del tipo *prueba y error*, que requieren una modificación continua del modelo de simulación.

En nuestra opinión, es necesario un proceso de abstracción que permita formalizar todo el conocimiento que se tiene del comportamiento de un sistema en un modelo conceptual fácil de interpretar y así reducir la complejidad de los modelos de simulación. Dichos modelos deben retener toda la información necesaria para obtener una representación de la dinámica de los sistemas con suficiente precisión para responder con exactitud a los objetivos de la simulación. Por proceso de abstracción se entiende: "Capturar los aspectos esenciales del comportamiento del sistema sin incluir los detalles sobre cómo dicho conocimiento se implementará en un código de simulación" [SIST98, MCGR00].

2.1 Formalización de modelos conceptuales

Desde 1960 hasta 1980, el desarrollo de modelos de simulación orientados a eventos discretos se basaba directamente en el uso de los lenguajes de programación (Fortran, Basic y más tarde C) o de lenguajes especializados de simulación asociados a los tres marcos de trabajo más extendidos: programación de sucesos (Simscript), exploración de actividades (CSL), interacción de procesos (GPSS, Simscript) [NANC99].

A partir de los 80, las limitaciones de dichas metodologías de modelado se hicieron patentes tanto en el mantenimiento como en la reutilización de los modelos desarrollados. El modelado de sistemas cada vez más grandes y complejos potenció la investigación sobre formalismos y metodologías de modelado a un nivel de abstracción superior que los lenguajes de simulación. De hecho puede decirse que fue a partir de los 90 cuando hubo un incremento considerable en el desarrollo de metodologías de modelado que permitieran especificar el modelo conceptual, respetando ciertas propiedades, que habían sido requeridas anteriormente, como [OVER82]:

- el formalismo debe ser independiente de los constructores y herramientas que ofrecen los entornos de simulación,
- el modelo formalizado debe poder ser analizado para determinar relaciones entre componentes y evaluar alternativas que permitan la simplificación del modelo,
- el formalismo debe permitir una fácil transformación a las representaciones soportadas por los entornos de simulación,
- algunos aspectos del modelo pueden dejarse sin especificar, sin que ello dificulte la transformación a otras representaciones,
- el modelo debe poder ser definido en términos que no restrinjan su codificación a un mecanismo particular de actualización del reloj del simulador.

© Los autores, 2002; © Edicions UPC, 2002.

Una propiedad importante que debe soportar la metodología escogida para desarrollar el modelo conceptual de un sistema multicomponente, es la capacidad de modularización, entendiendo como tal la posibilidad de describir el comportamiento de cada uno de los subsistemas que integran el sistema que se desea modelar de forma independiente del resto. Esta característica permite reducir la complejidad de los modelos, y es un prerequisito para minimizar el tiempo de desarrollo del proyecto de simulación en aquellos proyectos de simulación en que intervienen diferentes expertos en la formalización del modelo conceptual o en su codificación.

En la figura 2.1 se ilustra el incremento de canales de comunicación entre desarrolladores de modelos conforme aumenta el número de éstos en el desarrollo de un proyecto de simulación [BABI86].

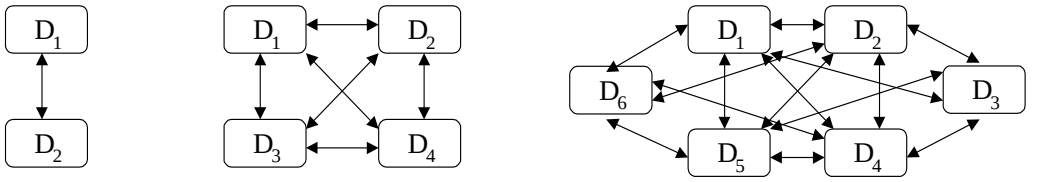


Figura 2.1 Número de canales de comunicación entre desarrolladores

La modularización permite formalizar el modelo conceptual desde una aproximación incremental, lo que facilita mejorar en etapas sucesivas el grado de detalle en cada submodelo, y su verificación y validación, ya que cada etapa sucesiva es finalizada con una fase de implementación.

Nótese que la modularidad hace posible que un cambio en el sistema modelado sea fácilmente trasladable al modelo conceptual a partir de identificar los subsistemas sobre los que se deben reflejar dichos cambios, e identificando los eventos que se ven afectados. Por otro lado, cuando existen relaciones directas entre los diferentes aspectos internos de los módulos, el mantenimiento del modelo se complica, al ser difícil la actualización de un módulo sin afectar a la validez de todo el modelo.

Para garantizar la modularidad es necesario satisfacer las siguientes condiciones [PIDD98]:

- un módulo no debe acceder directamente al estado de ningún otro módulo o componente,
- un módulo debe tener un conjunto de puertos de entrada/salida que le permitan interactuar con el resto del modelo.

De entre los distintos formalismos que se presentarán en este capítulo para describir el modelo conceptual, tanto el formalismo de *redes de Petri* (a partir de ahora RdP) como de *redes de Petri coloreadas* (a partir de ahora RdPC) pueden beneficiarse de todas las ventajas de desarrollo modular cuando la descomposición del sistema se realiza considerando tan sólo los eventos que afectan a su comportamiento y que forman parte de las dinámicas de interés, al asociarse a cada módulo un único evento. La interacción entre módulos aparece como efecto de compartir nodos de tipo lugar, el conjunto de los cuales determina el estado del sistema.

En el resto del presente capítulo se presenta, en primer lugar, un breve resumen del formalismo de modelo de flujos y del de redes de colas. A continuación, se introducirá la formalización de modelos

© Los autores, 2002; © Edicions UPC, 2002.

mediante RdP y RdPC, al considerar los autores que son ambos idóneos para representar tanto los cambios de estado como los eventos y sus relaciones, pues facilitan la correcta codificación del modelo conceptual en un entorno de simulación, y la validación y verificación del modelo. Cabe destacar que los lenguajes de modelado orientados a estados y transiciones, entre los que se incluyen las redes de Petri, han demostrado ser muy útiles para analizar la dinámica de los sistemas, así como la propuesta de alternativas para mejorar el rendimiento de los mismos.

2.2 Modelos de flujos

Los modelos de flujos representan gráficamente el conjunto de actividades que forman una tarea. Su ventaja principal es su simplicidad [HARR91]. En un modelo de flujos, la siguiente información debe ser especificada en cada actividad [ORTN99]:

- Precondiciones y postcondiciones: qué condiciones deben cumplirse antes del inicio de una actividad, y al finalizar ésta.
- Quién tiene control sobre la actividad.
- Cuáles son las otras actividades requeridas para completar una cierta actividad
- La entrada/salida de la actividad: los datos y el flujo de información necesarios para la realización de una cierta tarea.

La figura 2.2 muestra el diagrama de flujo simplificado de pacientes infantiles en el servicio de urgencias pediátricas de un hospital. Los pacientes que llegan al servicio son, en primer lugar, registrados en la administración del servicio. A continuación se efectúa un chequeo rápido inicial para determinar si el estado del paciente precisa su traslado directo al departamento de emergencias. Esta situación es poco frecuente y es debida a que los padres del niño no son conscientes de la gravedad de su estado dado que, en caso contrario, lo hubiesen trasladado directamente a emergencias en lugar del servicio de urgencias.

El resto de pacientes son atendidos, por orden de llegada, en los *boxes* dispuestos para tal efecto. Como resultado del chequeo el médico puede recomendar rayos X o analíticas complementarias. En estos casos, el paciente tiene que ser atendido por segunda vez en uno de los *boxes* para confirmar el diagnóstico. Finalmente, el paciente recibe el alta del servicio de pediatría.

El principal problema de los diagramas de flujo es que no contemplan de forma explícita el conjunto de recursos necesarios para el correcto funcionamiento del proceso. En el ejemplo anterior, los recursos más restrictivos son el número de médicos trabajando, el número de *boxes* y el número de salas de rayos X. Dado que muy a menudo uno de los objetivos fundamentales de un estudio de simulación es la determinación del conjunto de recursos necesarios, se cree necesario emplear otra metodología de modelado que contemple el conjunto de actividades, entidades temporales, eventos y recursos que forman un proceso.

© Los autores, 2002; © Edicions UPC, 2002.

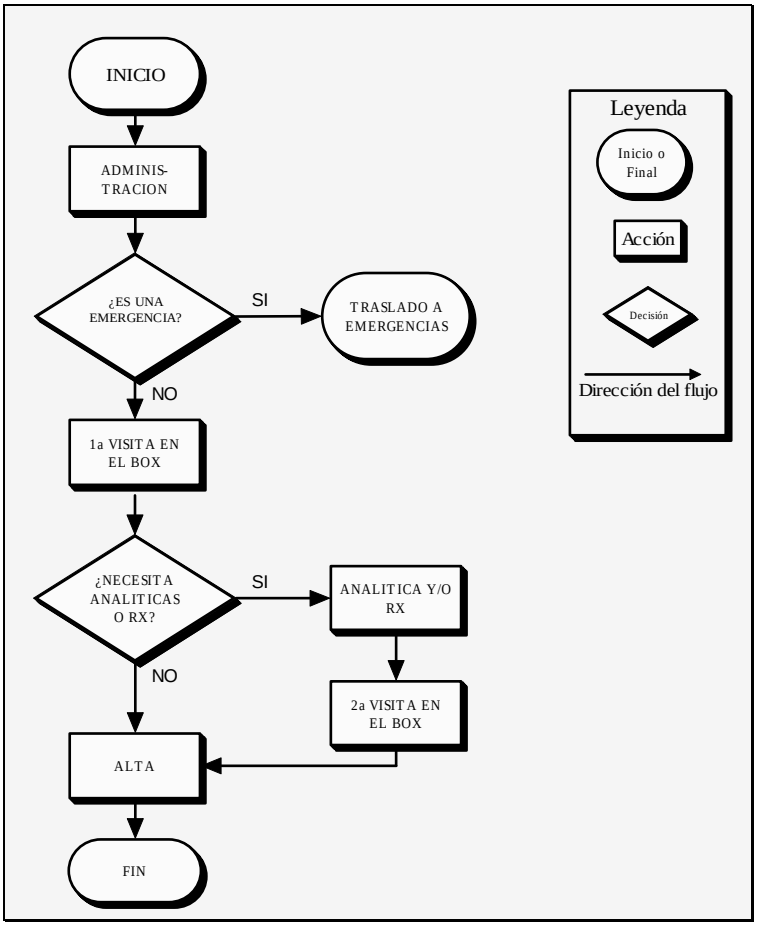


Figura 2.2 Diagrama de flujo del servicio de urgencias pediátricas

2.3 Redes de colas

La especificación de un sistema orientado a eventos discretos según el formalismo de redes de colas consiste en representar los distintos servidores donde se llevarán a cabo las actividades que se desarrollan en el sistema mediante un conjunto de colas (o estaciones de trabajo –figura 2.3-) interconectadas.

Los elementos básicos que forman un proceso de colas son:

- Las *entidades* temporales que compiten entre sí para capturar los recursos.
- Los *recursos* (generalmente limitados) que proporcionan un servicio a las entidades temporales.
- El espacio de espera de las entidades temporales cuando el recurso al que desean acceder está ocupado. Este espacio se denomina *cola*.

© Los autores, 2002; © Edicions UPC, 2002.

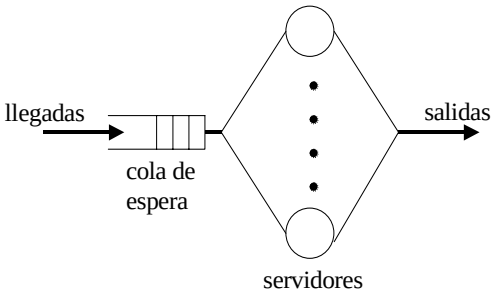


Figura 2.3 Representación de una estación de trabajo mediante redes de colas

Cada estación de trabajo viene descrita por un conjunto de servidores (una estación puede tener uno o más servidores representados por círculos) que atienden las peticiones de trabajo que llegan a la estación.

Ante la llegada de una petición de trabajo a una estación, ésta es atendida inmediatamente (se inicia una actividad) si hay algún servidor disponible, en caso contrario la petición se almacena en una cola a la espera de que un servidor quede libre. Una alternativa a dicho proceso aparece cuando la unidad de servicio tiene implementada una política *preemtive* por la cual, si no hay ningún servidor libre pero la petición de trabajo tiene mayor prioridad que alguno de los trabajos que están siendo procesados por alguno de los servidores de la estación, la nueva petición es atendida, y la actividad menos prioritaria para y es almacenada en la cola a la espera de un servidor libre.

El flujo de actividades que caracterizan un proceso productivo puede formalizarse mediante la notación de colas interconectando las distintas estaciones que describirán las tareas que son realizadas en el sistema. En el caso que la secuencia de actividades pueda ser realizada por diferentes secuencias de estaciones, se ha dotado al formalismo de redes de colas con un conjunto de elementos pasivos (figura 2.4) que permiten especificar elementos característicos de los sistemas orientados a eventos discretos.

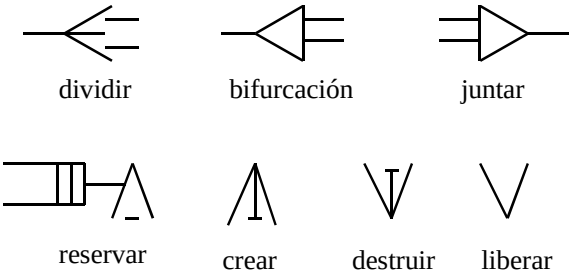


Figura 2.4 Primitivas de modelado en redes de colas

La teoría de colas ofrece una metodología analítica para evaluar el rendimiento de un sistema orientado a eventos discretos descrito en notación de colas (análisis cuantitativo), en la que las estaciones vienen descritas, según la notación de Kendall, por una expresión con un mínimo de tres símbolos (apéndice A):

A/B/m

© Los autores, 2002; © Edicions UPC, 2002.

donde:

- A: indica la distribución estadística que describe el proceso de llegadas,
- B: indica la distribución estadística que describe el tiempo requerido por cada petición de trabajo,
- m: indica el número de servidores de la estación de trabajo.

Un sistema *M/M/s* sería, por ejemplo, aquel en la que los tiempos entre llegadas y los tiempos de servicio siguen una distribución exponencial y en la que el número previsto de unidades de servicio fuera *s*.

Las distintas secuencias de estaciones en las que pueden ser realizadas las actividades se especifican indicando la probabilidad de que una petición de trabajo sea atendida en una estación o en otra en los nodos de bifurcación.

A pesar del amplio uso de la teoría de colas para evaluar y mejorar el rendimiento de sistemas, el formalismo de redes de colas no es suficientemente potente como para modelar las actividades concurrentes en todas sus variantes [VERN87, BAUS93].

Nótese que las redes de colas permiten formalizar de modo muy natural clientes o peticiones, recursos y procesos presentes en la mayor parte de los sistemas logísticos de producción, de servicios o de transporte. No permiten, sin embargo, describir situaciones con conflictos debido a la compartición de recursos, en la que un evento puede comportar la interrupción de ciertas actividades en función del estado del sistema.

2.4 Redes de Petri

Los modelos de eventos discretos se basan fundamentalmente en los conceptos de evento y actividad. Un evento genera un cambio en las variables de estado y una actividad encapsula lo que sucede entre dos eventos. Las redes de Petri (RdP) [PETRI66] permiten representar de forma natural un modelo de eventos discretos. En una RdP los eventos están asociados a transiciones y las actividades a lugares. Aunque no es el único formalismo que maneja eventos y actividades, es el único que representa formalmente el paralelismo y la sincronización [SILV89]. Otros aspectos que contribuyen a potenciar las RdP para el modelado de sistemas de eventos discretos son:

- Es posible efectuar un análisis cuantitativo del sistema, estudiando aspectos estructurales tales como las situaciones de bloqueo o la alcanzabilidad de ciertos estados.
- La RdP permite determinar de modo inmediato todos aquellos eventos que pueden aparecer cuando el sistema se encuentra en un cierto estado y todos los eventos que pueden ser desencadenados por la aparición de un evento determinado.
- La RdP permite formalizar un sistema a distintos niveles de abstracción de acuerdo con los objetivos de modelado.
- Permite la descripción de un sistema complejo mediante la metodología *Bottom-Up*: construcción del modelo del sistema completo a partir de las RdP (submodelos) de los subsistemas que han sido previamente desarrollados y verificados.

© Los autores, 2002; © Edicions UPC, 2002.

- Es un formalismo de modelado gráfico con muy pocas reglas sintácticas.

Este conjunto de características son de interés en las distintas etapas de desarrollo del modelo de simulación de un sistema: desde la fase inicial de identificar los objetivos para los cuales se va a desarrollar el modelo hasta la fase final de validación.

2.4.1 Definiciones

Una RdP es un caso particular de grafo dirigido, ponderado y bipartito, con 2 tipos de nodos: *lugares y transiciones*, donde los arcos pueden conectar los nodos lugar a los nodos transición o al revés.

Matemáticamente, *una RdP puede definirse a partir de la siguiente tupla de 5 elementos*:

$$RdP = (P,T,A,W,M_0)$$

donde el significado de cada una de las siglas utilizadas es el siguiente:

- $P = \{P_1,P_2,P_3,.....,P_n\}$: conjunto de nodos tipo lugar
- $T = \{T_1,T_2,T_3,.....,T_m\}$: conjunto de nodos tipo transición
- $A \subset (P \times T) \cup (T \times P)$: conjunto de arcos de la RdP = subconjunto del producto cartesiano de todos los nodos P y T
- $W = A \mapsto \{1,2,3,.....\} \forall A_i$: peso asociado a cada arco
- $M_0 = P_i \mapsto \{1,2,3,....\} \forall P_i$: número de marcas iniciales en cada nodo tipo lugar (marcado inicial)

El estado del sistema queda totalmente determinado por el número de marcas en cada nodo tipo lugar, y puede describirse matemáticamente por el vector P.

Para representar y analizar la RdP es conveniente definir la función $E(T_j)$ que representa el conjunto de lugares de entrada a la transición T_j. De forma similar, se puede definir la función $S(T_j)$ para indicar los lugares de salida de la transición T_j:

$$E(T_j) = \{P_i \in P, (P_i, T_j) \in A\}$$
$$S(T_j) = \{P_i \in P, (T_j, P_i) \in A\}$$

Los nodos tipo *transición* se representan gráficamente por rectángulos, y se utilizan para modelar los eventos que aparecen en la dinámica del sistema. Los nodos tipo *lugar* se representan gráficamente por círculos y se utilizan generalmente para describir tanto las colas de espera del sistema (pulmones, *stocks*, etc.), como las condiciones sobre el estado en que se encuentran los elementos o recursos que integran el sistema. El número de elementos (piezas, personas, recursos, etc.) que hay en una cola de espera se representa gráficamente por marcas (*tokens*) en el interior del nodo lugar, y el estado de una condición (verdadero o falso) se representa gráficamente con una marca (*token*) en el interior del nodo lugar, en el caso de que la condición se cumpla.

Los arcos que conectan los nodos lugar con los nodos transición suelen tener un peso asociado, el cual permite describir las condiciones necesarias para que el evento representado en la transición pueda activarse.

© Los autores, 2002; © Edicions UPC, 2002.

Las reglas para simular el comportamiento dinámico de una RdP son:

- Una transición T_i está activada si cada uno de los nodos P_j tipo lugar conectados a la entrada contienen al menos $W(P_j,T_i)$ marcas. $W(P_j,T_i)$ representa el peso del arco que une el nodo P_j con la transición T_i (si en un arco no aparece su peso, se supone que es 1). De forma resumida, se dice que una transición T_i está activada si

$$M(P_j) \geq W(P_j,T_i) \quad \text{para todo} \quad P_j \in E(T_i)$$

  donde $M(P_j)$ es el número de marcas en el nodo tipo lugar P_j.

- Una transición activada puede dispararse en cualquier momento.

- Como resultado de disparar una transición activada, se eliminan $W(P_j,T_i)$ marcas de cada nodo P_j a la entrada de T_i , y se añaden $W(T_i ,P_k)$ marcas a cada nodo P_k de salida de T_i.

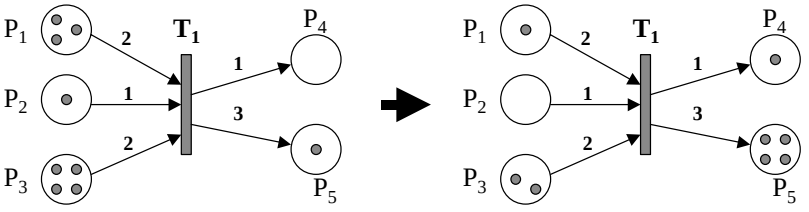


Figura 2.5 Resultado de disparar la transición activada T_1

La figura 2.5 ilustra mediante un ejemplo las reglas anteriores, donde puede visualizarse en la parte izquierda una transición activada (los nodos de entrada a la transición contienen como mínimo tantas marcas como el peso de los arcos que los conecta con la transición: M(P1)=3 ≥ 2, M(P2)=1 ≥ 1 y M(P3) =4 ≥ 2). En la parte derecha de la misma figura se ha representado el estado de la misma RdP una vez se ha disparado la transición (se han eliminado 2 marcas de los nodos P1 y P3, y una marca del nodo P2, y se ha añadido 1 marca sobre el nodo P 4, y 3 marcas sobre el nodo P5).

Una de las ventajas más importantes de las RdP es que permiten representar de forma elegante los mecanismos propios de los procesos orientados a eventos discretos:

- *Ejecución secuencial*: refleja la dependencia causal entre eventos. En la figura 2.6a se muestra que la transición T_2 sólo puede ser disparada después de activarse la transición T_1.

- *Toma de decisiones*: las transiciones T_1, T_2 y T_3 de la figura 2.6b están todas habilitadas para su disparo. No obstante, el disparo de cualquiera de ellas supone la deshabilitación del resto. Esta situación aparece, por ejemplo, cuando una pieza puede escoger una máquina entre varias disponibles. Estos conflictos pueden ser resueltos de forma determinista o probabilística.

- *Concurrencia*: las transiciones T_1, T_2 y T_3 de la figura 2.6c son concurrentes. La concurrencia es uno de los conceptos más importantes que encontramos en los procesos dinámicos de interés.

- *Sincronización*: en la figura 2.6d se muestra que la transición T_1 no se puede disparar hasta que no se tenga una marca en el lugar P_1. Esta estructura se emplea cuando se está a la espera de un

© Los autores, 2002; © Edicions UPC, 2002.

recurso para proseguir (por ejemplo, se está a la espera de un transportador libre o de una máquina libre), o bien en procesos de ensamblado donde se requieren todas las piezas para realizar una operación.

- *Agrupación*: en la figura 2.6e las marcas que llegan por diferentes caminos (transiciones T_1, T_2, T_2) se agrupan en el lugar P_1. Esta situación aparece, por ejemplo, cuando pacientes que han seguido caminos diferentes dentro de un hospital son visitados por el mismo médico.

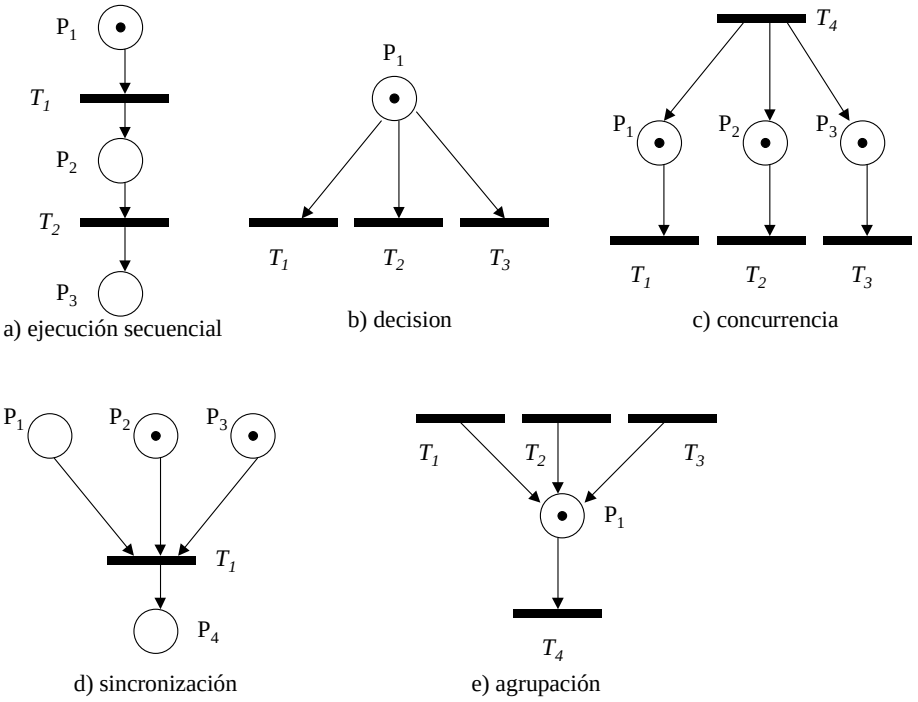


Figura 2.6 RdP de unidades sencillas

EJEMPLO 2.1 Modelo de un recurso limitado

La figura 2.7 representa un proceso simple, compuesto básicamente por una estación de trabajo que consta de dos máquinas. Las piezas que llegan para ser procesadas se esperan en una única cola FIFO (*first-in-first-out*) para ser procesadas. Finalmente, abandonan el proceso.

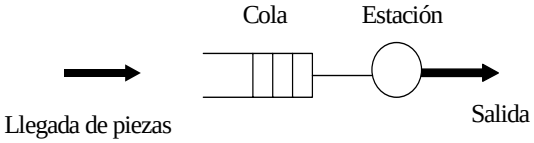


Figura 2.7 Estación de trabajo

La figura 2.8 muestra el modelo en RdP del proceso anterior, la cual queda especificada por:

© Los autores, 2002; © Edicions UPC, 2002.

- $P = \{P_1,P_2,P_3 ,P_4\}$
- $T = \{T_1,T_2,T_3 \}$
- $A =\{(T_1,P_1),(T_2,P_3),(T_3,P_2),(T_3,P_4),(P_1,T_2),(P_2,T_2),(P_3,T_3)\}$
- $W(T_1,P_1)=1 ,W(T_2,P_3)=1, ..., W(P_3,T_3)=1$
- $M_0 = \{0,2,0,0\}$

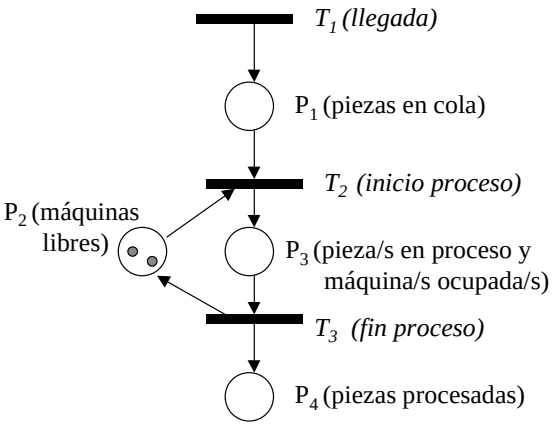


Figura 2.8 RdP de la estación de trabajo

Los aspectos de interés son:

- El disparo de la transición T_1, que modela el proceso de llegadas, no está condicionado.
- La transición T_2 sólo se puede disparar si hay una o dos máquinas libres y si hay una o más piezas en la cola (sincronización).

Las RdP no contemplan de forma explícita el tiempo consumido por los elementos que recorren la red. En este ejemplo, y en la mayoría de los ejemplos posteriores, se supone que el tiempo se consume en los *lugares*, ya sea en la cola a la espera de que un recurso esté libre (lugar P_1) o cuando la pieza está procesándose en la máquina (lugar P_2). En estos casos, las transiciones representan estrictamente los eventos asociados a las llegadas de entidades (el inicio y el final de actividades) y, consecuentemente, no ocupan tiempo.

En algunos casos, y con el objetivo de reducir el número de elementos de la RdP, asociamos a la transición los eventos de inicio y final de una actividad, así como el tiempo de duración de ésta. En la RdP de la figura 2.9 se muestra que la transición T_2 representa el inicio, proceso y fin de proceso. Por tanto, el tiempo de proceso está asociado a la transición T_2.

© Los autores, 2002; © Edicions UPC, 2002.

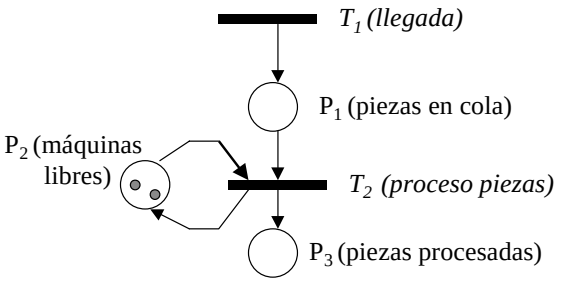


Figura 2.9 RdP de la estación de trabajo

EJEMPLO 2.2 Proceso robotizado simple

Este ejemplo muestra la codificación de un proceso robotizado simple que se caracteriza por que dos flujos diferentes de fabricación compiten por un mismo recurso, dado que el robot tiene que transportar las piezas tipo 1 y las piezas tipo 2.

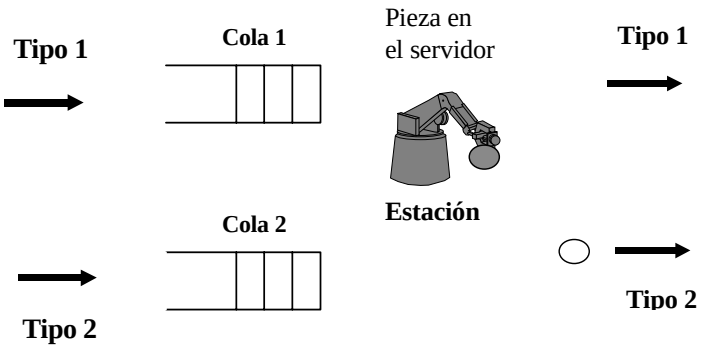


Figura 2.10 Ejemplo de proceso robotizado simple

La figura 2.11 muestra la RdP del proceso anterior. Obsérvese que el objeto que está en la posición P_1, y que indica que el robot está libre, puede emplearse para activar la transición T_3 o la T_4. Por tanto, en el caso que los lugares P_2 y P_3 tengan simultáneamente más de una pieza (marca) cada una, las dos marcas (piezas) compiten entre sí para capturar el recurso restringido *robot* representado por una marca en el lugar P_1 cuando éste está libre (conflicto).

© Los autores, 2002; © Edicions UPC, 2002.

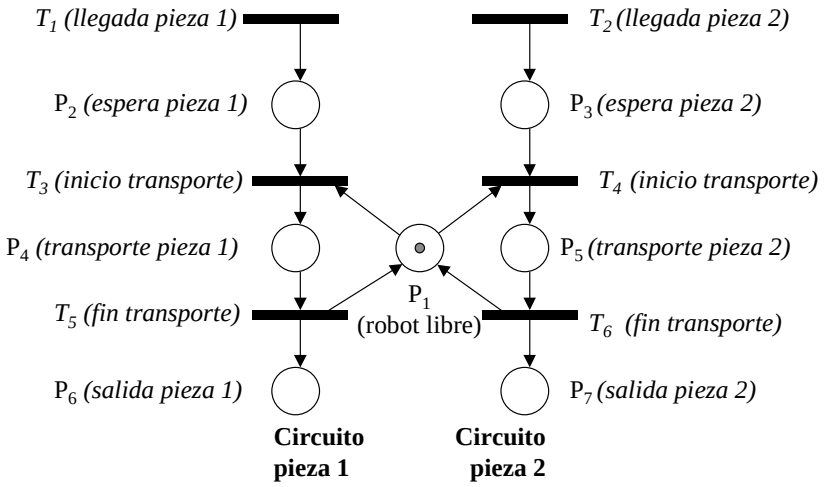


Figura 2.11 RdP del proceso robotizado simple.

EJEMPLO 2.3 Modelo de una máquina ensambladora

En la figura 2.12 se ha representado un sistema integrado por una máquina de ensamblar M1, y dos unidades de almacenamiento (piezas de tipo a y piezas de tipo b). La condición para que se pueda realizar la operación de ensamblado es que, como mínimo, haya una pieza de tipo a, y una de tipo b en las colas de espera de la máquina ensambladora. Nótese que no hay ningún proceso de llegadas de piezas de tipo b. Éstas son cargadas inicialmente por un operador en un almacén interno de la máquina (este sistema presenta características muy similares a las máquinas de bebidas, en las cuales las piezas de tipo b son las latas almacenadas en un *stock* interno, y las piezas de tipo a son los usuarios). Dada esta unidad de producción, se desea desarrollar el modelo que describa la evolución del almacén de piezas de tipo a (por ejemplo, número de piezas a en la cola de espera). Los eventos que afectan al estado del sistema son, tanto el proceso de llegadas de piezas de tipo a, como el proceso de ensamblado. Considérese como condición inicial que en el almacén de piezas de tipo b hay 6 piezas.

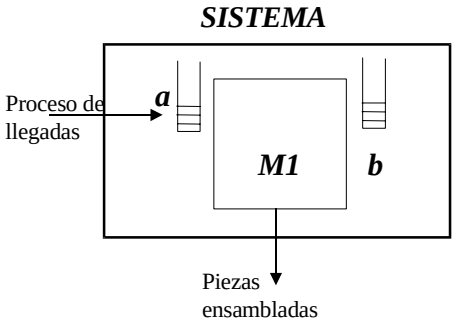


Figura 2.12 Máquina de ensamblar

© Los autores, 2002; © Edicions UPC, 2002.

En la figura 2.13a se ha representado la RdP que modela el sistema de la máquina ensambladora, donde:

- P_1: número de piezas de tipo b en el *stock* interno de la máquina;
- P_2: número de piezas de tipo a en el almacén de entrada;
- T_0 (proceso de llegadas): incrementa en 1 el número de piezas de tipo a en el sistema y no existe ninguna precondición que deba cumplirse para que pueda aparecer un evento de llegada. Nótese que la transición T_0 siempre se encuentra activada (no hay ningún lugar conectado a la entrada de la transición) y cada vez que se dispara se añade un objeto en el nodo P_2;
- T_1 (proceso de ensamblado): Se decrementa en 1 el número de piezas tipo a y tipo b. La condición para que pueda iniciarse el proceso de ensamblado es que, como mínimo, se disponga de una pieza de cada tipo en los almacenes asociados a la máquina. Esta condición se modela mediante los arcos de entrada a la transición T_1. Cada vez que se dispara esta transición se elimina 1 objeto del nodo P_1 y del nodo P_2 .

En la figura 2.13b, se ha representado el modelo del mismo sistema pero considerando que existe una limitación física de capacidad en el almacén de piezas de tipo a, que lo limita a un máximo de 3 piezas. Para modelar esta nueva condición se ha introducido un nuevo nodo P3 a la RdP, que limita el proceso de llegada de piezas (transición T_0).

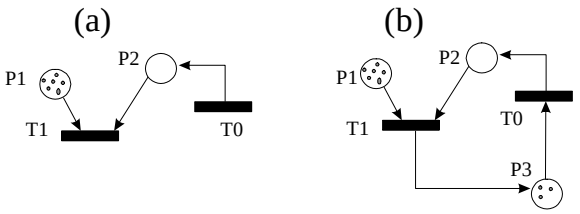


Figura 2.13 RdP de un proceso de llegadas /salidas

2.4.2 Programación *Bottom-Up*

Una de las propiedades que ofrece el formalismo de las redes de Petri como herramienta de modelado es la capacidad de describir modularmente los subsistemas en los que puede descomponerse un sistema, y construir el modelo de todo el sistema integrando cada uno de los submodelos (subredes de Petri).

EJEMPLO 2.4 Línea de producción

Para ilustrar la capacidad de programación *Bottom-Up* de las redes de Petri, considérese la línea de producción descrita en la figura 2.14, formada por las siguientes unidades de producción:

- dos robots: R_1 y R_2;
- tres máquinas CNC: M_1 , M_2 y M_3;
- dos unidades de almacenamiento: S_1 y S_2.

© Los autores, 2002; © Edicions UPC, 2002.

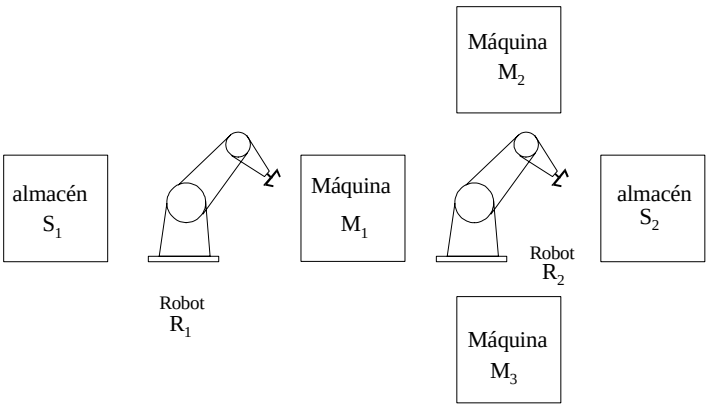


Figura 2.14 Línea flexible de fabricación

El funcionamiento de la línea de producción es el siguiente:

1. El robot R_1 realiza el transporte de las piezas almacenadas en S_1 y las carga en la máquina M_1.
2. La máquina M_1 realiza la primera operación sobre las piezas.
3. Una vez la máquina M_1 ha finalizado la primera operación, el robot R_2 transporta la pieza procesada en M_1 y la carga en la máquina M_2 o en la máquina M_3.
4. Una vez la máquina M_2 o la máquina M_3 ha finalizado la segunda operación, el robot R_2 transporta la pieza finalizada al almacén de salida S_2.

Con el objetivo de facilitar el desarrollo del modelo de la línea de producción, ésta se separará en tres subsistemas de complejidad inferior, para cada uno de los cuales se desarrollará una subred de Petri. Una vez se hayan desarrollado por separado cada uno de los submodelos, se utilizará el método de fusión de nodos tipo lugar de la metodología de desarrollo *Bottom-Up*, para construir el modelo en RdP de toda la línea de producción.

Subsistema 1: está integrado por el almacén de entrada S_1, el robot R_1, y la máquina M_1. Funcionalmente, en este primer subsistema deben describirse las operaciones de transporte de las piezas desde el almacén hasta la máquina M_1, y su procesado. Así pues, debe construirse la RdP que describa el estado del robot R_1, de la máquina M_1, y del almacén S_1, teniendo en cuenta que la condición para que el robot pueda transportar una pieza a la máquina es que: tanto ésta como el robot deben encontrarse libres, y debe haber piezas en el almacén.

En la figura 2.15 se ha representado la RdP de este primer subsistema considerando que hay 6 piezas inicialmente en el almacén S1, donde los nodos lugar tienen el siguiente significado:

- P1: máquina M_1 libre;
- P2: robot R_1 transportando pieza de S_1 a M_1;
- P3: robot R_1 libre;
- P4: máquina M_1 procesando;
- P5: máquina M_1 ocupada con una pieza ya procesada;
- P6: almacén S_1: número de piezas en el *stock*.

© Los autores, 2002; © Edicions UPC, 2002.

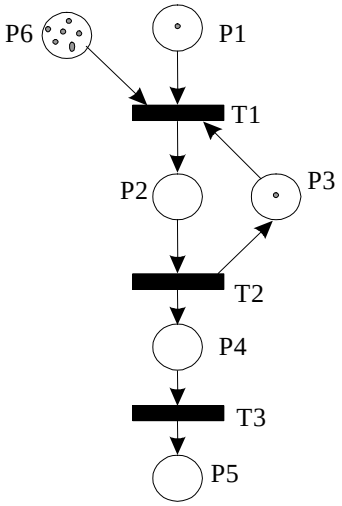


Figura 2.15 RdP del subsistema 1

Subsistema 2: está integrado por el almacén de salida S_2, el robot R_2 y la máquina M_2. Funcionalmente, en este segundo subsistema deben describirse las operaciones de transporte de las piezas desde la máquina M_1 hasta la máquina M_2, el procesamiento de la pieza en la máquina M_2, así como su transporte al almacén de salida S_2 por el robot R_2. Así pues, debe construirse la RdP que describa el estado del robot R_2, de las máquinas M_2 y M_1, y del almacén S_2, teniendo en cuenta, que la condición para que el robot pueda transportar una pieza de la máquina M_1 a la máquina M_2, la primera debe encontrase en el estado de "ocupada con pieza ya procesada", la segunda libre, y el robot R_2 libre. En cuanto a la operación de transporte de una pieza desde la máquina M_2 hasta el almacén de salida, las condiciones son que el robot R_2 se encuentre libre, y que la máquina M_2 se encuentre en el estado de "ocupada con pieza ya procesada".

En la figura 2.16 se ha representado la RdP del segundo subsistema, donde los nodos lugar tienen el siguiente significado:

- P7: máquina M_2 libre;
- P8: robot R_2 libre;
- P9: robot R_2 transportando una pieza de M_1 a M_2;
- P10: máquina M_2 procesando;
- P11: máquina M_2 ocupada con pieza ya procesada;
- P12: robot R_2 transportando una pieza de M_2 a S_2.

© Los autores, 2002; © Edicions UPC, 2002.

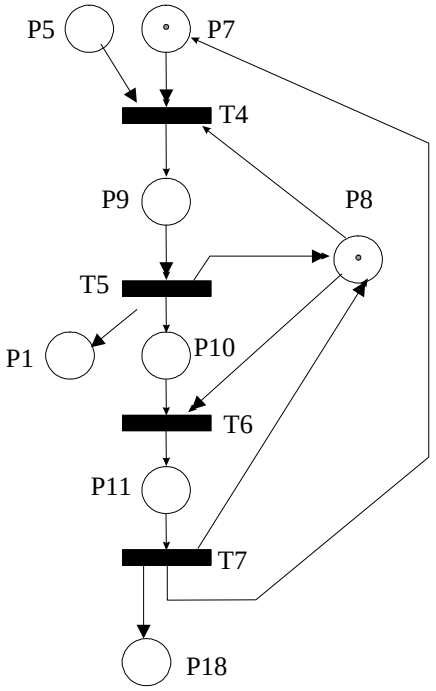


Figura 2.16 RdP del subsistema 2

Subsistema 3: está integrado por el almacén de salida, el robot R_2, y la máquina M_3. Funcionalmente, en este subsistema deben describirse las operaciones de transporte de las piezas desde la máquina M_1 hasta la máquina M_3, el procesamiento de la pieza en la máquina M_3, así como su transporte al almacén de salida S_2 por el robot R_2. Así pues, debe construirse la RdP que describa el estado del robot R_2, de las máquinas M_3 y M_1, y del almacén S_2, teniendo en cuenta la condición para que el robot pueda transportar una pieza desde la máquina M_1 a la máquina M_3: la primera debe encontrarse en el estado "ocupada con pieza ya procesada", la segunda libre, y el robot R_2 libre. En cuanto a la operación de transporte de una pieza desde la máquina M_3 hasta el almacén de salida, las condiciones son que el robot R_2 se encuentre libre, y que la máquina M_3 se encuentre en el estado de "ocupada con pieza ya procesada".

En la figura 2.17 se ha representado la RdP del tercer subsistema, donde los nodos lugar tienen el siguiente significado:

- P13: máquina M_3 libre;
- P14: robot R_2 transportando una pieza de M_1 a M_3;
- P15: máquina M_3 procesando;
- P16: máquina M_3 ocupada con pieza ya procesada;
- P17: robot R_2 transportando una pieza de M_3 a S_2.

© Los autores, 2002; © Edicions UPC, 2002.

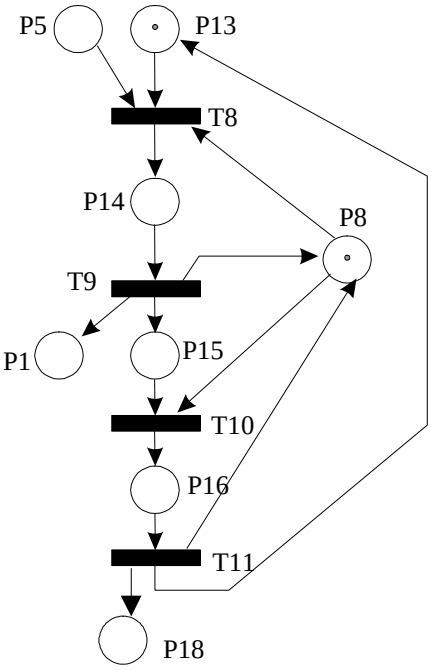


Figura 2.17 RdP del subsistema 3

Una vez se han construido los modelos de los subsistemas en los cuales se han agrupado las dinámicas de la línea de producción, ya es posible construir el modelo de todo el sistema a partir de modelar la interacción entre los subsistemas. En el formalismo de las RdP, la interacción entre subsistemas se describe simplemente identificando aquellos nodos lugar que hacen referencia a una misma condición o que tienen el mismo significado físico, y fusionándolos en un único nodo[1].

En la figura 2.18 se ha representado la RdP de todo el sistema utilizando la metodología de programación *Bottom-Up*, la cual consiste en unificar los nodos lugar que representan la misma condición o cola, a partir de les redes de Petri representadas en las figura 2.15, figura 2.16 y figura 2.17. Estas describen cada uno de los subsistemas en los que se ha descompuesto la línea de producción. Nótese, por ejemplo, que tanto el subsistema 2 como el subsistema 3 comparten dos nodos lugar con el mismo significado físico (el nodo P_8 representa el estado libre del robot R_2, y el nodo lugar P_5 representa máquina M_1 ocupada con pieza procesada), los cuales han sido fusionados en un mismo nodo. De modo análogo debería ocurrir con el nodo lugar P_1, el cual tiene el mismo significado físico en los 3 subsistemas (máquina M_1 libre), pero a fin de facilitar la lectura de la RdP (evitar arcos cruzando otros arcos y nodos), se ha optado por triplicar el nodo P_1, aunque funcionalmente debe ser considerado como un único nodo, de tal forma que al disparar alguna de las

[1] Para facilitar la lectura de la RdP resultante, se permite la redundancia de un mismo nodo lugar en diferentes partes de la RdP, pero teniendo siempre en cuenta que los nodos redundantes deben actualizarse todos al mismo valor cada vez que se dispare un transición que fuerce a un cambio en el número de elementos del nodo lugar.

© Los autores, 2002; © Edicions UPC, 2002.

transiciones T_5 o T_{10}, se añada un objeto en los 3 nodos lugar P_1, y al disparar la transición T_1, se elimine un objeto de los 3 nodos lugar P_1.

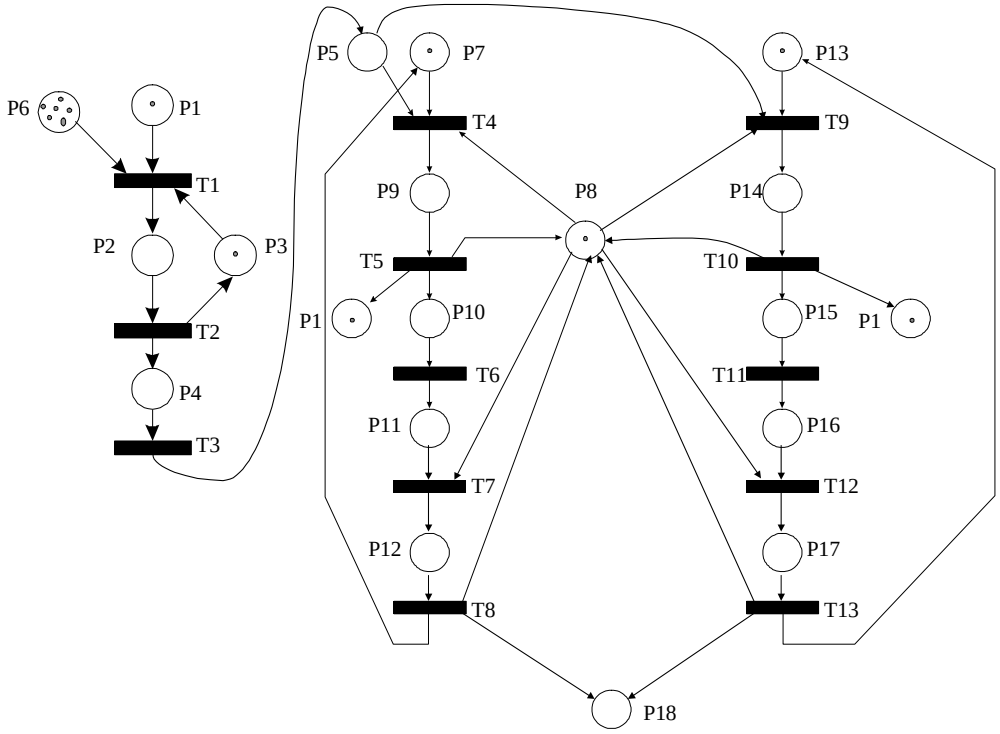


Figura 2.18 RdP de la línea de producción

2.4.3 Análisis de las redes de Petri

En este apartado se introducen brevemente algunas técnicas analíticas de las RdP con el objetivo final de que sirvan de ayuda, tanto en el análisis del modelo, como para la verificación, validación y análisis de los resultados de la simulación. El objetivo principal del análisis es determinar propiedades del modelo ligadas a su comportamiento dinámico:

- Determinación de bloqueos en el sistema.
- Encontrar los posibles caminos para alcanzar un estado final partiendo de un estado inicial, y obtener el coste de cada uno de los caminos.
- Obtener el conjunto de estados posibles a los que se puede llegar a partir de un estado inicial.

Árbol de alcance

Dada una RdP N=(P,T,A,W,M_0) se desea determinar si existe una secuencia σ de transiciones que permitan alcanzar un estado M a partir del estado inicial M_0:

$$M_0 \overset{\sigma}{\longmapsto} M$$

© Los autores, 2002; © Edicions UPC, 2002.

El problema de alcance consiste en determinar el conjunto de estados M a los que se puede acceder a partir del estado inicial M_0. Este conjunto de estados se determinan con la función de alcanzabilidad (*Reachability*), $R(M_0)$. Una forma de determinar el conjunto de estados que se pueden alcanzar consiste en construir el árbol de alcance. Las reglas de construcción del árbol de alcance son:

1. La raíz del árbol es el estado inicial del sistema (vector M_0).
2. Para cada uno de los nodos del árbol se generan tantos nodos hijos como transiciones hay activadas. Los nodos hijos representan el estado de la RdP una vez se han disparado cada una de las transiciones activadas. Los arcos que conectan los nodos padres con los nodos hijos indican gráficamente la transición que se ha disparado.
3. Cuando se genera un nodo hijo que ya existe en algún otro nivel del árbol anterior o igual al actual, éste se marca como nodo *old* y no se generan nuevos nodos hijos a partir de él.
4. Un nodo del árbol que no tiene ninguna transición activada se marca como *dead end*.
5. Un nodo que no esta marcado como *old* o como *dead end* es un nodo *new*.

En los sistemas complejos y altamente acoplados, la detección de errores y seguimiento de las trayectorias es sumamente difícil si no se sigue una metodología adecuada. La información que se obtiene de un árbol de alcance es muy útil en la fase de verificación y validación del modelo de simulación, y también para entender la trayectoria del sistema a partir de un estado cualquiera.

EJEMPLO 2.5 Construcción de un árbol de alcance

Para mostrar cómo se construye un árbol de alcance estudiemos la RdP de la figura 2.19.

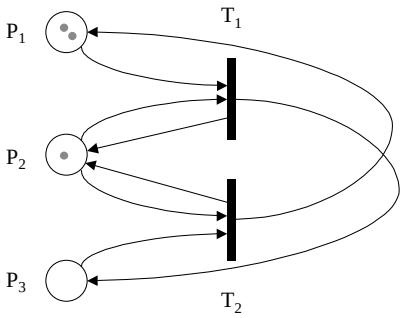


Figura 2.19 RdP de un proceso de transporte

El primer nivel del árbol de cobertura (figura 2.20) es el estado inicial M_0, el cual se deduce directamente a partir de la RdP de la figura anterior.

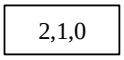


Figura 2.20 Primer nivel del árbol de alcance

A partir del estado inicial sólo está activada la transición T_1 ya que el nodo P_1 contiene dos marcas y el nodo P_2 una. La transición T_2 no está activada, pues en el nodo lugar P_3 no hay ninguna marca. Para

© Los autores, 2002; © Edicions UPC, 2002.

generar el segundo nivel del árbol mostramos cuál sería el estado del sistema después de disparar T_1 (figura 2.21).

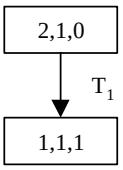


Figura 2.21 Segundo nivel del árbol de alcance

Para construir el tercer nivel (figura 2.22) observamos el estado resultante de disparar T_1 a partir de M_0. En este estado vemos que las dos transiciones T_1 y T_2 están activadas ya que los tres nodos lugar contienen una marca. Así pues, para este tercer nivel buscaremos cuál es el estado resultante del sistema cuando disparamos T_1 y cuando disparamos T_2.

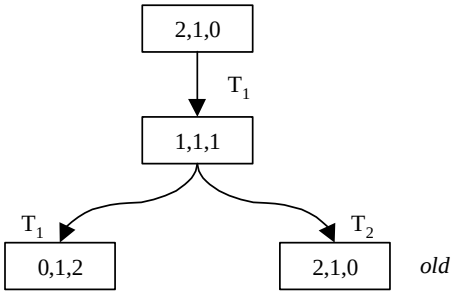


Figura 2.22 Tercer nivel del árbol de alcance

De los estados resultantes comprobamos que el estado resultante de disparar T_2 es el mismo que M_0, con lo que este estado se marca como *old* y no lo continuaremos estudiando. Así, el cuarto nivel del árbol será el resultante de disparar T_2 sobre [0,1,2] ya que éste es el único estado que nos queda por explorar. La transición T_2 es la única que está activada, ya que P_1 no tiene marcas y por lo tanto no se puede activar T_1.

El resultado vuelve a ser un nodo ya explorado, con lo que se ha finalizado la construcción del árbol de alcance.

© Los autores, 2002; © Edicions UPC, 2002.

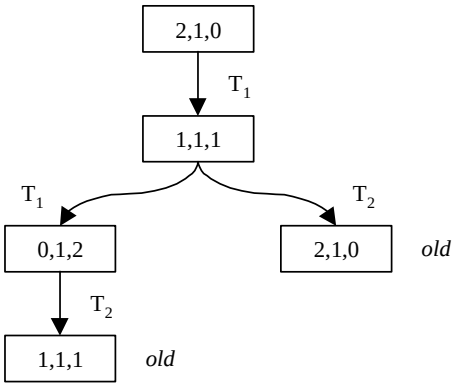


Figura 2.23 Cuarto nivel del árbol de alcance

EJEMPLO 2.6 Línea de producción

Si se considera de nuevo la línea de producción descrita en el ejemplo 2.4 y formalizada en la figura 2.18, y se supone que inicialmente hay 6 piezas en el almacén de entrada, que los 2 robots y las 3 máquinas se encuentran libres, y que el almacén de salida se encuentra vacío. La descripción matemática del estado del sistema queda formalizada en el vector M_0:

$$M_0 = [1,0,1,0,0,6,1,1,0,0,0,0,1,0,0,0]$$

Nótese que el estado del sistema es la información mínima y necesaria, a partir de la cual la trayectoria del mismo queda especificada si se conocen las entradas externas futuras. Así pues, a partir de la información descrita en el vector M_0, puede observarse que la única transición activada es la transición T_1 (los nodos lugares P_6, P_1 y P_3 tienen más de un objeto), lo cual corresponde con el sistema físico, en el que el único evento que puede aparecer es que el robot R_1 transporte una pieza del almacén S_1 a la máquina M_1.

Como consecuencia de la aparición del evento modelado por la transición T_1, el nuevo estado del sistema queda especificado por el vector M_1, el cual indica que el robot R_1 ya no está libre, sino que está realizando una operación de transporte, la máquina M_1 ya no puede considerarse libre, y en el almacén de entrada tan sólo hay 5 piezas.

$$M_1 = [0,1,0,0,0,5,1,1,0,0,0,0,1,0,0,0]$$

Dado el estado M_1, el único evento que puede aparecer en el sistema físico es el modelado por la transición T_2, que corresponde al fin de la operación de transporte de la pieza a la máquina M_1, el cual dejará al robot R_1 en estado libre, y la máquina M_1 en estado de procesamiento. Este nuevo estado al que se accede queda formalizado matemáticamente en el vector M_2:

$$M_2 = [0,0,1,1,0,5,1,1,0,0,0,0,1,0,0,0]$$

De modo análogo podría observarse que el único evento que podría aparecer en el sistema físico (transición activada) es el modelado por la transición T_3, la cual, una vez disparada, permitirá que

© Los autores, 2002; © Edicions UPC, 2002.

puedan aparecer los eventos modelados por las transiciones T_4 y T_8, los cuales, a su vez, podrán desencadenar nuevas secuencias de eventos.

En la figura 2.24 se han descrito los 7 primeros niveles del árbol de alcance de la línea de producción del ejemplo 2.3, en el que se observan las distintas secuencias de eventos que pueden desencadenarse a partir de un cierto estado del sistema.

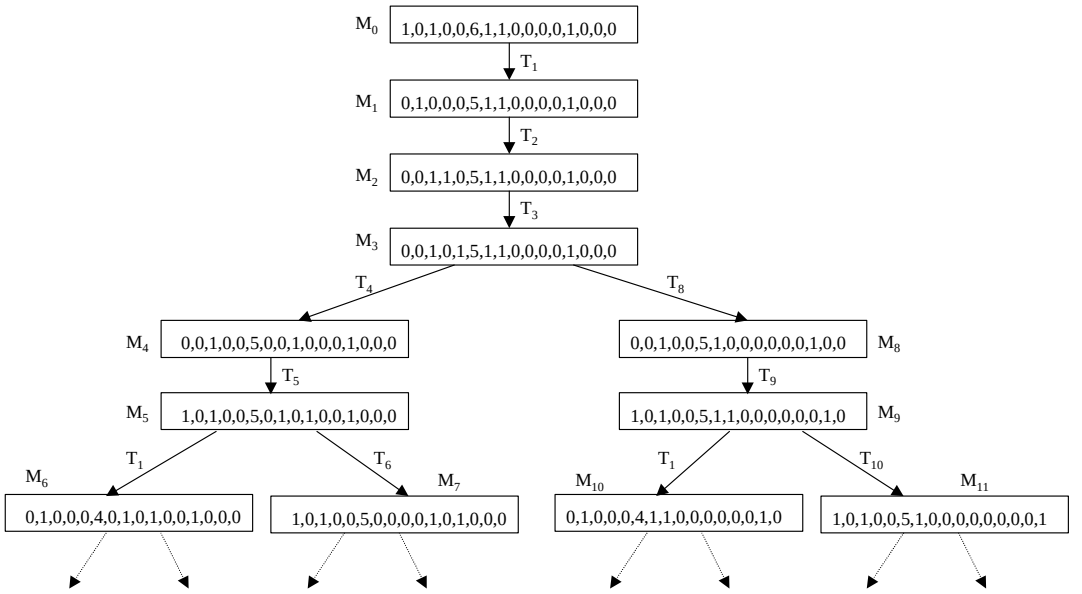


Figura 2.24 Árbol de alcance

Red de Petri acotada

En muchos casos reales, las marcas de la RdP representan piezas o clientes en un sistema con recursos compartidos, donde estas piezas o clientes entran en las colas a la espera de que los recursos estén libres. No parece deseable, por tanto, que la ocupación de las colas crezca hacia el infinito. Este crecimiento sin límite puede indicar que el sistema no esta bien dimensionado o un comportamiento inestable.

Un lugar $P_i \in P$ en una RdP N con estado inicial M_0 se denomina *K-acotado* si $M(P_i) \leq k$ para todos los estados $M \in R(M_0)$. Si es posible encontrar un valor entero K distinto de infinito tal que el lugar es K-acotado, entonces el lugar es acotado. Si todos los lugares de la RdP están acotados, la RdP es acotada. El árbol de alcance permite determinar si la RdP es acotada.

© Los autores, 2002; © Edicions UPC, 2002.

EJEMPLO 2.7 Sistema de producción

En la figura 2.25 se representa un sistema de producción formado por dos máquinas (M_1 y M_2), dos cintas transportadoras con un *palet* en cada una de ellas (C_1 y C_2), y dos tipos de piezas (a y b) para ser procesadas. La secuencia de operaciones tiene que respetar las siguientes restricciones:

1. La secuencia de operaciones para una pieza de tipo a se inicia con el proceso en la máquina M_1; una vez finalizada la operación, se emplea el *palet* C_1 para transportar la pieza a la máquina M_2, una vez procesada sale del sistema.

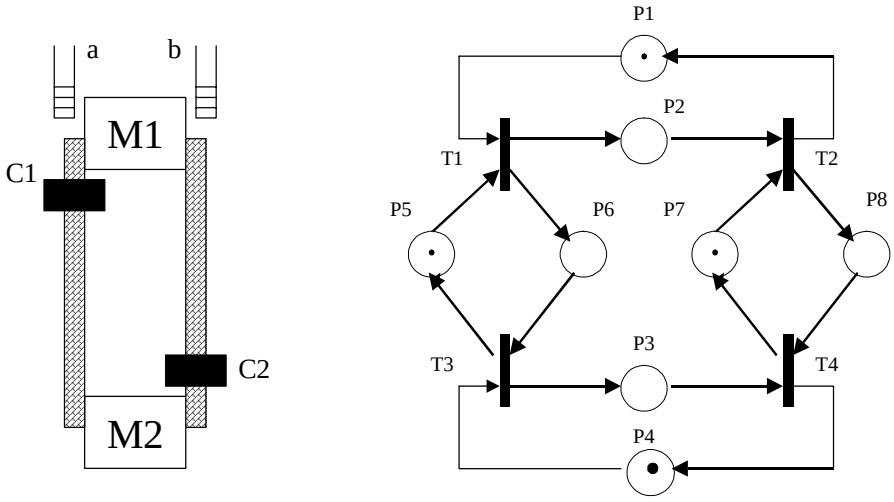


Figura 2.25 Sistema de producción y su RdP

2. La secuencia de operaciones para la pieza de tipo b es la misma que la descrita para las piezas de tipo a, pero se emplea el palet C_2.
3. Las máquinas no pueden realizar consecutivamente dos operaciones sobre el mismo tipo de pieza. Por tanto, cuando una máquina acaba de procesar una pieza de tipo a, la siguiente tiene que ser de tipo b, y viceversa.
4. Para que la máquina M_1 quede libre una vez ha finalizado el proceso de una pieza de tipo a, es necesario que el palet C_1 se encuentre al lado de la máquina M_1.
5. Para que la máquina M_1 quede libre una vez ha finalizado el proceso de una pieza de tipo b, es necesario que el palet C_2 se encuentre al lado de la máquina M_1.

En la figura 2.25 se ha representado la RdP que modela el sistema de producción descrito. Se considera que el almacén de piezas tipo a y tipo b tiene tamaño infinito. También se ha supuesto que los palets C_1 y C_2 tengan solo dos posiciones cada uno, posición superior y posición inferior. El significado asociado a los nodos de la RdP es:

- P_1: palet C_1 en la posición superior (M_1)
- P_2: palet C_1 en la posición inferior (M_2)
- P_3: palet C_2 en la posición inferior (M_2)

© Los autores, 2002; © Edicions UPC, 2002.

- P_4: palet C_2 en la posición superior (M_1)
- P_5: máquina M_1 procesando pieza tipo a
- P_6: máquina M_1 procesando pieza tipo b
- P_7: máquina M_2 procesando pieza tipo b
- P_8: máquina M_2 procesando pieza tipo a

En la figura 2.26 se muestra el árbol de alcance de la RdP. Se observa que se puede llegar a 5 estados distintos a partir del estado inicial M_0. La RdP es, por tanto, acotada (1-acotada), el número de estados a los que se puede llegar a partir del estado inicial es finito y el máximo número de marcas en cada lugar es 1.

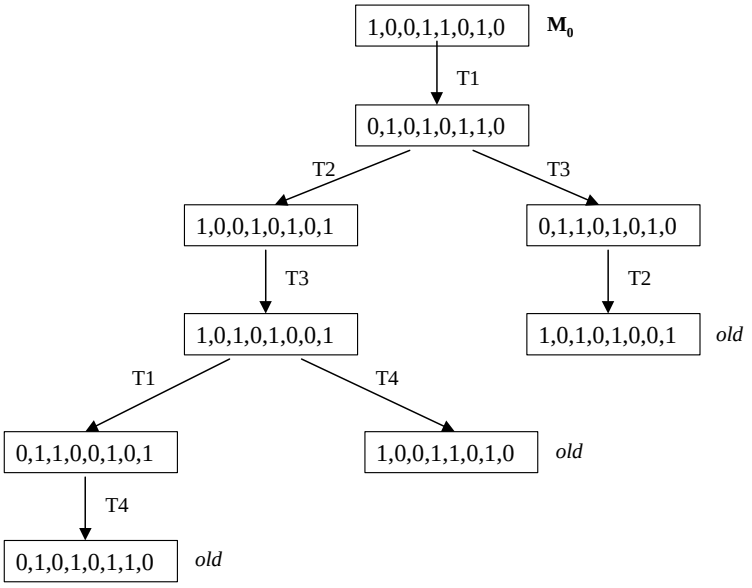


Figura 2.26 Árbol de alcance

Árbol de cobertura

El árbol de cobertura es una variante del árbol de alcance que se emplea para analizar la RdP cuando ésta no es acotada. No es viable trabajar directamente con el árbol de alcance de una RdP no acotada dado que tiene tamaño infinito.

El árbol de cobertura se obtiene a partir del árbol de alcance teniendo en cuenta la siguiente regla adicional:

1. Si existe un nodo M en el camino que hay entre el nodo raíz M_0 y un nodo M^* de cualquier nivel del árbol tal que:

© Los autores, 2002; © Edicions UPC, 2002.

$$M^*(p) \geq M(p) \quad \forall p_i \in P$$
$$M^*(p_k) > M(p_k) \quad \text{para algún } p_k \in P \text{ como mínimo}$$

Entonces el valor p_k se sustituye por ω (ω significa infinito). El valor de p_k se mantendrá en ω en los hijos sucesivos del nodo M*. Todo el conjunto de reglas es también valido para los nodos con valores ω.

EJEMPLO 2.8 Construcción de un árbol de cobertura

La RdP de la figura 2.27 es una red no acotada. Esto lo podemos observar a partir de su estructura. Como vemos, esta red sigue una estructura secuencial y repetitiva dado que, después del disparo de la transición T_1, solo puede dispararse la transición T_2 (secuencialidad). Después de T_2 sólo puede dispararse T_1 (repetitividad). Cuando se dispara la transición T_1 la marca del nodo P_1 vuelve a este nodo. Después del disparo de T_2 se genera otra marca que también va al nodo P_1. De este modo, y para cada secuencia de disparos T_1, T_2, se habrá añadido una marca en el nodo P_1, con lo que el número de marcas de este nodo lugar crecerá hasta el infinito.

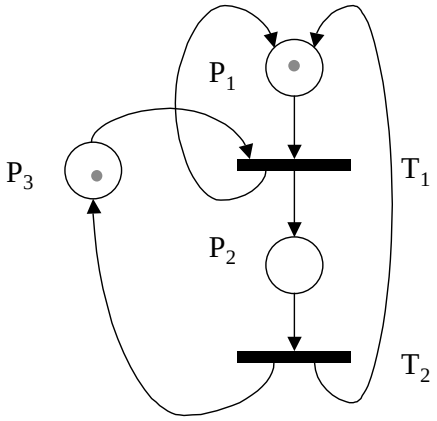


Figura 2.27 RdP no acotada

En consecuencia, el árbol de alcance de esta RdP presenta un crecimiento que tiende al infinito, (figura 2.28a). Si se analizan los estados del árbol de alcance, se puede observar que entre M_0 y cualquier M* es posible aplicar la regla de cobertura. Por ejemplo: si se escoge como M* el estado [3,0,1] existen dos estados entre M_0 y M*, como son [1,0,1] y [2,0,1], donde todos los $M^*(p) \geq M(p)$. Adicionalmente, se cumple también que existe un p_1 que cumple $M^*(p_1) > M(p_1)$. Por tanto es posible convertir p_1 en ω. En la figura 2.28b se muestra el árbol de cobertura para esta RdP.

© Los autores, 2002; © Edicions UPC, 2002.

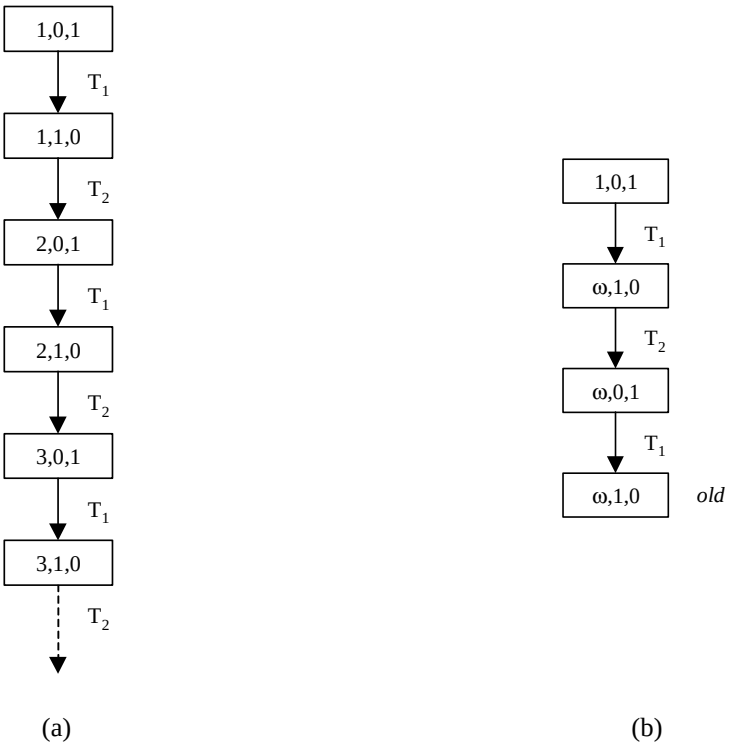


Figura 2.28 Árbol de alcance y árbol de cobertura

EJEMPLO 2.9 RdP no acotada

En la figura 2.29 se muestra una RdP no acotada y su árbol de cobertura. Nótese que el árbol de alcance tiene infinitos nodos. Es decir, el árbol de alcance sustituye el número de marcas de un lugar en un nodo por infinito cuando se observa que este número puede crecer de forma indefinida.

El lugar P1 es 3-acotado, el lugar P2 es 3-acotado pero el lugar P3 no está acotado. Por tanto, la RdP no está acotada al existir, como mínimo, un lugar no acotado.

Bloqueos

Uno de los aspectos más importantes en el análisis y diseño de sistemas concurrentes que comparten recursos, es garantizar que el sistema no pueda entrar nunca en una situación de bloqueo. Imagínese, por ejemplo, las posibles consecuencias del bloqueo del sistema de control de un avión de pasajeros.

Los paquetes de simulación actuales no proporcionan mecanismos para la detección y resolución de bloqueos. Los efectos de los bloqueos generalmente se pueden detectar observando los resultados de la simulación. Normalmente, el resultado de los bloqueos son recursos infrautilizados y colas de espera elevadas y/o una elevada utilización de recursos pero con productividad baja [VENK94].

© Los autores, 2002; © Edicions UPC, 2002.

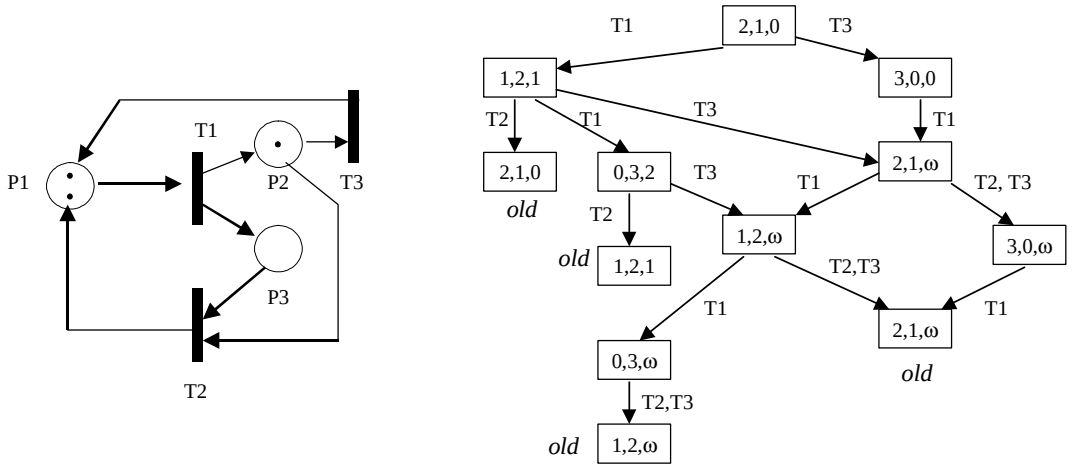


Figura 2.29 Diagrama de cobertura de un RdP no acotada

Las RdP son una buena herramienta para describir las interacciones lógicas entre los subsistemas que integran un proceso. Las RdP ofrecen toda la información necesaria para poder determinar si un sistema, que está correctamente expresado en RdP, puede entrar o no en estado de bloqueo. Más concretamente, aquellos nodos del árbol de cobertura que no tienen ninguna transición activada (nodos *dead end*) describen aquellos estados del sistema en los que, una vez alcanzados, el sistema queda parado. Normalmente, estos nodos indican que el sistema entra en una situación de bloqueo.

Si se ha determinado que el sistema se puede bloquear, a menudo es necesario incluir nuevas restricciones (o eliminar restricciones existentes) para evitar el bloqueo.

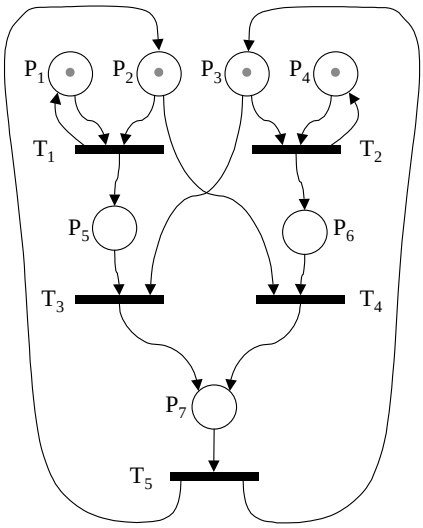


Figura 2.30 RdP con posibles bloqueos

© Los autores, 2002; © Edicions UPC, 2002.

EJEMPLO 2.10 Situación básica de bloqueo

El bloqueo más simple de un sistema ocurre cuando dos procesos concurrentes utilizan recursos compartidos y quedan parados en mitad del proceso en espera de recursos que tiene reservados el otro proceso.

En la RdP de la figura 2.30 se ha presentado el modelo de un sistema con una alta probabilidad de bloquearse. Si la secuencia de disparo de las transiciones es T_1, T_2,... o bien T_2, T_1,... el sistema queda bloqueado ya que no existe ninguna transición activada.

EJEMPLO 2.11 Línea de producción

En la figura 2.31 se representa una línea de producción formada por dos máquinas, M_1 y M_2, y dos manipuladores para transportar una pieza cada uno de forma independiente. Se dispone de dos tipos de piezas a y b que deben ser procesadas según las especificaciones siguientes:

- Se deben realizar dos operaciones sobre la pieza de tipo a, la primera en la máquina M_1 y la segunda en la máquina M_2.
- Se deben realizar dos operaciones sobre la pieza de tipo b, la primera en la máquina M_2 y la segunda en la máquina M_1.
- Para que una pieza del almacén pueda acceder a una máquina es necesario que ésta se encuentre libre y que tengamos una unidad de transporte libre (cualquiera de las dos).

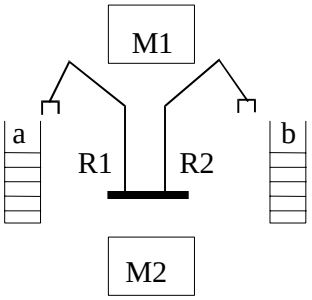


Figura 2.31 Línea de producción

- Una vez se ha asignado una unidad de transporte a una pieza, esta unidad no queda libre hasta que finaliza la segunda operación.
- Después de concluida la primera operación, la pieza es transportada automáticamente (y la máquina de la primera operación queda libre) a la segunda máquina, si está se encuentra libre. En caso contrario, se espera hasta que la segunda máquina queda libre; no pudiéndose liberar la primera máquina.
- Cuando una pieza finaliza la segunda operación, sale del sistema y libera automáticamente tanto la unidad de transporte asignada como la máquina empleada en la segunda operación.

© Los autores, 2002; © Edicions UPC, 2002.

- No puede haber más de una pieza tipo a ni de tipo b procesándose al mismo tiempo en la línea de producción.

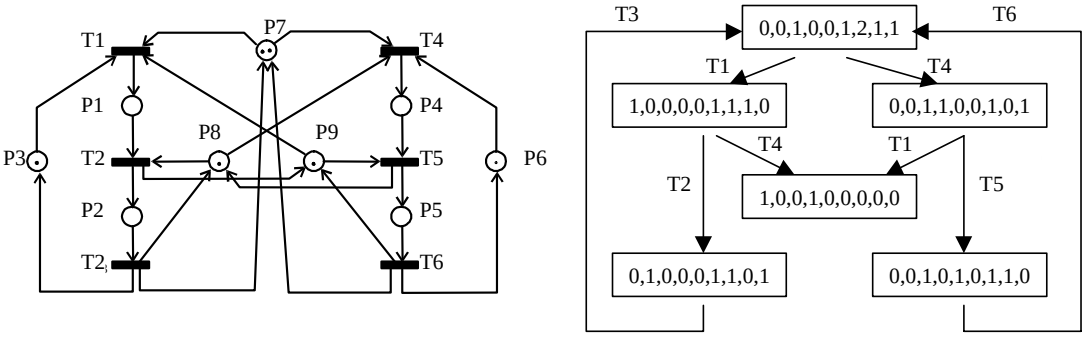


Figura 2.32 RdP y árbol de cobertura de la línea de producción

En la figura 2.32 se representa la RdP que modela el sistema de producción presentado así como su árbol de cobertura (coincide con el de alcance dado que la RdP es acotada).

Los lugares son:

- P_1: M1 procesa una pieza de tipo a
- P_2: M2 procesa una pieza de tipo a
- P_3: limita el máximo número de piezas de tipo a en el sistema
- P_4: M2 procesa una pieza de tipo b
- P_5: M1 procesa una pieza de tipo b
- P_6: limita el máximo número de piezas de tipo b en el sistema
- P_7: número de unidades de transporte libres
- P_8: M2 libre
- P_9: M1 libre

y las transiciones:

- T_1: primera operación sobre una pieza de tipo a en la máquina M1
- T_2: segunda operación sobre una pieza de tipo a en la máquina M2
- T_3: fin se la secuencia de operaciones a realizar sobre una pieza de tipo a
- T_4: primera operación sobre una pieza de tipo b en la máquina M2
- T_5: segunda operación sobre una pieza de tipo b en la máquina M1
- T_6: fin se la secuencia de operaciones a realizar sobre una pieza de tipo b

Se observa en el árbol de cobertura que el sistema se bloquea cuando la máquina M1 y la máquina M2 están realizando la primera operación sobre una pieza de tipo a y b, respectivamente (estado [1,0,0,1,0,0,0,0,0]). Una forma de evitar el bloqueo es incorporar la restricción de que tanto un manipulador como las dos máquinas estén libres antes de iniciar la operación de una pieza, y antes de iniciar la operación ocupar (reservar) las dos máquinas. Si se emplea esta política sólo es necesario un elemento de transporte.

© Los autores, 2002; © Edicions UPC, 2002.

2.4.4 Algunas consideraciones sobre la formalización de modelos conceptuales en RdP

El disponer de un método estructurado para representar el conocimiento que se tiene sobre el comportamiento del sistema es imprescindible para garantizar una futura reutilización de los modelos de simulación. En este sentido, las RdP son ideales como metodología de modelado para capturar las relaciones causales y de precedencia entre situaciones y eventos, las cuales formalizan el conocimiento sobre el comportamiento del sistema necesario para codificar un modelo de simulación.

La especificación del modelo de un sistema orientado a eventos discretos mediante el formalismo de RdP permite obtener información, bien analizando el comportamiento de la red, bien analizando su estructura. El análisis de la estructura permite en algunos casos alcanzar conclusiones rápidas sobre el comportamiento del sistema, relacionando propiedades estructurales y de comportamiento. Aunque se han introducido técnicas de análisis de las RdP, el análisis del comportamiento del modelo se aborda principalmente mediante técnicas de simulación, por lo que la información necesaria para el análisis se obtendrá a partir de la evolución del estado del sistema mediante un simulador orientado a eventos discretos.

A diferencia de la formalización en redes de colas, las RdP permiten formalizar de modo muy natural clientes o peticiones, recursos y procesos como marcas situados en los nodos lugar. Esta generalidad que ofrecen las RdP presenta el inconveniente de no disponer de símbolos especiales para representar entidades, y la interpretación de la red (en algunos casos) se realiza a partir de una correcta asignación de nombres a los nodos lugar y transición. Una diferencia significativa entre ambos formalismos es que las redes de colas especifican la semántica de los componentes del modelo en el propio formalismo de modelado, mientras que las RdP no. Esta característica es una gran ventaja en cuanto permite que las RdP puedan ser utilizadas como herramientas de modelado en nuevos campos de aplicación. Nótese que este grado de libertad que ofrecen las RdP, si es correctamente explotado, permite escoger el nivel de abstracción idóneo al problema que se quiera tratar [QMIP93].

Otra característica importante es que cualquier tipo de sincronización (semáforos, monitores, *rendez-vous*) puede ser formalizada en RdP a partir de un constructor básico (más de un arco de entrada a una transición), mientras que en redes de colas todas las variaciones de mecanismos de sincronización deben ser enumeradas para preservar la semántica de los componentes (recursos pasivos, bloqueos, bifurcadores, separadores, etc.).

En opinión de los autores, la potencia, en cuanto a herramienta de modelado que ofrece el formalismo de las RdP respecto a otros formalismos, es la descripción de la relación estado-evento. Como se ha podido observar a través de los ejemplos presentados, las RdP permiten representar de modo muy simplificado los cuatro conceptos básicos de modelado: dependencia causal (secuencias de eventos), nodos de decisión, concurrencia y sincronismo. Otros aspectos imprescindibles al modelar un sistema orientado a eventos discretos que hacen de las RdP un formalismo muy potente, es el modelado progresivo mediante la aproximación *top-down* (a partir de un modelo inicial muy abstracto mediante una secuencia de mejoras) o *bottom-up* (desarrollo de varios submodelos en paralelo).

No obstante, en general, tanto en los sistemas de producción como en los sistemas de transporte y de servicios, los eventos que se deben desencadenar no dependen tan sólo de la arquitectura del sistema, sino que también dependen de información asociada a los elementos que integran el sistema o a las entidades temporales que fluyen entre las unidades descritas en la arquitectura. La codificación en el

© Los autores, 2002; © Edicions UPC, 2002.

formalismo de RdP de la información que determina los posibles eventos que se deben disparar (por ejemplo, en función de la urgencia de ciertas peticiones) repercute en un incremento considerable en el número de nodos lugar y nodos transición, lo que dificulta el mantenimiento y posible explotación del modelo desarrollado. Las redes de Petri coloreadas (RdPC) reducen la dimensión del modelo al incrementar su nivel de abstracción y permitir que las marcas tengan asociada información que denominaremos *colores* (equivalente al concepto de *atributos*).

2.5 Las redes de Petri coloreadas

Para una correcta codificación y futuro mantenimiento del modelo de simulación es muy conveniente que el modelo conceptual formalice tanto el flujo de información en el proceso como los cambios de estado. Las RdP, como herramienta de modelado, ofrecen el formalismo necesario para representar tanto el estado de cada uno de los componentes del sistema, como la secuencia de eventos que pueden desencadenarse a partir de un cierto estado del sistema; pero no permiten especificar el flujo de información que suele especificarse mediante datos asignados a entidades cuyos valores cambian en función de los eventos que aparecen.

Las RdPC permiten construir modelos más compactos y paramétricos, lo que facilita considerablemente su mantenimiento y su posterior codificación. Estos modelos requerirían de estructuras con un número elevado de componentes si fueran desarrollados con el formalismo de las RdP.

EJEMPLO 2.12 Proceso productivo

Considérese un sistema de producción integrado por 4 máquinas, M_1,..M_4, en el cual fluyen indistintamente 2 tipos de piezas (A: piezas de cobre y B: piezas de acero), sobre las que se deben realizar el mismo tipo de operaciones (fresado, perforación, erosión y ensamblado) y en la misma secuencia, pero el tiempo de operación dependerá de si se trata de una pieza de cobre o de acero. En la figura 2.33 se ilustra la secuencia de 4 máquinas, cada una de ellas con un almacén local, junto con un almacén de materia prima y otro de producto acabado. En la misma figura también se representa la RdP (parte izquierda) y RdPC del sistema de producción, teniendo en cuenta que para realizar una operación de ensamblado es necesario disponer de una pieza de acero y una de cobre correctamente procesadas.

Las transiciones de la RdP son:

- T1: fresado sobre una pieza de acero
- T2: perforación sobre una pieza de acero
- T3: erosión sobre una pieza de acero
- T4: ensamblado de una pieza de acero con una pieza de cobre
- T5: fresado sobre una pieza de cobre
- T6: perforación sobre una pieza de cobre
- T7: erosión sobre una pieza de cobre

© Los autores, 2002; © Edicions UPC, 2002.

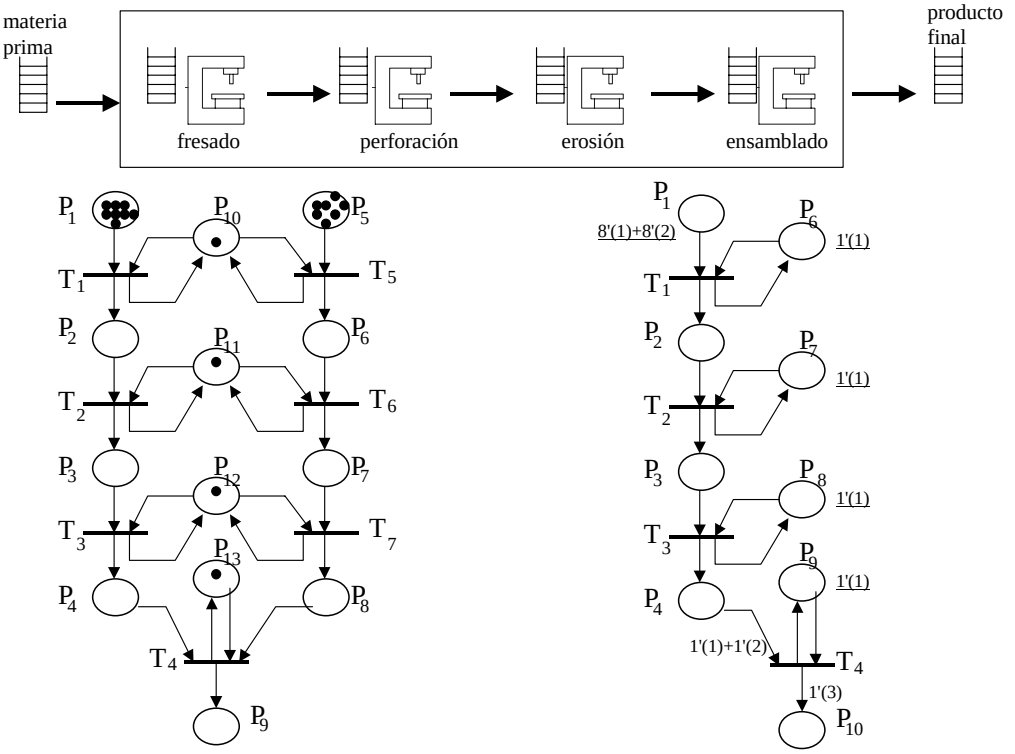


Figura 2.33 RdP y RdPC de una secuencia de tareas sobre 2 tipos diferentes de piezas

El significado de los nodos lugar es:

- P1: almacén de piezas de acero
- P2: piezas de acero fresadas
- P3: piezas de acero perforadas
- P4: piezas de acero erosionadas
- P5: almacén de piezas de cobre
- P6: piezas de cobre fresadas
- P7: piezas de cobre perforadas
- P8: piezas de cobre erosionadas
- P9: piezas ensambladas.
- P10: máquina fresadora libre
- P11: máquina perforadora libre
- P12: máquina erosionadora libre
- P13: máquina ensambladora libre

En el ejemplo descrito, puede observarse que la RdP que formaliza el flujo de piezas de tipo cobre es la misma que describe el flujo de piezas de tipo acero. A pesar de ello, es necesario formalizar ambas redes para indicar al programador que el tiempo de los eventos depende del tipo de pieza que se va a procesar.

© Los autores, 2002; © Edicions UPC, 2002.

De modo análogo, existe un conjunto de aspectos asociados a las entidades que fluyen entre los elementos que integran el sistema a modelar, que suelen repercutir no tan sólo en el tiempo de disparo de las transiciones (duración de los eventos), sino también en la activación de las transiciones, así como en el flujo de nuevos eventos a desencadenar. En general, son muchas las situaciones en las que es necesario particularizar (diferenciar) los objetos (marcas) que fluyen en el sistema (en la RdP). El formalismo de las RdP no facilita la descripción de aquellas actividades cuyo comportamiento depende del estado de las entidades.

Las RdPC permiten formalizar tanto los atributos o características de los objetos que fluyen en el sistema, como las propiedades que deben tener para que un cierto evento pueda suceder. Ejemplos de atributos asociados a una pieza (entidad) que debe ir avanzando en las diferentes etapas del proceso productivo, pueden ser: tiempo de llegada al sistema, fecha de entrega al cliente, prioridad, color, forma, etc.

2.5.1 Definiciones

La principal diferencia que aportan las RdPC respecto a las RdP ordinarias es la capacidad de asociar a cada objeto (marca) un tipo de datos (conjunto de valores) denominado *color del objeto* (color de la marca). El uso de colores es análogo al uso de tipos de datos en los lenguajes de programación, lo cual dota a las RdPC de la potencia necesaria para poder formalizar el modelo de cualquier sistema, por complejo que éste sea (análogo a la potencia de los actuales lenguajes de programación).

Matemáticamente, *una RdPC puede definirse a partir de la siguiente tupla* [JENS97]:

$$RdPC = (\Sigma,P,T,A,N,C,G,E,I)$$

donde:

- $\Sigma = \{C_1,C_2,...,C_{nc}\}$: conjuntos finitos y no vacíos de colores. Permite especificar los atributos que deben definirse para cada tipo de entidad que se quiera modelar. Ci son los colores (atributos) de la entidad de tipo i.
- $P = \{P_1,P_2,P_3,...,P_{np}\}$: conjunto finito de nodos lugar que permiten especificar el estado del sistema.
- $T = \{T_1,T_2,T_3,...,T_{nt}\}$: conjunto finito de nodos tipo transición. Las transiciones en el modelo de simulación corresponden a eventos que suelen codificarse como el inicio o fin de una cierta actividad, o bien como el fin de un suceso externo como sería el caso de un proceso de llegadas.
- $A = \{A_1,A_2,...,A_{na}\}$: conjunto finito de arcos.
- N: función de nodo, $N(A_i)$, que permite asociar a cada arco sus nodos terminales (el nodo origen y el nodo destino). Los dos nodos tienen que ser de tipo diferente; por tanto, si un nodo es una transición el otro debe ser un lugar y viceversa.

© Los autores, 2002; © Edicions UPC, 2002.

C: conjunto de funciones color, $C(P_i)$, que permiten especificar para cada nodo lugar el tipo de entidades que pueden almacenarse:

$$C(P_i) = C_j \qquad P_i \in P,\ C_j \in \Sigma$$

G: función guarda asociada a los nodos tipo transición, $G(T_i)$, que suele utilizarse para desinhibir el evento asociado a la transición en función de los valores de los atributos de la entidad que se quiere procesar.

E: expresiones asociadas a arcos, $E(A_i)$, que permiten especificar el tipo de entidad del nodo lugar de entrada a la transición que debe escogerse de entre las entidades almacenadas en el nodo lugar para habilitar el evento. Cuando la expresión E se encuentra asociada a un arco de salida de la transición, la expresión se utiliza para evaluar los nuevos valores de los atributos de las entidades de salida.

I: función de inicialización, $I(P_i)$, que permite especificar los valores de los colores (atributos) de las entidades inicialmente almacenadas en los nodos lugar.

A continuación se explican más en detalle algunos de los conceptos anteriores:

- *Expresiones de inicialización (I)*: el estado inicial se determina ejecutando las expresiones de inicialización asociadas a cada nodo lugar, las cuales determinarán el número de objetos en cada nodo, así como los valores de los colores de los objetos.

  Las *expresiones de inicialización* indican el número inicial de objetos y su color en cada uno de los nodos lugar. Permiten obtener el marcado inicial M_0. Gráficamente, las expresiones de inicialización expresan el número de objetos en un nodo lugar con un número en un círculo al lado del nodo. Los colores de los objetos se especifican mediante una expresión subrayada junto al nodo lugar con la siguiente información:

  $$\underline{n'(c_1, c_2, c_3, \ldots, c_k)}$$

  donde:

  n: número de objetos con los valores de los colores descritos dentro del paréntesis
  c_i: valor del color i

  Cuando los valores de los colores de los objetos no son idénticos para todos los elementos del mismo lugar, se utiliza el operador "+" para especificar los valores de los colores de cada objeto. A modo de ejemplo, la expresión de inicialización del nodo lugar P_1 formalizada en la figura 2.33 significa que el almacén de materia prima contiene inicialmente 8 piezas de cobre (atributo tipo de pieza = 1) y 8 piezas de acero (atributo tipo de pieza = 2).

- *Expresiones de arco (E)*: los colores de los objetos pueden ser inspeccionados en las transiciones, lo que permitirá activar las transiciones no tan sólo en función del número de objetos en los nodos lugar conectados a la entrada de la transición, sino también en función del tipo (color) de los objetos disponibles en dichos nodos lugar, y al mismo tiempo permitirá también modelar los efectos de cada transición, definiendo nuevos colores en los objetos de salida.

© Los autores, 2002; © Edicions UPC, 2002.

Las expresiones de arco consisten en la formalización de restricciones entre los colores de los distintos objetos de los nodos lugar conectados a la entrada de la transición, para la cual pueden utilizarse variables que, una vez instanciadas a valores concretos de los colores de los objetos, fuerzan a una selección de aquellos objetos cuyos colores coincidan con los valores de las variables instanciadas. A modo de ejemplo, si se considera la transición T_4 de la RdPC representada en la figura 2.33, para que ésta se encuentre activada es necesario que en el nodo lugar P_4 exista como mínimo una pieza de cobre (1'(1)) y una pieza de acero (1'(2)).

- *Guardas (G):* los guardas tienen una funcionalidad similar a las expresiones de arco, pero tan sólo son expresiones booleanas que imponen ciertos valores a los colores de los objetos que pueden ser escogidos para activar una transición. Permiten imponer condiciones ligadas a más de un arco de entrada. Gráficamente se formalizan entre corchetes situados al lado de la transición.
- *Nodos lugar:* cada nodo lugar tan sólo puede tener objetos con el mismo color (el mismo tipo de datos), el cual es conocido como *conjunto color del lugar* (*color set*).

Adicionalmente (aunque dicha información no obedece al estándar de RdPC), es posible formalizar las *prioridades entre transiciones* mediante una función Π: T $\rightarrow$ N que permite asignar a cada transición un número natural no negativo, el cual representa su prioridad. De este modo, para disparar una transición habilitada es necesario que ninguna otra transición habilitada tenga una prioridad superior.

Las técnicas de análisis presentadas para las RdP también son aplicables para las RdPC. La diferencia más significativa es que el *estado* (marcado M de la RdPC) del sistema no tan sólo se describe mediante la especificación del número de objetos (*tokens*) en cada nodo lugar, sino que también son necesarios los valores de los colores de cada uno de los objetos.

EJEMPLO 2.13

Para ilustrar el funcionamiento de las RdPC se emplea una versión reducida del sistema del ejemplo 2.12, donde existen dos procesos paralelos que comparten un mismo recurso.

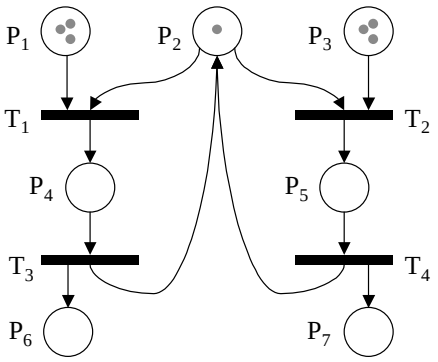


Figura 2.34 RdP de un proceso paralelo

© Los autores, 2002; © Edicions UPC, 2002.

El primer paso a realizar es expresar esta RdP de forma coloreada. La transformación básica de esta RdP es la formalización de los dos procesos mediante un único nodo lugar, donde se diferencia el tipo de elemento que se está procesando y el estado de la máquina. Así pues, el nodo P_1 y P_3 se fusionan en un único nodo que almacenará marcas de tipo 1 y tipo 2 (tres de cada tipo).

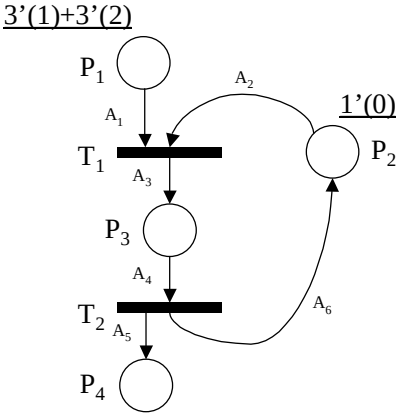


Figura 2.35 RdPC de un proceso paralelo

En primer lugar hemos de definir qué colores (atributos) deben representarse. Para este ejemplo concreto, la información a codificar es el tipo de pieza que se procesa y el estado de la máquina. Puesto que el nodo P_3 representa la máquina procesando una pieza, será necesario un tercer color que sea el producto cartesiano de los otros dos.

- $\Sigma = \{x,y,R\}$, donde:
  - x: tipo de pieza que se debe procesar (1, 2)
  - y: estado de la máquina (0, libre; 1, ocupada)
  - R: producto cartesiano de x por y. Da información del estado de la máquina y del tipo de pieza que procesa

Los nodos lugar son una fusión de los nodos de la RdP de la figura 2.34:

- $P=\{P_1,P_2,P_3,P_4\}$, donde:
  - P_1: representa el almacén de piezas
  - P_2: máquina libre
  - P_3: procesado de las piezas
  - P_4: salida de piezas procesadas

Las transiciones de esta RdPC también son una fusión de las transiciones de la RdP de figura 2.34.

- $T=\{T_1,T_2\}$, donde:
  - T_1: se inicia el proceso de una pieza en la máquina
  - T_2: finaliza el proceso de la pieza en la máquina

© Los autores, 2002; © Edicions UPC, 2002.

Los arcos de la RdPC son los siguientes:

- $A=\{A_1,A_2,A_3,A_4,A_5,A_6\}$

Las funciones de nodo N especifican los nodos origen y destino de cada arco:

- $N(A_1)=(P_1,T_1)$; $N(A_2)=(P_2,T_1)$; $N(A_3)=(T_1,P_3)$; $N(A_4)=(P_3,T_2)$; $N(A_5)=(T_2,P_4)$; $N(A_6)=(T_2,P_2)$

Las guardas de las transiciones estarán todas activas ya que en ningún momento se restringe el tipo de pieza que activa una transición.

- $G(T_1)=true$; $G(T_2)=true$

Los conjuntos de colores asociados a cada nodo vienen determinados por los elementos que hay en este nodo. En los nodos P_1 y P_4 hay piezas, luego el color será el tipo de pieza. En el nodo P_2 hay elementos máquina, pues el color asociado será el estado de la máquina. Finalmente el nodo P_3 representa el procesado de una pieza por una máquina, luego tendrán que estar presentes los colores del tipo de pieza y estado de la máquina.

- $C(P_1)=x$; $C(P_2)=y$; $C(P_3)=R$; $C(P_4)=x$

El estado inicial de la RdPC se puede deducir a partir de la RdP. En el nodo P_1 estarán las piezas que se deberán procesar, tres de tipo 1 y tres de tipo 2. En el nodo P_2 se indicará que la máquina se encuentra libre.

- $I(P_1)=3'(1)+3'(2)$; $I(P_2)=1'(0)$

Finalmente queda especificar las expresiones de arco, que indicarán cuántas y de qué color son las marcas que pueden transitar por un arco determinado.

- $E(A_1)=1'(x)$; $E(A_2)=1'(0)$; $E(A_3)=1'(x,1)$; $E(A_4)=1'(x,1)$; $E(A_5)=1'(x)$; $E(A_6)=1'(0)$

EJEMPLO 2.14

Se considera un sistema de producción (figura 2.36) integrado por 3 tipos de máquinas (2 máquinas de erosión, 2 máquinas de mecanizado y 2 máquinas de ensamblado), con un conjunto de piezas que deben realizar las 3 operaciones (en el orden expuesto). La ocurrencia de un evento correspondiente a la finalización de una tarea en una de las máquinas de mecanizado puede comportar bien el inicio de una nueva actividad de mecanizado si existe una pieza esperando (con la operación de erosión finalizada), bien un estado de hibernación a la espera de que se finalice una operación de erosión, con lo que el sistema de producción tendría una máquina de mecanizado libre.

© Los autores, 2002; © Edicions UPC, 2002.

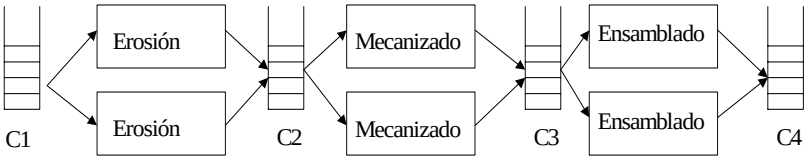


Figura 2.36 Sistema de producción

En la figura 2.37 se describe la RdPC del proceso. Si se compara con la RdPC de la figura 2.33 se observa que, aunque los procesos son muy parecidos, el nivel de abstracción conseguido en este ejemplo es muy superior. No obstante, no es posible afirmar que este modelo sea mejor que el otro. El nivel idóneo de abstracción dependerá del propio usuario, de la herramienta de simulación adoptada y del uso deseado del modelo. Sin embargo, según los autores, es conveniente que la estructura de la RdPC del proceso mantenga un paralelismo con la estructura del proceso real. Esto facilitará el mantenimiento, mejora, adaptación y reutilización posterior del modelo en RdPC y del modelo de simulación.

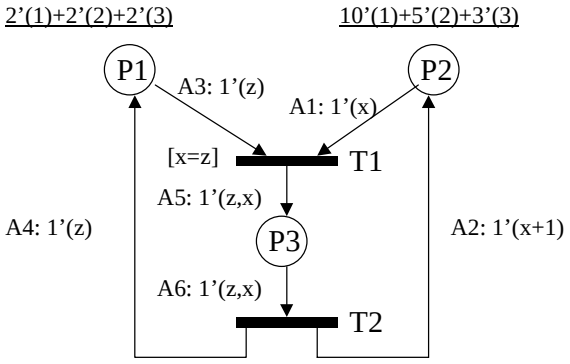


Figura 2.37 RdPC del sistema de producción

- $\Sigma = \{x,z,R\}$, donde:
  - x: conjunto de colores de valores 1, 2, 3 y 4 que indican la cola donde está situada la pieza (colas C1,C2, C3 o C4, respectivamente)
  - z: tipo de máquina (1, erosión; 2, mecanizado; 3, ensamblado)
  - R: producto cartesiano $z \times x$. Da información sobre el estado de la máquina

- P={P1,P2,P3}, donde:
  - P1: representa el estado en que las máquinas están libres. El conjunto color asignado a esta plaza es el color z
  - P2: piezas en las colas pendientes de procesar. El conjunto color que describe el estado de esta plaza es el color x
  - P3: piezas en las máquinas procesándose. El conjunto color que describe el estado de esta plaza es el color R

© Los autores, 2002; © Edicions UPC, 2002.

- T={T1,T2}, donde:
  T1: se inicia el proceso de una pieza en la máquina
  T2: finaliza el proceso de la pieza en la máquina

- A={A1,A2,A3,A4,A5,A6} es el conjunto de arcos de la RdPC.

- N(A1) = (P2,T1), ..., N(A6) = (P3,T2), especifica los nodos origen y destino de cada arco.

- C(P1) = z, C(P2) = x, C(P3) = R, conjuntos de colores asociados a cada lugar.

- G(T1) = {x=z}, G(T2) = verdadero. Para la transición T1 sólo se permite la activación del evento si el valor de z (tipo de máquina) coincide con el de x (cola donde está situada la pieza). Es decir, por ejemplo, la máquina de ensamblado sólo puede procesar piezas que están en la cola C3.

- I(P1) = 2'(1)+2'(2)+2'(3) . Inicialmente tenemos dos máquinas libres de cada tipo.
  I(P2) = 10'(1)+5'(2)+3'(3). Inicialmente hay 10 piezas en la cola C1, 5 en la C2 y 3 en la C3.

- E(A1) = 1'(x) . Para que se active la transición T1 es necesaria una pieza en la cola x.
  E(A2) = 1'(x+1) . Se incrementa el número de piezas en la cola x+1.
  E(A3) = 1'(z) . Es necesaria una máquina libre para operar la pieza.
  E(A4) = 1'(z) . La máquina de tipo z ha quedado libre después de la operación.
  E(A5) = 1'(z,x) . La máquina z pasa a estar ocupada con la pieza x.
  E(A6) = 1'(z,x) . Es necesaria una máquina ocupada con la pieza x para que después pase a libre.

El marcado inicial M_0 de la RdPC se puede obtener a partir de las expresiones de inicialización:

$$M_0 = [2'(1) + 2'(2) + 2'(3),\ 10'(1) + 5'(2) + 3'(3),\ \varnothing]$$

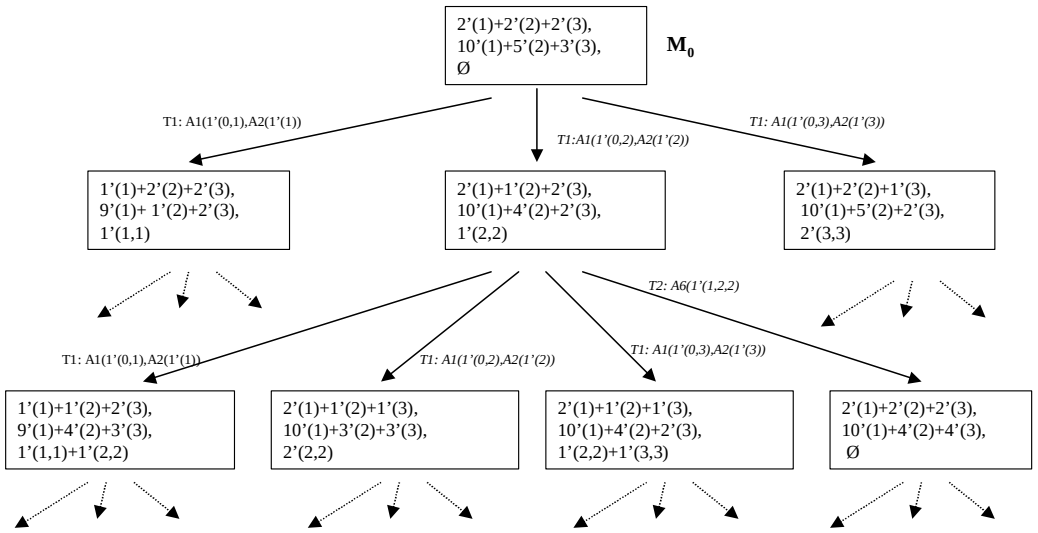


Figura 2.38 Árbol de cobertura

© Los autores, 2002; © Edicions UPC, 2002.

La figura 2.38 muestra el árbol de cobertura del proceso anterior. Se observa que se puede activar más de una vez una misma transición a partir de un mismo nodo del árbol, cada transición tiene diferentes colores de activación. Debido a la explosión combinatoria, el análisis directo de procesos complejos a través de sus redes de Petri sólo es a menudo posible mediante la ayuda de herramientas informáticas. Su interés principal en el contexto de este libro es cómo ayuda al modelado, a la verificación y a la validación del modelo de simulación.

EJEMPLO 2.15

Se desea modelar el servicio de urgencias de un hospital. Los elementos estructurales principales del servicio son:

1. Un punto de recepción al cual se dirigen en primer lugar los pacientes atendido por una enfermera de recepción (er).
2. Una sala de espera externa (see) para los pacientes que están pendientes de la primera visita.
3. Cinco habitaciones (*boxes* -b-) equipadas para la atención del paciente.
4. Una sala auxiliar (sa) para efectuar extracciones de sangre, vendajes o otras pruebas auxiliares.
5. Un equipo de rayos X (rx).
6. Una sala de espera interna (sei) para los pacientes que ya han efectuado la primera visita y están pendientes de una segunda visita.

El equipo médico principal del servicio es:

1. Una enfermera de recepción (er) que registra a los pacientes y efectúa una primera evaluación de su gravedad.
2. Dos médicos con experiencia (séniors -s-) y un médico en formación (júnior -j-). Las dos diferencias más significativas entre los dos tipos de médicos son que el médico sénior es capaz de atender al paciente con menos tiempo y sin merma en la calidad; y que el medico júnior puede precisar de la ayuda de un sénior para confirmar el diagnóstico.
3. Un enfermero en la sala auxiliar para las pruebas complementarias.
4. Un técnico en la sala de rayos X.

El flujo del paciente dentro del servicio de urgencias depende fundamentalmente del tipo de diagnóstico y del tratamiento específico que desee efectuar el médico. Se han determinado ocho circuitos básicos mostrados en la siguiente tabla:

núm.								
1	er	see	b	sa	rx	sei	b	alta
2	er	see	b	sa	sei	b	alta	
3	er	see	b	rx	sa	b	alta	
4	er	see	b	rx	b	alta		
5	er	see	b	sa	b	alta		
6	er	see	b	alta				
7	er	see	b	sei	b	alta		
8	er	em						

Tabla 2.1 Circuitos asistenciales

© Los autores, 2002; © Edicions UPC, 2002.

Los aspectos a destacar son:

- En todos los circuitos el paciente pasa por la recepción (er). Exceptuando el circuito 8, en el resto de los circuitos el paciente se espera en la sala de espera externa (see) a la espera de que quede un *box* (b) libre para poder ser atendido por un médico (m). El circuito 8 corresponde a aquellos casos en donde la enfermera de recepción considera que el paciente está muy grave y lo deriva directamente al servicio de emergencias del hospital (em).
- Después de efectuada la primera visita, el paciente se puede dirigir a la sala auxiliar para:
  1. Efectuar pruebas analíticas, extracción de sangre y orina principalmente (circuitos 1 y 2), u otras pruebas, como por ejemplo vendajes (circuito 5).
  2. Efectuar una placa de rayos-X (rx).
- En otros muchos casos, el paciente recibe directamente el alta después de esta primera visita. Otra opción es que se derive a la sala de espera interna; esto puede ocurrir, por ejemplo, cuando hay mareos o vómitos que hacen recomendable que el paciente permanezca en observación en la sala de espera interna (sei).
- Si hay análisis de sangre u orina (circuitos 1 y 2), el paciente debe esperar en la sala de espera interna (sei) hasta que el laboratorio no entregue los resultados.
- Exceptuando los circuitos 6 y 8, en el resto de los circuitos es necesaria una segunda revisión médica en el *box* para confirmar el diagnóstico con la ayuda de las pruebas o placas efectuadas. La cola de espera para la segunda revisión está en la sala de espera interna (sei).
- Por alta se entiende tanto el alta hospitalaria (y el paciente regresa a casa) como el alta del servicio y posterior ingreso en el hospital.

La figura 2.39 muestra la RdPC del servicio de emergencias, los elementos de la tupla de la red son:

- Σ = {p,x,m,b,er,sa,rx,PX,PXM}, donde:
  - p: tipos de pacientes según el circuito que recorren (1,2,...,8)
  - x: primera o segunda visita al *box* (1, primera; 2, segunda)
  - m: tipos de médicos según experiencia (s, sénior; j, júnior)
  - b: *boxes* de atención a los pacientes
  - er: enfermera de recepción
  - sa: sala auxiliar
  - rx: sala de rayos X
  - PX: producto cartesiano p*x. Da información sobre el tipo de paciente y su estado (vuelta 1 o vuelta 2)
  - PXM: producto cartesiano p*x*m. Da información sobre el tipo de paciente, su estado y el tipo de médico que lo visita

- P={P_1,P_2,...,P_{20}}, donde:
  - P_1: cola de espera en la recepción
  - P_2: enfermera de recepción (er) libre
  - P_3: enfermera de recepción que está registrando a un paciente
  - P_4: paciente en la sala de espera externa (see) en la cola para la primera visita en el box

© Los autores, 2002; © Edicions UPC, 2002.

P_5: *boxes* (b) libres
P_6: pacientes en la sala de espera interna a la espera de la segunda visita
P_7: pacientes en el *box* preparándose para la visita (por ejemplo, desvistiéndose)
P_8: médicos libres
P_9: médicos visitando a pacientes en los *boxes*
P_{10}: médico júnior y paciente esperando a un médico sénior
P_{11}: médico júnior y sénior visitando al paciente
P_{12}: paciente en proceso de desocupar el *box* (por ejemplo, vistiéndose)
P_{13}: pacientes en cola para acceder a la sala auxiliar
P_{14}: paciente en la sala auxiliar
P_{15}: sala auxiliar (sa) libre
P_{16}: paciente en la cola para acceder a la sala de rayos X (rx)
P_{17}: sala de rayos X libre
P_{18}: paciente que está ocupando la sala de rayos X
P_{19}: paciente en la sala de espera interna a la espera de los resultados de la analítica o en observación
P_{20}: alta
P_{21}: derivación del paciente a emergencias

- T={T1,T2,... ,T15},donde:
  T_1: proceso de llegada de pacientes
  T_2: inicio del proceso de registro
  T_3: fin del proceso de registro
  T_4: entrada por primera vez en el *box*
  T_5: entrada por segunda vez en el *box*
  T_6: inicio de la visita del médico
  T_7: fin de la visita del médico o inicio de la espera de un médico sénior
  T_8: inicio de la visita conjunta de un médico senior y un médico júnior
  T_9: fin de la visita conjunta de un médico sénior y un médico junior
  T_{10}: el paciente desocupa el *box*
  T_{11}: entrada en la sala auxiliar
  T_{12}: salida de la sala auxiliar
  T_{13}: entrada en la sala de rayos X
  T_{14}: salida de la sala de rayos X
  T_{15}: fin del proceso de espera o de observación en la sala de espera interna

- A={A1,A2,...,A51} es el conjunto de arcos de la RdPC.

- $N(A1) = (T_1,P_1)$, ..., $N(A51) = (T_3,P_{21})$ especifica los nodos origen y destino de cada arco.

- $C(pl) = p$ si $pl \in \{P_1,P_3,P_{21}\}$
  $C(pl) = er$ si $pl = P_2$
  $C(pl) = PX$ si $pl \in \{P_4,P_6,P_7,P_{10},P_{11},P_{12}, P_{13},P_{14},P_{16},P_{18},P_{19},P_{20}\}$
  $C(pl) = b$ si $pl = P_5$

© Los autores, 2002; © Edicions UPC, 2002.

C(pl) = m si pl = P_8
C(pl) = sa si pl = P_{15}
C(pl) = rx si pl = P_{17}
C(pl) = PXM si pl = P_9

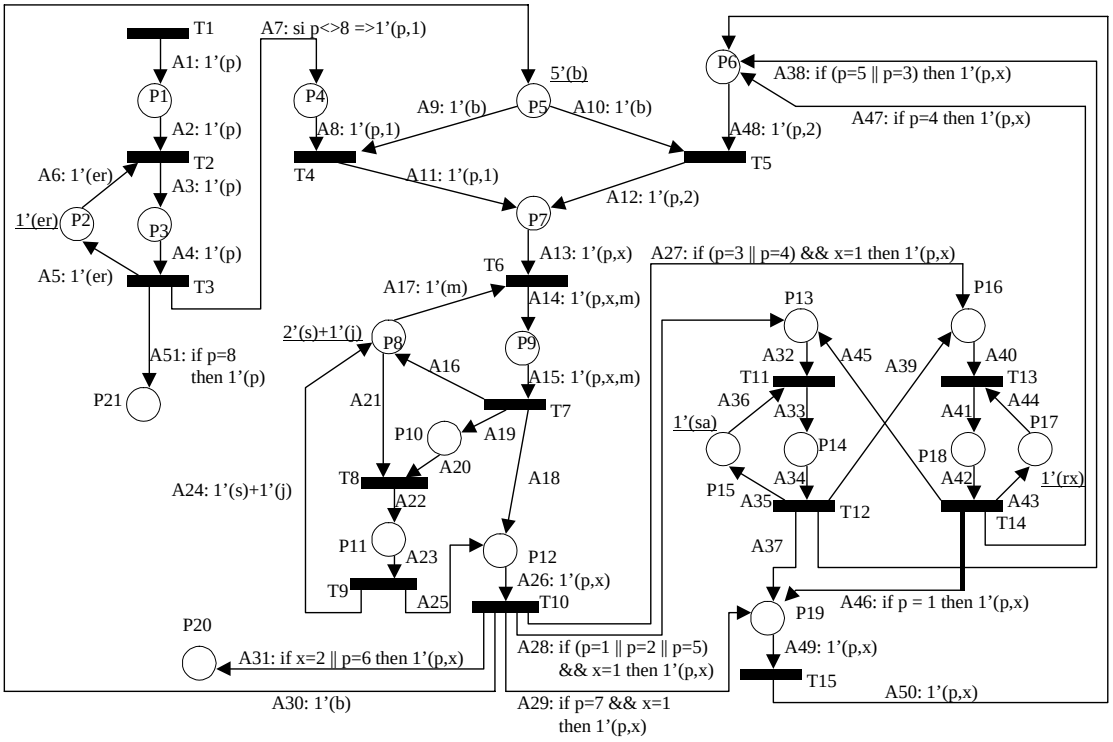


Figura 2.39 RdPC del servicio de urgencias

- G(t) = verdadero si t ∈ $\{T_1,T_2,...,T_{15}\}$

- $I(P_1)$ = 1'(er) . Inicialmente, tenemos una enfermera de recepción libre.
  $I(P_5)$ = 5'(b). Inicialmente, hay 5 *boxes* libres.
  $I(P_8)$ = 2'(s)+1'(j). Inicialmente, hay 2 médicos séniors y 1 médico júnior libres.
  $I(P_{15})$ = 1'(sa). Inicialmente, la sala auxiliar está libre.
  $I(P_{17})$ = 1'(rx). Inicialmente, la sala de rayos X está libre.

- E(a) = 1'(p) si a ∈ {A1,A2,A3,A4}
  E(a) = 1'(er) si a ∈ {A5,A6}
  E(a) = 1'(p,1) si a ∈ {A8,A11}
  E(a) = 1'(p,x) si a ∈ {A13,A20,A22,A23,A25,A26,A32,A33, A34,A40,A41,A42,A49, A50}

© Los autores, 2002; © Edicions UPC, 2002.

E(a) = 1'(b) si a ∈ {A9,A10}
E(a) = 1'(p,x,m) si a ∈ {A14,A15}
E(a) = 1'(m) si a ∈ {A17}
E(a) = 1'(sa) si a ∈ {A35,A36}
E(a) = 1'(rx) si a ∈ {A43,A44}
E(a) = 1'(p,2) si a ∈ {A12}
E(a) = 1'(s) si a ∈ {A21}
E(a) = 1'(s)+1'(j) si a ∈ {24}
E(a) = if p=8 then 1'p si a = A51
E(a) = if p<>8 then 1'(p,1) si a = A7
E(a) = if ((p=3|| p=4) && x=1) then 1'(p,x) si a = A27
E(a) = if ((p=1|| p=2 || p=5)) && x=1 then 1'(p,x) si a = A28
E(a) = if p=7 && x=1 then 1'(p,x) si a = A29
E(a) = if x=2 || p=6 then 1'(p,x) si a = A31
E(a) = if p=2 then 1'(p,x) si a = A37
E(a) = if (p=3 || p=5) then 1'(p,x) si a = A38
E(a) = if p=1 then 1'(p,x) si a = A39
E(a) = if p=3 then 1'(p,x) si a = A45
E(a) = if p=1 then 1'(p,x) si a = A46
E(a) = if p=4 then 1'(p,x) si a = A47
E(a) = if ((m=j && f1()=0) || (m=s)) then 1'(m) si a = A16
E(a) = if (m=j && f1()=1) then 1'(p,x) si a = A19
E(a) = if ((m=j && f1()=0) || (m=s)) then 1'(p,x) si a = A18

donde:

1. f1() es una función probabilística que devuelve el valor 1 si el medico júnior necesita la ayuda de un médico sénior para efectuar el diagnóstico.
2. && es el operador lógico AND.
3. || es el operador lógico OR

Los aspectos que se pueden destacar del modelo son:

- La subred que contiene las transiciones T_1,T_2 y T_3 modela el proceso de llegadas y la recepción de los pacientes.
- La subred que contiene las transiciones T_{11} y T_{12} modela la espera en la sala auxiliar y el servicio en la sala.
- La subred que contiene las transiciones T_{13} y T_{14} modelas la espera y el proceso en la sala de rayos X.
- Los arcos de salida de las transiciones T_3, T_{10}, T_{12} y T_{14} contienen la expresión de control de flujo de los pacientes. El objetivo de las expresiones es forzar a que cada paciente siga su circuito definido en la transición de llegadas.

© Los autores, 2002; © Edicions UPC, 2002.

- Las transiciones T_4 y T_5 compiten para acceder al recurso *box*. Es decir, desean acceder a los *boxes* tanto los pacientes que efectúan la primera visita como los que efectúan la segunda visita, después de realizadas las pruebas que se consideran necesarias. Si las dos transiciones están activas simultáneamente, es prioritaria la transición T_5; son los que acceden al *box* en segunda vuelta.
- Si no se cumple la expresión condicional *if*, en un arco de salida de una transición, no se genera una marca en el lugar de destino del arco de salida.

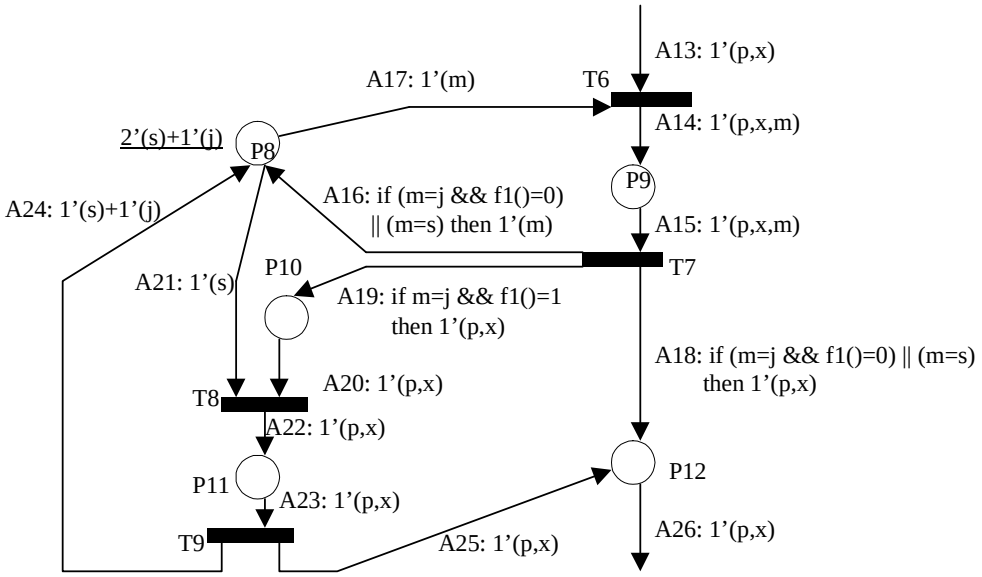


Figura 2.40 RdPC del proceso de visitas en el box

- La figura 2.40 muestra la subred del proceso de visitas en el *box*. Para que se inicie la visita es necesario que haya un médico disponible (T_6). En A14 se incorpora la información sobre el tipo de médico, dado que tanto el tiempo de proceso como el flujo posterior dependen del tipo.
- En la transición T_7, si el médico es sénior, se finaliza la visita y el médico queda libre. Si el médico es júnior se ejecuta la función probabilística f1(). Si su valor es 1, indica que es necesaria la presencia de un médico sénior para determinar o confirmar el diagnostico; en este caso, el lugar P_{10} indica la espera hasta que un médico sénior queda libre. La transición T_8 es prioritaria sobre la T_6 si las dos se pueden activar.
- En el lugar P_{11} no es necesario almacenar el tipo de médico, a diferencia del lugar P_9, dado que sabemos positivamente que en el lugar P11 hay los dos tipos de médicos.
- Al finalizar la visita con los dos médicos quedan libre los dos (arco A24: 1'(s)+1'(j)).

© Los autores, 2002; © Edicions UPC, 2002.

2.5.2 Ventajas del formalismo de RdPC

Puesto que las expresiones de arco permiten parametrizar las transiciones, éstas pueden ser fácilmente codificadas para representar una clase de eventos en lugar de un único evento como ocurría con las RdP.

Aunque aparentemente la simplificación de una RdP en una RdPC debería facilitar su mantenimiento, puesto que el objetivo que se persigue es la especificación del modelo conceptual de un sistema que sea fácilmente codificable en un entorno de simulación, es muy conveniente que los eventos formalizados en la nueva RdPC ofrezcan un nivel de abstracción similar a la descripción de eventos en el entorno de simulación en el que se codificará el modelo de simulación.

Las RdPC permiten formalizar los modelos manteniendo una analogía con la especificación del sistema según el formalismo de colas. En la figura 2.41 se ha representado el acceso a un servidor por parte de diferentes clientes, tanto en el formalismo de RdP como en el formalismo de RdPC. Nótese que en RdPC las políticas de colas pueden ser fácilmente representadas de modo explícito utilizando colores para la selección de *tokens* de los lugares de entrada (por ejemplo, añadiendo un campo contador en el color del *token* de acuerdo al orden de llegada).

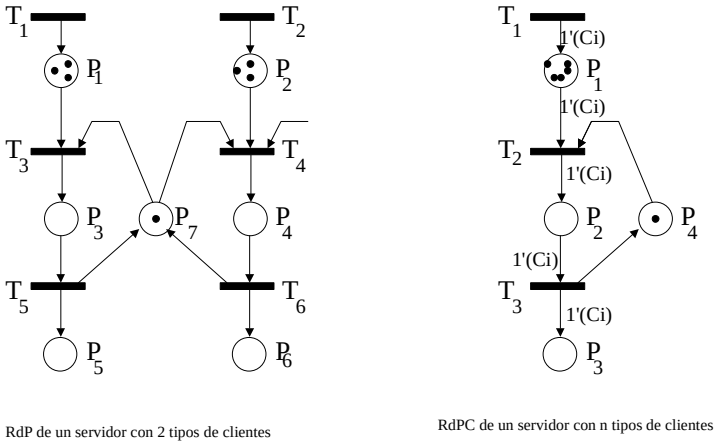


Figura 2.41 Formalización de una cola de espera en RdP y RdPC

2.6 Otras aproximaciones formales para el modelado de sistemas de eventos discretos

Existen muchos otros métodos formales para el modelado de sistemas de eventos discretos. Un estudio comparativo no es fácil y está fuera del alcance de este libro. Algunos de estos métodos pueden tener mecanismos equivalentes a los de las redes de Petri e incluso aspectos claramente mejores. En consecuencia, no nos atrevemos a afirmar que las redes de Petri sean superiores al resto de las metodologías disponibles. No obstante, creemos que las redes de Petri son muy útiles y que, en el contexto de este libro, facilitan alcanzar los objetivos perseguidos dado que:

© Los autores, 2002; © Edicions UPC, 2002.

1. Las redes de Petri representan de forma explícita los estados y los eventos del modelo.
2. Los mecanismos de concurrencia, sincronismo y dependencia causal están representados de forma natural.
3. El conjunto de recursos restringidos se representa de forma explícita en el modelo.
4. Hay muy pocas primitivas o reglas, lo que facilita su aprendizaje.
5. Su representación gráfica es muy intuitiva.
6. Su semántica es muy precisa y sin ambigüedades.
7. Es independiente de la herramienta de simulación empleada.

Otros métodos formales empleados son, por ejemplo:

- Métodos basados en DEVS: algunas aproximaciones para el modelado de sistemas de eventos discretos tienen su origen en la teoría general de sistemas. Dentro de este grupo, la metodología más desarrollada es el formalismo *Discrete Event System Specification* (DEVS) desarrollado por Zeigler [ZEIG00].

- Diagramas de actividades: el diagrama de actividades proporciona una representación gráfica, parecida a la de la red de Petri, del modelo de simulación [PAUL93]. En un diagrama de actividades típico, el modelo de simulación se ve como un conjunto de entidades que interaccionan entre ellas. Una entidad es parecida a una marca de la red de Petri, mantiene su entidad a lo largo de la simulación. Las entidades pueden estar pasivas (en colas) o activas si participan en actividades que consumen tiempo. Esta metodología de modelado es apropiada para simuladores orientados a la exploración de actividades (capítulo 4).

- Grafos de eventos: es una metodología apropiada para los simuladores orientados a eventos (programación de sucesos -capítulo 4-). Un grafo de eventos es un grafo dirigido [SCHR83] que muestra las relaciones entre eventos. Para construir el grafo de eventos, los eventos son definidos, numerados y representados como nodos en el digrafo. Los grafos son conectados por arcos que indican cómo un evento provoca la aparición de otros eventos.

En la literatura también se encuentran otros métodos diseñados para facilitar la construcción y mantenimiento del modelo. Uno de estos métodos es el IDEF0 (*Integration Definition for Function Modelling*). IDEF0 (http://www.idef.com/idef0.html) es un método diseñado para modelar las actividades, acciones y decisiones de una organización o sistema. Esta metodología de modelado presenta muchas ventajas cuando es utilizada para realizar un análisis funcional, pues facilita la interacción entre el analista y el cliente. Facilita la descomposición de un sistema complejo en un conjunto de subsistemas más sencillos mediante una aproximación *top-down*.

Gráficamente, un modelo en IDEF0 se especifica mediante un conjunto de celdas (cajas y conectores), cada una de las cuales representa una función cuya interfase queda especificada por los conectores de entrada y salida. La interacción entre bloques funcionales queda determinada por los conectores que permiten restringir cuándo y cómo son ejecutadas las funciones. En la figura siguiente se ilustra la representación gráfica de la descripción funcional de un sistema de producción.

© Los autores, 2002; © Edicions UPC, 2002.

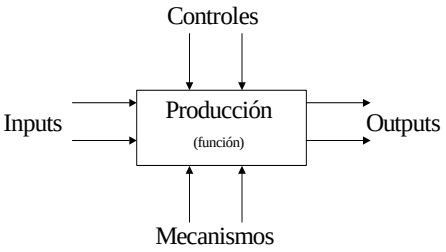


Figura 2.42 Representación gráfica de funciones en IDEF0

Cabe notar que la capacidad de estructuración jerárquica de IDEF0 permite que un bloque funcional represente una estructura de bloques funcionales más sencillos, los cuales, a su vez, pueden contener una descripción jerárquica de bloques estructurales más sencillos, con una limitación de no permitir un nivel de detalle superior a 6 subbloques funcionales para cada bloque funcional.

A pesar de las ventajas como metodología de modelado que puede ser utilizada en un análisis funcional, IDEF0 tan sólo permite la representación de actividades y sus relaciones más importantes sin considerar las relaciones temporales.

Las actividades se encuentran descritas por sus entradas, salidas, controles y mecanismos. El flujo de entradas/salidas entre bloques funcionales es una aproximación gráfica muy útil para representar secuencias de actividades, pero tal y como ya ha sido presentado, existe una gran diversidad de sistemas orientados a eventos discretos, entre los que cabe destacar los sistemas logísticos (sistemas de producción, transporte y servicios), cuyo comportamiento no puede ser descrito por una secuencia ordenada de actividades, entre otro motivos porque se trata de actividades concurrentes asíncronas con conflictos, debido a la compartición de recursos.

2.7 Ejercicios

En este apartado se presentan ejercicios relacionados con el modelado de sistemas. Adicionalmente, en el capítulo 6 encontrarás muchos problemas en los que uno de los primeros objetivos es también obtener el modelo del proceso.

1. Dada la red de Petri de la figura 2.43:

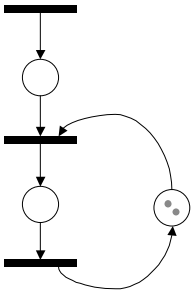


Figura 2.43 Red de Petri

© Los autores, 2002; © Edicions UPC, 2002.

a) Identifica los nodos lugar y los nodos transición de esta red.
b) Describe tres sistemas diferentes que se puedan modelar con esta redes de Petri.
c) Expresa la red de Petri según su definición.
d) Encuentra el árbol de alcance hasta el nivel 6.
e) Encuentra el árbol de cobertura.

2. Considera la RdP de la figura siguiente, con el estado inicial M_0=[1,1,0,2], donde W(T1,P2)=2. Se pide:

   a) Después de dos transiciones, encuentra un estado en el que no es posible otra transición. Es decir, encontrar las dos transiciones que conducen a un estado en el que no es posible otra transición.
   b) Supón que se aplican la siguiente secuencia de transiciones (T2,T1,T3,T1,…). Muestra que esto no es posible a partir de una transición determinada.
   c) Encuentra el estado M_s resultante del disparo de las siguientes transiciones en secuencia (T1,T2,T3,T3,T3).

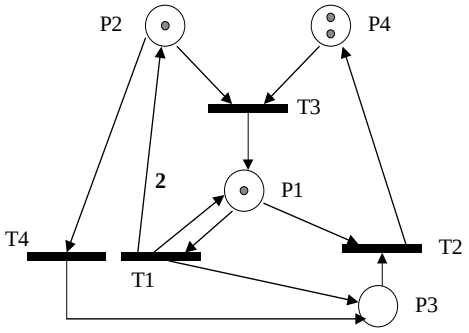


Figura 2.44 RdP

3. Considera la RdP definida por:

   - P = {P1,P2,P3}
   - T = {T1,T2,T3}
   - A = {(P1,T1),(P1,T3),(P2,T1),(P2,T2),(P3,T3),(T1,P2),(T1,P3),(T2,P3),(T3,P1),(T3,P2)}
   - Todos los arcos tienen peso 1 excepto W(P1,P1)=2

   a) Representa gráficamente la RdP.
   b) Si M_0=[1,0,1]. Muestra que para este y todos los estados posteriores nunca se puede disparar la transición T1.
   c) Si M_0=[2,1,1]. Muestra que para toda transición posterior o bien se entra en bloqueo o se vuelve al estado M_0.

4. Una librería dispone de 3 vendedores. Éstos están encargados del cobro de las compras de los clientes y de responder a las preguntas que les formulen. En la librería existen tres cajas registradoras pero solamente un ordenador para realizar consultas. Así, un cliente puede necesitar a un vendedor para pagar sus compras o para realizar una consulta. Si el cliente desea pagar, éste

© Los autores, 2002; © Edicions UPC, 2002.

se dirigirá a un vendedor para que le cobre. Si el cliente quiere realizar una consulta se dirigirá a un vendedor y este último realizará la consulta en el ordenador. Una vez realizada la consulta, si el cliente tiene que pagar alguna compra el mismo vendedor le cobrará.

a) Representa gráficamente la RdP que modela el sistema.
b) Representa gráficamente la RdPC que modela el sistema
c) Describe ventajas e inconvenientes de las dos representaciones realizadas.

5. En el ejemplo 2.14 de este capítulo se describe el modelo de un proceso hospitalario. Si se observa cuidadosamente la RdPC de la figura 2.39, se observa que, tanto el proceso de visita a la sala de rayos X como el proceso de visita a la sala auxiliar, tienen estructuras paralelas. Este paralelismo parece indicar la posibilidad de juntar ambos procesos en una sola subred. Se pide modificar la RdPC de la figura para incluir en una única subred la sala de rayos X y la sala auxiliar. Incorporar, si es necesario, nuevos conjuntos de colores para diferenciar las dos salas y en qué sala está el paciente.

6. Estudia los posibles casos de bloqueo que presenta el sistema modelado en el ejemplo 2.10:

a) Realiza el árbol de alcance.
b) Estudia las posibles soluciones de estos bloqueos.

7. El problema de los filósofos es un ejemplo clásico de un proceso que puede evolucionar hacia el bloqueo [DIJK71]. Cinco filósofos están sentados alrededor de una mesa circular tal como se muestra en la figura siguiente. Los filósofos alternan entre comer y meditar. En el centro de la mesa hay un plato de espaguetis, y entre cada par de filósofos hay un solo tenedor. Para poder comer, un filósofo necesita los dos tenedores, situados a su izquierda y su derecha. Esta restricción impide que dos filósofos vecinos puedan comer al mismo tiempo.

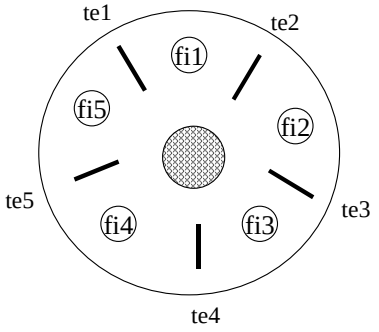


Figura 2.45 Problema de los cinco filósofos

© Los autores, 2002; © Edicions UPC, 2002.

a) Representa el sistema, en primer lugar, mediante una RdP. Se supone que cada filósofo coge/deja los dos tenedores simultáneamente. Cada filósofo puede ser representado por dos estados (comer, meditar) y dos transiciones (tomar y dejar los tenedores). Cada tenedor puede ser representado por un solo lugar, que tiene una marca cuando el tenedor está libre.

b) Describe el mismo sistema empleando una RdPC que contenga dos conjuntos de colores FI={fi_1,fi_2,...,fi_5} y TE = {te_1,te_2,...,te_5}, que representen a los filósofos y a los tenedores respectivamente. Para construir la red sólo es necesario emplear tres lugares (meditar, comer y tenedores no usados) y dos transiciones (tomar tenedores, dejar tenedores).

c) Modifica la RdP y la RdPC anteriores de forma que cada filósofo primero tome su tenedor derecho, y a continuación el izquierdo cuando esté disponible. Si no lo está, se queda a la espera (sin meditar) a que el cubierto que le falta esté libre.

d) ¿Cómo quedarán modificadas la RdP y la RdPC anteriores si los filósofos toman los dos tenedores uno detrás del otro pero en un orden arbitrario?.

e) Analiza si las RdPC de los apartados 2 y 3 anteriores pueden alcanzar una situación de bloqueo. Emplea para ello el árbol de cobertura y la propia red. El estado inicial M_0 corresponde a todos los filósofos cuando están meditando.

8. Dos carros A y B transportan un determinado material desde los puntos de carga C_a y C_b, respectivamente, hasta el punto de descarga D (figura 2.46). Los movimientos de los dos carros, hacia la izquierda o hacia la derecha, son controlados mediante las acciones i_a (izquierda a), i_b, d_a (derecha a) y d_b. Los dos pulsadores M_a y M_b controlan el inicio del ciclo de los dos carros. Si A se encuentra en C_a y M_a es pulsado, se inicia el ciclo C_a-D-C_a con las siguientes características:

- Espera eventual en E_a hasta que el tramo común G-D a los carros esté libre, con el objetivo de evitar colisiones.
- Espera obligada en D para la descarga del material.

El carro B tiene un funcionamiento similar (pulsador M_b, ciclo C_b-D-C_b y tiempo de descarga de material en D). El recorrido E_a-D y el recorrido E_b-D se establece gracias al posicionamiento de un cambio de agujas automático controlado por la posición *G o* $\overline{G}$ respectivamente. Obtener el modelo en RdP del sistema físico presentado en este ejercicio.

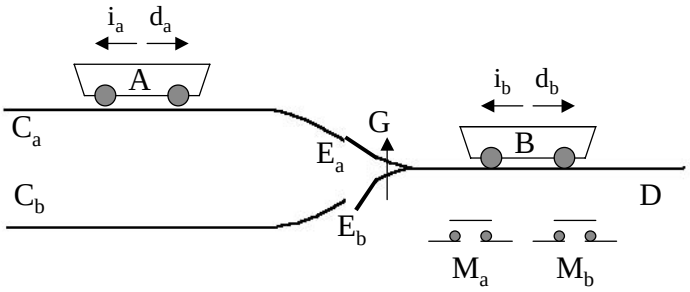


Figura 2.46 Sistema de transporte de dos carros con camino común.

© Los autores, 2002; © Edicions UPC, 2002.

9. Representa gráficamente la RdPC especificada por:
   - $\Sigma = \{x,y,z,R,S\}$
   - $P = \{P_1, P_2, P_3, P_4, P_5\}$
   - $T = \{T_1,T_2, T_3, T_4, T_5\}$
   - $A = \{A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8, A_9, A_{10}, A_{11}, A_{12}, A_{13}, A_{14}, A_{15}, A_{16}, A_{17}, A_{18}\}$
   - $N(A_1)=(T_1,P_2)$; $N(A_2)=(P_1,T_2)$; $N(A_3)=(P_1,T_3)$; $N(A_4)=(P_1,T_4)$; $N(A_5)=(P_2,T_2)$; $N(A_6)=(P_2,T_3)$; $N(A_7)=(P_2,T_4)$; $N(A_8)=(P_3,T_3)$; $N(A_9)=(P_3,T_4)$; $N(A_{10})=(T_2,P_1)$; $N(A_{11})=(T_2,P_5)$; $N(A_{12})=(T_3,P_4)$; $N(A_{13})=(T_3,P_1)$; $N(A_{14})=(T_4,P_4)$; $N(A_{15})=(T_4,P_1)$; $N(A_{16})=(P_4,T_5)$; $N(A_{17})=(T_5,P_5)$; $N(A_{12})=(T_5,P_3)$;
   - $G(T_1)$=true; $G(T_2)$=true; $G(T_3)=y\neq z$; $G(T_4)=y=z$; $G(T_5)$=true;
   - $I(P_1)=1'(14,1)+1'(3,2)+1'(3,3)$; $I(P_3)=14'(1)+3'(2)+3'(3)$;
   - $C(P_1)=S$; $C(P_2)=z$; $C(P_3)=y$; $C(P_4)=R$; $C(P_5)=z$;
   - $E(A_1)=1'(z)$; $E(A_2)$=if (x=0) then 3'(x,y); $E(A_3)$=if (y=1)&&(x>0) then 1'(x,y); $E(A_4)=1'(x,y)$; $E(A_5)=1'(z)$; $E(A_6)=1'(z)$; $E(A_7)=1'(z)$; $E(A_8)=1'(y)$; $E(A_9)=1'(y)$; $E(A_{10})=3'(x,y)$; $E(A_{11})=1'(z)$; $E(A_{12})=1'(y,z)$; $E(A_{13})=1'(x-1,y)$; $E(A_{14})=1'(y,z)$; $E(A_{15})=1'(x-1,y)$; $E(A_{16})=1'(y,z)$; $E(A_{17})=1'(z)$; $E(A_5)=1'(y)$;

10. Un pequeño supermercado situado en el centro de una ciudad dispone de un aparcamiento de 20 plazas. De estas 20 plazas, 3 están reservadas para vehículos de personas minusválidas y 3 para trabajadores del supermercado. Las otras 14 plazas restantes están a disposición de los clientes. Los usuarios del aparcamiento ocupan las plazas acordes a su categoría según llegan al sistema. En el caso que las plazas de minusválidos o de trabajadores estén ocupadas, también pueden ocupar plazas de clientes normales. Si un vehículo llega al aparcamiento y todas las plazas están ocupadas no se espera a que queden plazas libres y marcha del aparcamiento.

    a) La RdPC del ejercicio 9 representa este sistema. Estudia su comportamiento si:
       - $\Sigma = \{x,y,z,R,S\}$
         x: número de plazas disponibles
         y: tipo de plaza
         z: tipo de vehículo
         R: producto cartesiano de $y \times z$
         S: producto cartesiano de $x \times y$
       - $P = \{P_1, P_2, P_3, P_4, P_5\}$
         P1: estado de las plazas
         P2: automóvil buscando plaza
         P3: plazas disponibles
         P4: coche aparcado (cliente comprando en el supermercado)
         P5: coche que sale del aparcamiento
       - $T = \{T_1,T_2, T_3, T_4, T_5\}$
         T1: llegada de un vehículo
         T2: salida del aparcamiento porque está totalmente lleno
         T3: ocupa plaza normal porque las reservadas están llenas
         T4: ocupa una plaza según el tipo de vehículo
         T5: salida después de la compra

© Los autores, 2002; © Edicions UPC, 2002.

b) Un problema que presenta esta red es que, inicialmente, si hay plazas libres para trabajadores o minusválidos, éstos no pueden ocupar una plaza normal hasta que sus plazas estén totalmente ocupadas. Modifica la red para que estos colectivos puedan aparcar libremente.
c) Otro problema que presenta el sistema tal como está diseñada es que los trabajadores pueden ocupar más plazas de las que tienen asignadas. Se pide restringir el modelo para que si las plazas de trabajadores están ocupadas no se ocupen plazas para clientes normales.

11. La figura siguiente muestra un proceso de fabricación.

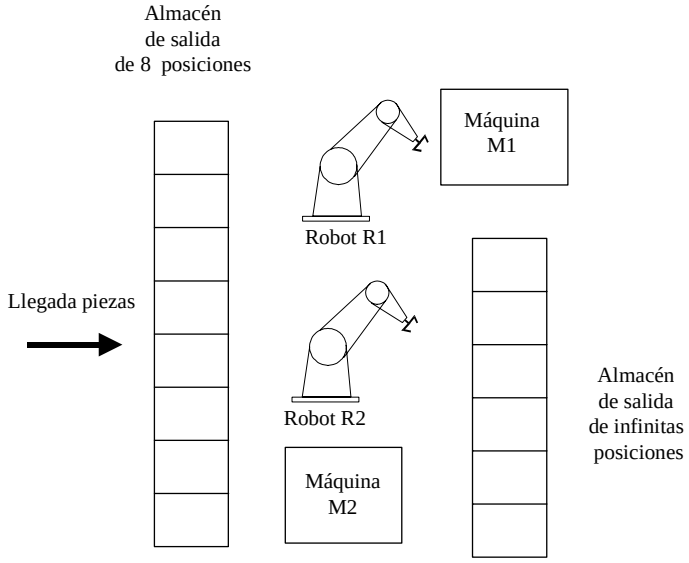


Figura 2.47 Proceso de fabricación

Las piezas que entran son depositadas directamente en el almacén de entrada, el primer robot traslada la pieza a una de las dos máquinas M1 disponibles para efectuar la primera operación. A continuación el robot R1 devuelve la pieza procesada al almacén de entrada. El robot R2 toma la pieza procesada por la máquina M1 y que está guardada en el almacén de entrada, y la traslada a una de las dos máquinas tipo 2. Finalmente, este mismo robot traslada la pieza procesada en la máquina 2 al almacén de salida. Hay 2 máquinas tipo 1 y 2 máquinas tipo 2. Sólo hay un robot de cada tipo. Modela el comportamiento del sistema mediante la RdP. La RdP tiene que estar protegida para evitar su bloqueo.

12. Se desea obtener el RdPC del proceso productivo mostrado en la figura 2.48. El procedimiento de producción es:
    - Se producen dos tipos de productos.
    - Las líneas de producción son dedicadas a cada producto exceptuando el recurso compartido R3.

© Los autores, 2002; © Edicions UPC, 2002.

- Para fabricar el producto 1 el robot R1 toma una pieza del almacén A1 y la carga en la máquina M1. Entonces la máquina M1 empieza a procesar. Una vez finalizado el proceso, el robot M3 coge la pieza de la máquina M1 y la deposita en el almacén A3. El producto 2 sigue el mismo esquema productivo empleando el almacén A2, la máquina M2, el robot R3 y el almacén A4.

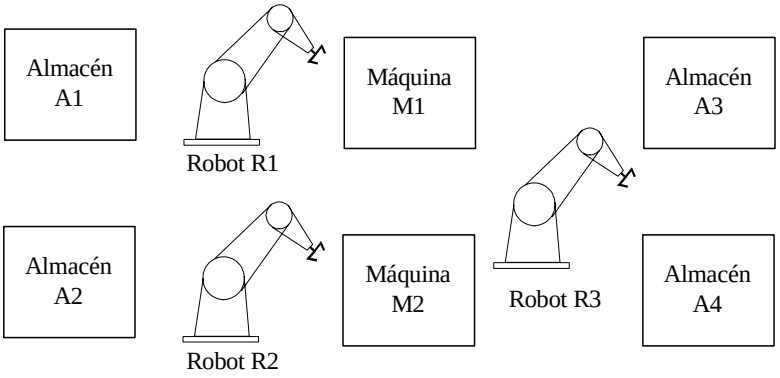


Figura 2.48 Planta de producción con un recurso compartido.

13. La planta que se muestra en la figura 2.49 sólo fabrica un tipo de producto, el cual está formado por dos piezas iniciales (materia prima). Se necesita el procesado previo en dos máquinas antes de la operación de ensamblado para obtener el producto final. Se pide obtener la RdP y la RdPC del proceso. Para realizar este ejercicio se supondrá que:

- Hay infinitas piezas en los almacenes de entrada A1 y A2 .
- El almacén A3 tiene capacidad infinita.
- La cintas C1 y C2 que transportan las piezas de A1 a M1 y de A2 a M2, respectivamente, tienen capacidad limitada CC1 y CC2.

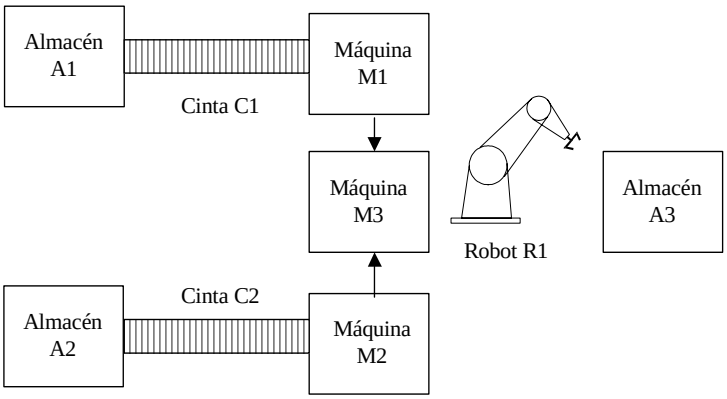


Figura 2.49 Proceso de ensamblado

© Los autores, 2002; © Edicions UPC, 2002.

El proceso de producción es el siguiente:

1) La cinta transporta las piezas a las máquinas respectivas.
2) Cada máquina sólo procesa una pieza a la vez.
3) Para poder iniciarse el proceso de ensamblado el robot debe cargar en primer lugar la pieza de M1 en M3 y en segundo lugar la de M2 en M3.
4) M3 ensambla las piezas.
5) Una vez finalizado el ensamblado el robot R1 traslada el producto final a A3.

14. En la figura 2.50 se ha representado una célula de fabricación flexible compuesta por 3 máquinas de control numérico, cada una de las cuales tiene un almacén local con capacidad máxima de 3 piezas. La cinta transportadora es la encargada de alimentar a las máquinas con las piezas que se quiere procesar, de tal manera que si la máquina M1 tiene espacio en su almacén, el manipulador asociado a la máquina coge la pieza de la cinta y la deposita en el almacén local M1. Si no hay espacio en M1, la tomará el manipulador M2 si hay espacio en su almacén. Finalmente, si no hay espacio ni en los almacenes de M1 y M2, se intenta la operación con M3. Por último, si no hay espacio en ninguno de los almacenes locales, la pieza sale del sistema y se dirige a otra célula de fabricación. Cuando las piezas han sido procesadas en M1, M2 o M3, se retiran automáticamente sobre el almacén de salida. Obtener para este sistema la RdP y la RdPC.

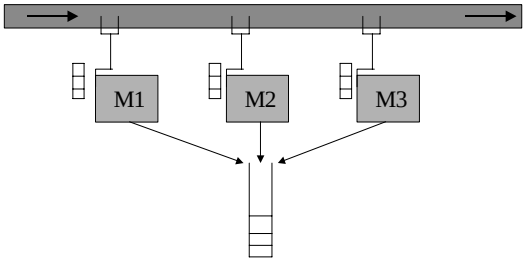


Figura 2.50 Célula de fabricación flexible.

15. La figura 2.51 muestra un esquema de una célula de fabricación flexible [ZHOU93] formada por los componentes siguientes:
    - *Sistema de transporte*: compuesto por dos mesas de transporte, TT1 y TT2, y dos cintas transportadoras que trasladan los carros desde una estación de trabajo a otra en el sentido de las agujas del reloj. Hay cuatro carros, A, B, C y D, para el transporte de piezas y cada carro transporta una pieza.
    - *Máquina fresadora*: efectúa una operación previamente programada en un entorno CAD.
    - *Robot R1*: es un recurso compartido que se emplea para cargar el material a la fresadora desde la cinta o descargar la fresadora sobre la cinta. También se emplea para trasladar el material desde la estación de entrada/salida a la cinta transportadora o viceversa.

© Los autores, 2002; © Edicions UPC, 2002.

- *Estación de entrada/salida*: alimenta con materia prima a la célula. Y descarga sobre dos almacenes de salida las piezas procesadas. En el almacén A1 se guardan las piezas procesadas correctamente y en el A2 las defectuosas.
- *Sistema de visión por ordenador*: permite la inspección automática de las piezas procesadas.
- *Estación de perforación*: se encarga de hacer un agujero en cada una de las esquinas de la pieza rectangular.

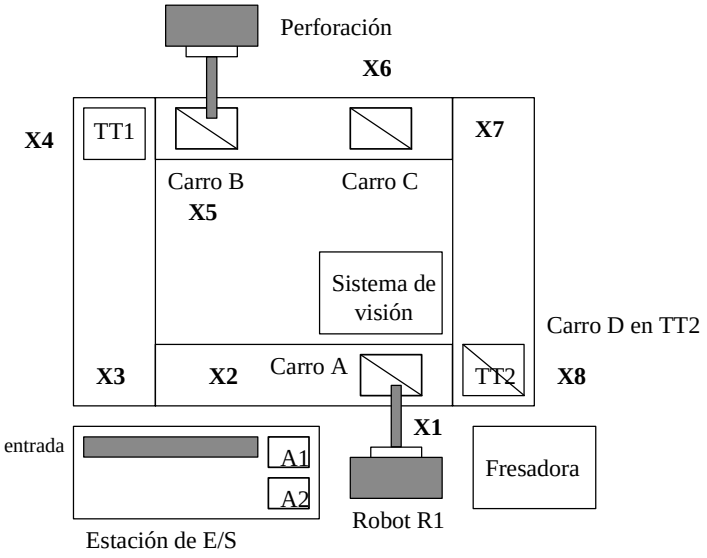


Figura 2.51 Célula de fabricación flexible.

La planta se controla con un PLC (controlador lógico programable). El procedimiento de producción se muestra en los siguientes pasos. Los pasos 1 y 2 corresponden a la operación de carga inicial de los carros y, los pasos 3 al 13 describen el proceso de fabricación de las piezas:

1) Inicialmente, los cuatro carros que se encuentran en el sistema de transporte están vacíos y disponibles para cargar material desde la estación de entrada/salida.
2) El robot R1 carga una pieza en cada carro. Los carros se mueven en el sentido de las agujas del reloj a medida que se van cargando.
3) Una vez se han cargado todas las piezas, se supone que la posición de los carros y de las mesas es la mostrada en la figura. El carro B, que está en la posición X5, tiene una pieza a punto para ser perforada. El carro C está en la posición X6 pendiente de que la mesa TT2 descargue el carro sobre la posición X1 y se desplace a la posición X7 para cargarlo (el C) y desplazarlo a la posición X1. El carro D está sobre la mesa TT2 en la posición X8, a la espera de trasladarse a la posición X1 tan pronto como quede libre. El carro A está en la posición X1 con una pieza a punto para ser fresada, y posteriormente inspeccionada antes de ser guardada en uno de los dos almacenes de salida.
4) La estación de perforación procesa la pieza que está en el carro tan pronto como el carro para en la posición X5.
5) El robot R1 coge la pieza del carro que está en la posición X1 y la carga sobre la fresadora.

© Los autores, 2002; © Edicions UPC, 2002.

6) Una vez el robot R1 ha descargado la pieza, éste retrocede y la fresadora inicia el procesado.
7) Después de finalizarse la operación de fresado, el robot se desplaza hasta la fresadora para descargar la pieza.
8) La pieza es devuelta al mismo carro donde estaba anteriormente, en la posición X1.
9) Se envía una señal a la cámara para que inspeccione la pieza.
10) El sistema de visión devuelve una señal que indica si se acepta o rechaza la pieza.
11) El robot traslada la pieza al almacén A1 o A2 en función de si ha sido aceptada o rechazada.
12) El robot se desplaza a la estación de entrada (piezas de entrada) y carga una nueva pieza sobre el carro que todavía está en la posición X1.
13) El carro se libera y se inicia otra vez el ciclo.

Se supone, para realizar el ejercicio, que los carros ya están cargados en las posiciones indicadas en la figura. Por tanto, no se tienen en cuenta los dos pasos iniciales. Se pide determinar la RdP de cada uno de los subsistemas siguientes:

a) El sistema de transporte. Cintas transportadoras y mesas. Para ello hay que tener en cuenta que hay solo 8 posiciones posibles: X1, X2,... X8.
b) La máquina fresadora.
c) La estación de perforación.
d) Juntar los subsistemas anteriores y encontrar la RdP de todo el sistema.

16. Últimamente se han comercializado en el mercado juguetes electrónicos llamados “mascotas virtuales”. Estos juguetes imitan el comportamiento de mascotas domésticas. Estos sistemas se basan en reproducir los comportamientos de las mascotas y pedir al usuario/jugador una interacción con ellas. El objetivo de este juego es mantener la mascota viva el máximo de tiempo. Si la interacción es la correcta, por ejemplo, si la mascota reclama comida, el usuario proporciona alimento, si la mascota quiere jugar, el usuario juega con ella, etc., la mascota será longeva. Si la interacción no es la adecuada, por ejemplo, el jugador nunca baña a la mascota, no juega con ella, etc., la mascota enfermará y morirá. Un subconjunto de las reglas de juego puede ser:

- La mascota está por defecto en un estado de reposo.
- En el estado de reposo la mascota puede pedir al jugador que interaccione con ella.
- Las peticiones que realiza la mascota al jugador son: comida, higiene, juegos y mimos.
- Cuando la mascota se considere cansada puede dormirse sin necesitar una interacción con el jugador.
- El jugador puede realizar las siguientes acciones con la mascota: alimentarla, lavarla, jugar con ella, acariciarla y despertarla. El jugador también puede interaccionar con la mascota sin que ella lo pida.
- Si el jugador no interacciona con la mascota después de sus peticiones, ésta volverá al estado de reposo.
- Si la mascota está dormida y se la despierta vuelve al estado de reposo.

a) Realiza la RdPC que modele el sistema mascota/jugador e incluye un color que indique el estado de ánimo de la mascota. Los estados de ánimo posibles son: contenta, normal, agobiada y enfadada. Las reglas de comportamiento son:

- La mascota está inicialmente con un humor normal.

© Los autores, 2002; © Edicions UPC, 2002.

- Si en humor normal el usuario interacciona con ella a causa de una petición, la mascota estará contenta.
- Si en humor normal el usuario no interacciona con ella cuando lo pide, la mascota se enfadará.
- Si en humor normal el usuario interacciona con ella por iniciativa propia la mascota se pondrá contenta.
- Si en humor contento la mascota pide interacción y el usuario interacciona con ella continuará contenta.
- Si en humor contento la mascota pide interacción y el usuario no interacciona con ella la mascota volverá a tener un estado de humor normal.
- Si en humor contento el usuario interacciona sin que la mascota lo pida, ésta se agobiará.
- Si la mascota pide interacción estando agobiada y el usuario interacciona con ella ésta se pondrá contenta.
- Si la mascota pide interacción estando agobiada y el usuario no interacciona con ella ésta se enfadará.
- Si el usuario interacciona con la mascota agobiada y ella no lo ha pedido, morirá debido al agobio.
- Si la mascota está enfadada y pide interacción, se morirá por falta de atención si el usuario no interacciona.
- Si la mascota está enfadada, pide interacción y el usuario responde, el estado de humor será normal.
- Independientemente del estado de humor de la mascota, si ésta es despertada se enfadará.

b) Modifica la RdPC anterior para que la mascota sólo muera si se producen tres interacciones negativas consecutivas estando agobiada o enfadada.

17. La figura 2.52 muestra un esquema de una célula de fabricación flexible (adaptado de [JENG97]) formada por tres maquinas iguales y un *pool* de herramientas (dos herramientas de tipo H1 y dos de tipo H2).

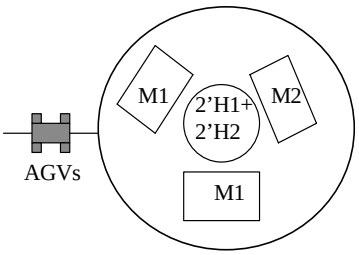


Figura 2.52 Célula de fabricación flexible.

Hay dos tipos de piezas (PZ1 y PZ2) las cuales precisan de dos operaciones cada una. Las dos operaciones se realizan en la misma máquina y para cada operación es necesaria una herramienta diferente, tal como se muestra en la siguiente tabla. También se muestra en la tabla la media y la variancia de la distribución normal de los tiempos conjuntos de carga y proceso.

© Los autores, 2002; © Edicions UPC, 2002.

	Operación 1	Operación 2
Pieza 1	H1, N(20,4)	H2, N(17,5)
Pieza 2	H2, N(15,2)	H1, N(21,8)

Tabla 2.2 Secuencia de operaciones

La figura 2.53 muestra la RdP del proceso donde:

- P1: número de máquinas libres
- P2: número de herramientas H1 libres
- P3: número de herramientas H2 libres
- P4: carga de una pieza tipo 1, captura de una herramienta H1 y primera operación
- P5: captura de una herramienta H2. Segunda operación sobre la pieza tipo 1, y descarga de la pieza y devolución de las herramientas H1 y H2
- P6 y P7: equivalente a las P4 y P5 pero el orden de captura de las herramientas es inverso

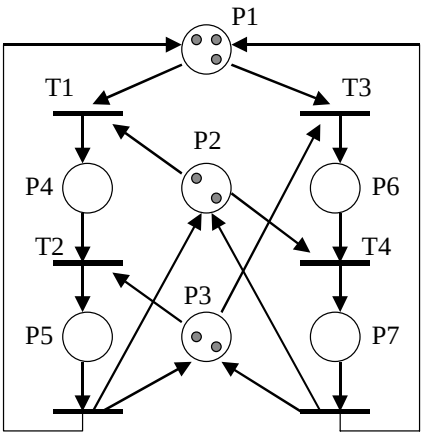


Figura 2.53 RdP de la célula de fabricación flexible.

a) Analiza si el proceso se puede bloquear, a partir del árbol de cobertura.
b) Simula la RdP varias veces durante un periodo de 8 horas empleando una herramienta de simulación y calcula la probabilidad de que el sistema se pueda bloquear.

© Los autores, 2002; © Edicions UPC, 2002.

3 Modelos estadísticos en simulación

Una característica fundamental que permite clasificar los modelos de simulación en deterministas o estocásticos es el uso o no de variables aleatorias para formalizar la evolución del sistema que se representa. En este capítulo se introduce el uso de modelos de simulación estocásticos para representar aquellas actividades cuyos efectos no pueden ser descritas completamente en términos de las entradas y del estado actual del sistema (procesos estocásticos), y también para formalizar actividades propiamente deterministas.[2]

Considérese el sistema ilustrado en la figura 3.1. Se trata de un manipulador que debe recoger las piezas de una caja, y almacenarlas en distintos *stocks* en función del tipo de pieza de que se trata. Desde un punto de vista tecnológico, la grapa del manipulador dispone de un conjunto de sensores cuyas señales van a permitir determinar la posición y orientación de la pieza, información necesaria para que el manipulador pueda recogerla y clasificarla en un cierto *stock*. El tiempo necesario para finalizar un ciclo de trabajo dependerá de distintos factores, tales como: posición y orientación de la siguiente pieza que debe ser procesada respecto a la grapa del manipulador, posición final a la que debe ser transportada, peso de la pieza, etc. Nótese que el conocimiento de estos datos, junto con el conjunto de ecuaciones algebraico-diferenciales que describen la dinámica del manipulador, así como la orientación y posición de la siguiente pieza, deberían ser suficientes para poder predecir con exactitud (sistema determinista) el tiempo que se invertirá en la siguiente operación de transporte. Así pues, es posible desarrollar un modelo de simulación determinista que permita generar en un ordenador las mismas salidas y obtener los mismos cambios de estado que se obtendrían experimentando con el sistema real.

De modo análogo, y recordando que el modelo de un sistema es cualquier representación *lo más simplificada posible* de las dinámicas de interés, tanto el tiempo invertido en cada una de las operaciones de transporte, como los instantes de tiempo en los cuales el robot finalizará la operación en curso (cambios de estado), pueden ser considerados el resultado de una actividad aleatoria (a pesar de ser determinista). Así pues, observando (a partir de datos históricos) la frecuencia de tiempos que el manipulador ha necesitado para finalizar las distintas operaciones de transporte es posible, mediante el

[2] Cuando es posible describir completamente el resultado de una actividad en términos de su entrada, se dice que la actividad es determinista. Cuando los efectos de la actividad varían aleatoriamente en distintas salidas, se dice que la actividad es estocástica.

© Los autores, 2002; © Edicions UPC, 2002.

uso de variables aleatorias, generar una secuencia de datos que presenten las mismas propiedades (frecuencia de aparición) que los datos observados del sistema real.

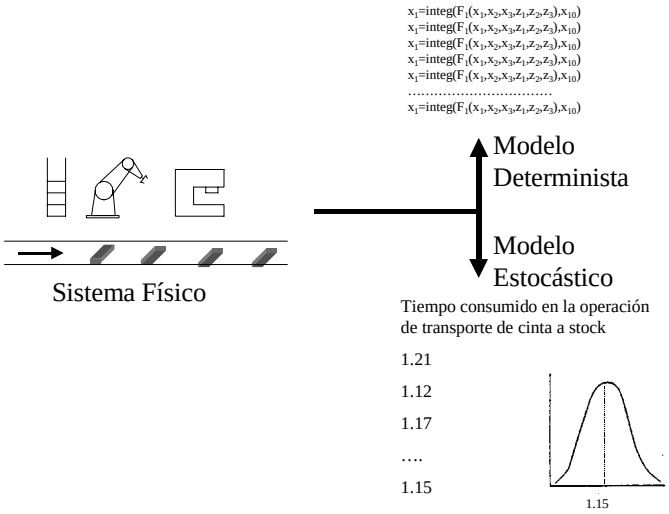


Figura 3.1 Modelo estocástico y modelo determinista de un sistema físico

Aunque las unidades de producción, así como los manipuladores y unidades de transporte que pueden encontrarse en las líneas de producción, estén lejos (afortunadamente) de presentar un comportamiento aleatorio, el uso de modelos estocásticos para representar la secuencia de eventos durante los cuales un sistema puede presentar cambios de estado, facilita la tarea de modelado y también la fase de validación del modelo. El uso de variables aleatorias en la formalización de la dinámica de un sistema simplifica la tarea de modelado, pues permite replicar su comportamiento sin tener que formalizar todas aquellas relaciones causa-efecto que provocan variaciones en la salida respecto a un valor promedio.

Al margen de los sistemas deterministas, también existe un conjunto nada despreciable de dinámicas cuya descripción mediante formulaciones deterministas es desaconsejable, y es prácticamente imposible formalizarlas pues aparecen como efecto de una actividad aleatoria. Dentro de este conjunto de procesos *estocásticos* se encuentran, por ejemplo, el tiempo de funcionamiento de una máquina entre avería y avería, su tiempo de reparación y el tiempo que necesita un operador humano para realizar una determinada operación.

Para llegar a modelar correctamente un proceso estocástico es necesario comprender numerosos conceptos de probabilidad y estadística. En este capítulo se introducen brevemente aunque en ningún caso se pretende ofrecer un desarrollo detallado de los mismos.

© Los autores, 2002; © Edicions UPC, 2002.

EJEMPLO 3.1 Estación de trabajo

La RdP de la figura 3.2 representa un proceso simple constituido, básicamente, por una estación de trabajo con dos máquinas, descrito en el ejemplo 2.1. Las piezas que llegan para ser procesadas pueden esperar en una única cola FIFO (*first-in-first-out*). Finalmente, abandonan el proceso.

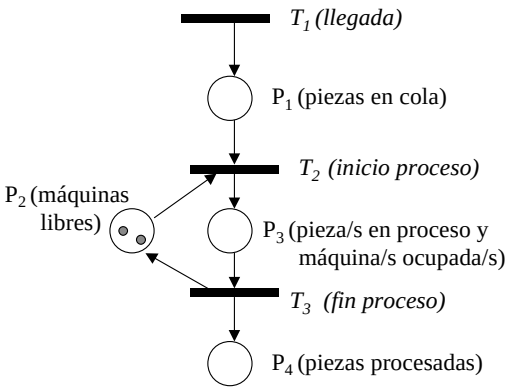


Figura 3.2 RdP de la estación de trabajo

En el capítulo anterior se han introducido las redes de Petri para representar formalmente los modelos. Las redes de Petri representan de forma gráfica las relaciones causales entre eventos, y entre eventos y actividades. Para que estos modelos puedan ser simulados es necesario parametrizarlos especificando el valor de las variables de interés. Para los modelos de eventos discretos muchas de estas variables son aleatorias. Éstas nos permiten especificar el tiempo entre llegadas (asociado a la transición T1) y el tiempo de duración del procesado de las piezas (acotado por las transiciones T2 y T3).

Las redes de Petri no contemplan de forma explícita el tiempo consumido por los elementos que recorren la red. Se supone, en general, que el tiempo se consume en los lugares, ya sea en la cola a la espera de que un recurso esté libre (lugar P_1), o cuando la pieza está siendo procesada en la máquina (lugar P_3). En estos casos, las transiciones representan los eventos asociados a las llegadas de entidades, el inicio y el final de actividades.

3.1 Definiciones

En esta sección se presentan algunos conceptos básicos necesarios para la construcción y análisis del modelo de un proceso aleatorio:

- *Experimento*: es un proceso cuyo resultado no se conoce con certeza. El conjunto de posibles resultados de un experimento se denomina el *espacio de muestreo* S. Un resultado posible del experimento se denomina *punto de muestreo* en el espacio de muestreo.
- *Variable aleatoria:* X es una variable aleatoria si puede tener asignado cualquier valor (no previsible) de un rango finito (variable aleatoria discreta) o infinito (variable aleatoria continua) de posibles valores. Más formalmente, una variable aleatoria es una función que asigna un

© Los autores, 2002; © Edicions UPC, 2002.

número a cada posible resultado del experimento (espacio de muestreo). Aunque la secuencia exacta de valores que serán asignados a una variable aleatoria no puede ser prevista, sí que es posible conocer el rango de valores en los que puede variar, así como la probabilidad de tener asignado un cierto valor.

- *Proceso estocástico*: es un proceso que evoluciona en el tiempo y/o espacio, y que involucra una variable aleatoria, de tal modo que el comportamiento del proceso no puede preverse con exactitud.
- *Distribución de probabilidad*: permite relacionar un conjunto de valores o medidas con su frecuencia relativa de aparición.
- *Función de densidad de probabilidad f(x):* describe la probabilidad que una variable aleatoria X asuma un cierto valor x_i:

$$f(x_i) = P(X=x_i)$$

- *Función de distribución acumulativa F(x):* describe la probabilidad de que una variable aleatoria X asuma un valor más pequeño o igual que un cierto valor x_i:

$$F(x_i) = P(X \leq x_i)$$

- *Estimación de la media*: dado un conjunto de variables aleatorias X_1, X_2, ..., X_n (muestra), la media de la muestra es

$$\overline{X}(n) = \frac{\sum_{i=1}^{n} X_i}{n}$$

- *Estimación de la variancia*: dado un conjunto de variables aleatorias X_1, X_2, ..., X_n (muestra), la variancia de la muestra es

$$S^2(n) = \frac{\sum_{i=1}^{n} \left[X_i - \overline{X}(n)\right]^2}{n-1}$$

EJEMPLO 3.2 Distribución de probabilidad discreta

Considerar el experimento de tirar un dado. El espacio de muestreo es S={1,2,3,4,5,6}. Se puede definir X como la variable aleatoria discreta que representa el número obtenido después de tirar el dado. Si se supone que la probabilidad de obtener una cara determinada es proporcional a su valor, la distribución de probabilidad discreta para este experimento aleatorio es

x_i	1	2	3	4	5	6
$f(x_i)$	1/21	2/21	3/21	4/21	5/21	6/21

© Los autores, 2002; © Edicions UPC, 2002.

3.2 Descripción de un sistema con características estocásticas

Las variables de interés de un sistema físico que no se pueden prever con exactitud se modelarán como procesos estocásticos, es decir, definiendo el rango de valores que puede asumir la variable aleatoria asociada al proceso estocástico, así como la probabilidad de asumir cada uno de los valores posibles.

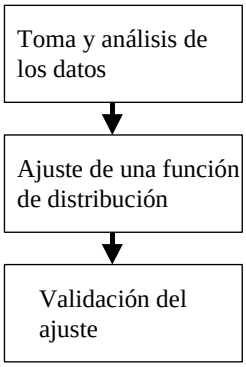


Figura 3.3 Desarrollo de un modelo estadístico

En la figura 3.3 se presenta la secuencia de etapas que deberían realizarse en la obtención del modelo estadístico que caracterice un cierto tipo de proceso aleatorio:

- Toma y análisis de los datos: la obtención de datos de calidad es una de las etapas más importantes y difíciles en el proceso de creación de un modelo estadístico. En la práctica, la validez del modelo estadístico está limitada por la calidad de los datos empleados.
- Ajuste de una función de distribución: el objetivo de esta etapa es obtener una función de distribución (función de densidad de probabilidad o función de distribución acumulativa) a partir de los datos recogidos.
- Validación del ajuste: existe un conjunto de tests estadísticos que permite determinar la validez del ajuste efectuado en la etapa anterior. En la práctica, especialmente si el ajuste no es bueno, el conocimiento que se tiene sobre el proceso de interés tiene que poder facilitar la elección de la distribución de probabilidad más adecuada.

3.3 Adquisición y análisis de datos

La recogida de datos es una de las etapas más laboriosas en la construcción de un modelo estocástico [VINC98]. En general, y de modo muy especial en los sistemas complejos, el conocimiento sobre el funcionamiento del sistema necesario para construir el modelo está, en una empresa, distribuido entre diferentes personas. Para obtener toda esta información el modelador tiene que estar dispuesto a determinar y buscar la información necesaria y las fuentes que puedan proporcionarla [CARS86].

El proceso de recogida y análisis de la información puede ser abrumador. Los datos no se presentan en general en el formato necesario para definir el modelo. Muchos esfuerzos acumulan muchos datos

© Los autores, 2002; © Edicions UPC, 2002.

pero proporcionan poca información útil para la construcción del modelo. Si la información obtenida es insuficiente deberán efectuarse hipótesis razonables para proseguir el estudio.

En general, se recomienda cuestionar siempre toda la información y datos disponibles. ¿cuál es la fuente?, ¿cuándo se recogió?, ¿cómo fue recogida?, ¿tiene ésta sentido?, ¿tenemos insuficientes datos o son excesivos? Para obtener buenos resultados es condición indispensable trabajar con buenos datos. En muchos casos no se dispone, desgraciadamente, de toda la información y datos necesarios. Aun así, se requiere dar una respuesta a las preguntas planteadas. En estos casos debemos efectuar hipótesis razonables en colaboración con el usuario final. Si los datos son limitados o su calidad es dudosa es conveniente ser prudente a la hora de extraer conclusiones. No obstante, aun en los casos en los que hay problemas con los datos, dichos resultados y el conocimiento adquirido durante el estudio de simulación siempre podrán aportar información valiosa para la toma de decisiones.

En el modelado de un componente aleatorio de un sistema, tan sólo deben registrarse los datos referentes al proceso estocástico, sin considerar ni las causas de la actividad aleatoria, ni sus consecuencias. Como ejemplo, en la tabla 3.1 se ha representado el tiempo (en minutos) entre dos llegadas sucesivas de materia prima a una unidad de producción en una línea de fabricación.

0.50	3.35	20.85	7.81	0.44	0.03	3.82	7.09	3.02	2.80
2.08	6.53	52.53	10.23	0.76	0.00	28.21	15.51	4.86	10.41
5.25	11.67	46.23	28.06	6.05	4.82	46.36	2.90	5.47	17.42
7.20	41.15	9.54	4.88	19.10	9.17	0.83	7.43	9.98	4.11
10.28	23.44	6.19	2.39	7.57	12.97	12.62	7.65	18.49	6.95
1.08	9.89	5.49	2.16	14.18	11.89	12.73	0.51	14.61	27.01
1.91	18.77	4.98	6.41	1.45	1.71	5.21	2.89	8.38	3.50
2.86	17.60	4.89	11.74	15.31	36.64	3.62	21.78	2.15	6.70
17.13	0.11	17.58	1.30	2.44	9.59	1.74	5.02	6.46	18.76
1.49	7.92	4.03	3.13	1.67	23.31	3.13	9.35	0.10	0.51

Tabla 3.1 Muestra del tiempo entre dos llegadas sucesivas

En general, se supone que los valores aleatorios de la muestra de datos son independientes y están idénticamente distribuidos (IID). Tenemos que ser conscientes de esta presunción para verificar que se cumple razonablemente para el caso de interés y, en caso contrario, considerar hipótesis alternativas. Las técnicas para validación de los ajustes que se presentan más adelante presuponen que las variables aleatorias son IID. La hipótesis que una muestra de datos es IID implica:

1. Todos los valores aleatorios de la muestra son independientes entre sí. Es decir, el conjunto de valores no está correlacionado. Supongamos que las variables X_1, X_2,..., X_n representan el tiempo que se tarda en realizar una operación manual con importantes exigencias físicas. Es de esperar que el tiempo que se tarde en efectuar la operación se incremente debido a la fatiga a medida que avanza el turno de 8 horas. Existen dos técnicas para evaluar la independencia de los valores: las gráficas de autocorrelación y los diagramas dispersión.

2. Todos los valores aleatorios de la muestra siguen la misma distribución de probabilidad y, por tanto, están idénticamente distribuidos. En procesos no estacionarios, los parámetros de la

© Los autores, 2002; © Edicions UPC, 2002.

distribución de probabilidad son función del tiempo y, por lo tanto, los valores de la muestra no están idénticamente distribuidos. Esto ocurre, por ejemplo, en restaurantes, aeropuertos y hospitales en los que el ritmo de llegadas es claramente función del tiempo. Aun en estos casos, se podrá suponer que los valores están idénticamente distribuidos si las variaciones son pequeñas o se trabaja con el peor escenario, el que tiene un ritmo de llegadas más exigente. Detectar que la distribución es inestable puede resultar complicado. Un método empleado para el proceso de llegadas es representar en una gráfica la proporción de llegadas hasta el instante T en función de T, tal como está explicado en el apartado 3.3.2.

EJEMPLO 3.3 Ritmo de llegadas al servicio de urgencias de un hospital

El ritmo de llegadas de pacientes infantiles al servicio de urgencias de pediatría de un hospital depende de la franja horaria, del día de la semana y de la época del año. Por tanto, el proceso no es IID dado que la distribución de probabilidad es dependiente del tiempo.

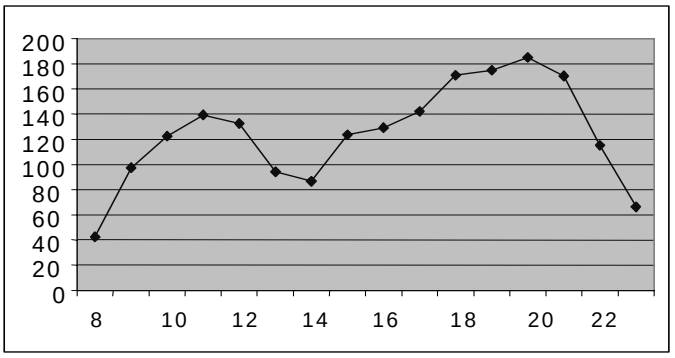


Figura 3.4 Número de pacientes entrados en urgencias pediátricas en función de la hora del día (reproducido con el permiso de la Corporación Sanitaria Parc Taulí)

La figura anterior muestra el número de llegadas por franja horaria, promediadas a lo largo de seis meses, al servicio de urgencias de pediatría de la Corporación Sanitaria Parc Taulí (Sabadell, España). El pico de llegadas por la tarde es debido al incremento de accidentes infantiles después de finalizar el horario escolar y a que los padres tienen más facilidades para llevar a sus hijos a urgencias una vez finalizado su trabajo (aplicable para aquellas urgencias que no son críticas).

El diferente ritmo de llegadas en función del día de la semana tiene su origen principalmente en el incremento notable de llegadas que se produce en los fines de semana debido a una menor disponibilidad de los servicios de atención primaria. La variación de las entradas en función de la época del año tiene su origen en la estacionalidad de enfermedades como la gripe y en el descenso de la población en la zona de influencia en los periodos vacacionales.

© Los autores, 2002; © Edicions UPC, 2002.

3.3.1 Técnicas para evaluar la independencia de los valores de una muestra

A continuación se introducen dos de las técnicas heurísticas empleadas habitualmente para evaluar la independencia de los valores de una muestra: tablas o gráficas de autocorrelación y los diagramas dispersión.

Gráficas de autocorrelación

Las gráficas de autocorrelación se emplean normalmente para evaluar la independencia (aleatoriedad) de una muestra. La independencia se determina calculando los valores de autocorrelación para diferentes valores de retardo. La autocorrelación de retardo k es la correlación entre muestras separadas por k valores. Por ejemplo, la autocorrelación de retardo 2 se aplica a todos los pares de valores V_i consecutivos (V_1,V_2), (V_2,V_3), (V_3,V_4),... y así sucesivamente. Para una muestra de tamaño n, la autocorrelación de retardo k se calcula empleando n-k pares de muestras. Por tanto, es necesario que n sea mayor que k.

Las correlaciones determinadas con muestras pequeñas son poco fiables debido a la variabilidad inherente a la propia estimación. Los valores obtenidos para retardos entre k=1 y k=10 son los que aportan mayor información sobre la independencia de la muestra, mientras que retardos superiores a k=20 no aportan ninguna información. Para muestras independientes, se espera que los valores obtenidos estén centrados en el cero y sean de pequeña magnitud. Para muestras dependientes, y por tanto no aleatorias, los valores de autocorrelación serán significativamente diferentes de cero (los valores de autocorrelación límite son –1 y 1).

La gráfica de autocorrelación está formada por

- Eje vertical: el coeficiente de autocorrelación

$$R_k = \frac{C_k}{S^2(n)}$$

  donde C_k es la función de autocovariancia

$$C_k = \frac{\sum_{i=1}^{n-k}\left[X_i - \overline{X}(n)\right]\cdot\left[X_{i+k} - \overline{X}(n)\right]}{n-k}$$

  y $S^2(n)$ es la variancia de la muestra

- Eje horizontal: retardo temporal k (k=1,2,3,...)

Diagrama de dispersión

El diagrama de dispersión de las observaciones X_1, X_2,...,X_n es una gráfica de los pares (X_i,X_{i+1}) para i=1,2,...n-1. Si los valores X_i son independientes, los puntos (X_i,X_{i+1}) estarán distribuidos aleatoriamente en el plano. No obstante, la naturaleza de la distribución en el plano depende

© Los autores, 2002; © Edicions UPC, 2002.

fundamentalmente de la distribución de probabilidad de los valores X_i. Si los valores X_i están correlacionados positivamente, los puntos de la gráfica tenderán a forma una línea de pendiente positiva en el primer cuadrante si los valores X_i son positivos. Si los valores X_i están correlacionados negativamente, los puntos tenderán a formar una línea con pendiente negativa en el segundo cuadrante si los valores de X_i son positivos. Se recomienda un mínimo de 50 pares de valores para sacar conclusiones [PRAT97].

EJEMPLO 3.4 Análisis de la independencia de dos muestras

Para ilustrar los dos métodos presentados en esta sección, se parte de un modelo cola-servidor con tiempo entre llegadas que sigue una distribución de densidad de probabilidad exponencial de media 1 minuto, un tiempo de servicio según una distribución de probabilidad exponencial de media 0.9 minutos y una cola FIFO (*First In First Out*). Se desea analizar la independencia de las dos muestras siguientes:

- El tiempo de servicio de los primeros 100 clientes servidos.
- El tiempo en cola de los primeros 100 clientes servidos.

La figura 3.5 muestra el gráfico de autocorrelación para la muestra de tiempos de servicio exponencial de media 0.9 minutos. El resultado muestra que los valores son independientes.

La figura 3.6 representa el gráfico de autocorrelación para la muestra de valores de tiempo en cola antes de ser servidos. El resultado indica una importante autocorrelación entre los valores de la muestra, especialmente si el retardo es menor que 10. Queda claro, por tanto, que los valores no son independientes entre sí. Nuestra intuición también confirma este resultado dado que si un cliente ha estado un tiempo elevado en la cola, es de esperar que, en general, el siguiente cliente también esté un tiempo elevado. Lo mismo se podría decir si el tiempo en cola hubiera sido pequeño. Nótese que para obtener el gráfico de autocorrelación (y también para el diagrama de dispersión) es necesario que la muestra esté ordenada por el orden de ocurrencia.

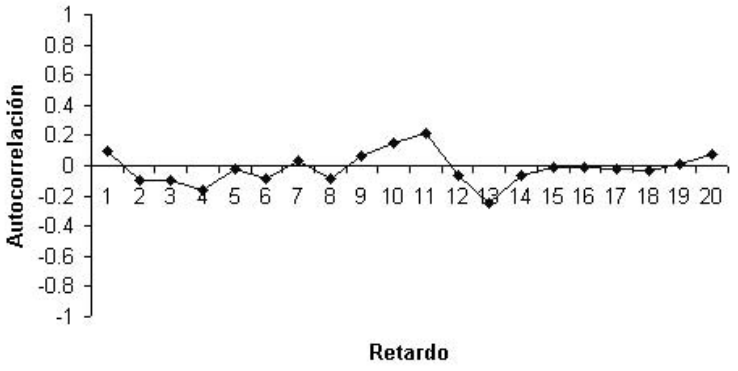


Figura 3.5 Valores de autocorrelación estimados para una muestra IID exponencial de tiempos de servicio

© Los autores, 2002; © Edicions UPC, 2002.

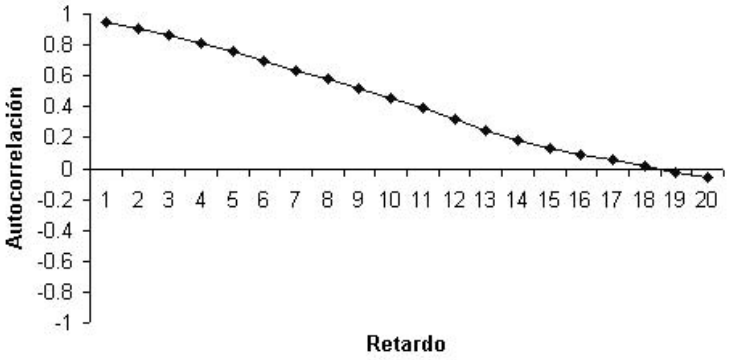


Figura 3.6 Valores de autocorrelación estimados para una muestra de tiempos de espera en la cola

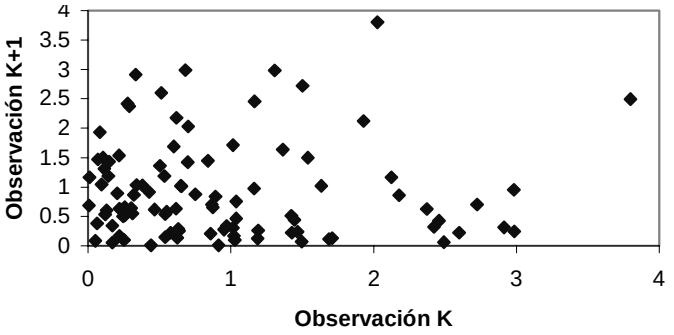


Figura 3.7 Diagrama dispersión para una muestra IID exponencial de tiempos de servicio

Las figuras 3.7 y 3.8 muestran los diagramas de dispersión para la muestra de los tiempos de servicio y para el tiempo de espera en la cola, respectivamente. En la gráfica de los tiempos de espera en la cola se observa que hay correlación positiva entre los valores de la muestra, dado que los puntos se orientan según una línea de pendiente positiva.

3.3.2 Técnicas para evaluar la estabilidad de la distribución

Muchas distribuciones de probabilidad relacionadas con actividades humanas no son estables en el tiempo, incluso en cortos intervalos de tiempo. No obstante, esta falta de estabilidad puede no ser relevante en el contexto del estudio:

- A menudo se desea simular un periodo de tiempo en el que los fenómenos aleatorios son relativamente estables.
- En otras ocasiones se desea simular el peor escenario en el que los fenómenos aleatorios son más restrictivos.

© Los autores, 2002; © Edicions UPC, 2002.

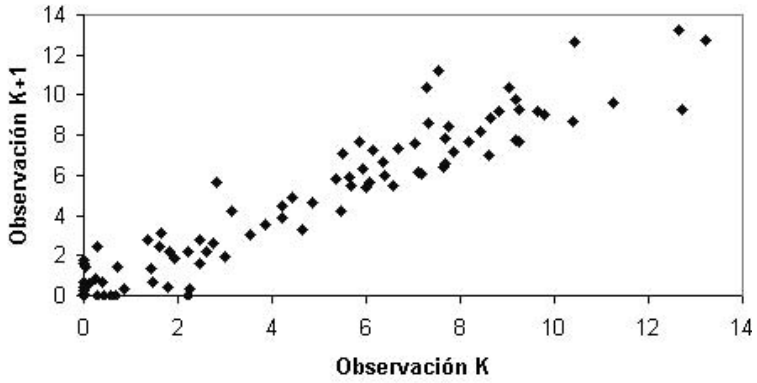


Figura 3.8 Diagrama de dispersión para una muestra de tiempos de espera en la cola

Detectar inestabilidades en las distribuciones probabilísticas puede ser complicado. Para procesos de llegada, una opción recomendada es representar en una gráfica la proporción de llegadas hasta el tiempo T en función del tiempo T. Si la frecuencia de llegadas es relativamente constante a lo largo del tiempo considerado, la gráfica será aproximadamente lineal.

EJEMPLO 3.5 Comportamiento de las llegadas del proceso cola-servidor

La figura 3.9 muestra el comportamiento de las llegadas del proceso cola-servidor descrito en el ejemplo anterior. La evolución aproximadamente lineal de la gráfica indica estabilidad en la frecuencia de llegadas (tiempo medio entre llegadas constante).

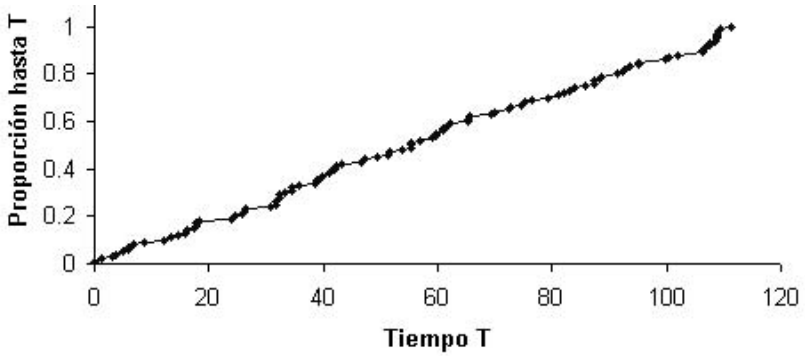


Figura 3.9 Comportamiento de las llegadas (IID exponencial de media 1 min.)

Si se modifica el ritmo de llegadas de tal modo que el tiempo entre llegadas sea función del tiempo (exponencial (1+tiempo/50) min.), se observa que la evolución de la gráfica correspondiente no es lineal (figura 3.10).

© Los autores, 2002; © Edicions UPC, 2002.

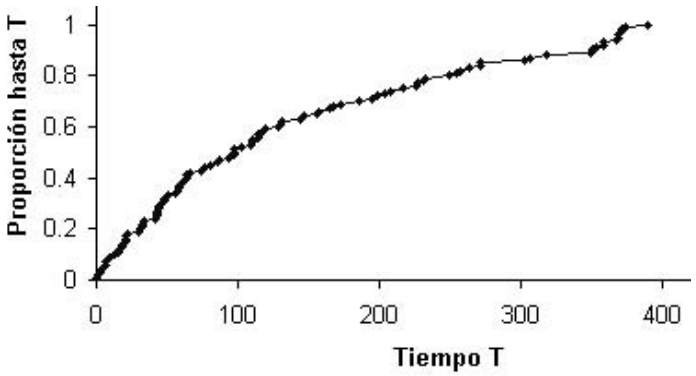


Figura 3.10 Comportamiento de las llegadas (Poisson no estacionaria decreciente)

3.4 Distribución de probabilidad (histograma y diagrama de barras)

Para una muestra de valores continuos, un histograma es esencialmente una descripción gráfica de la distribución de probabilidad de los datos. Adicionalmente, constituye una estimación de la función de densidad de probabilidad. Estas funciones de densidad tienen formas reconocibles en muchos casos. Para una muestra discreta, el gráfico equivalente es el diagrama de barras.

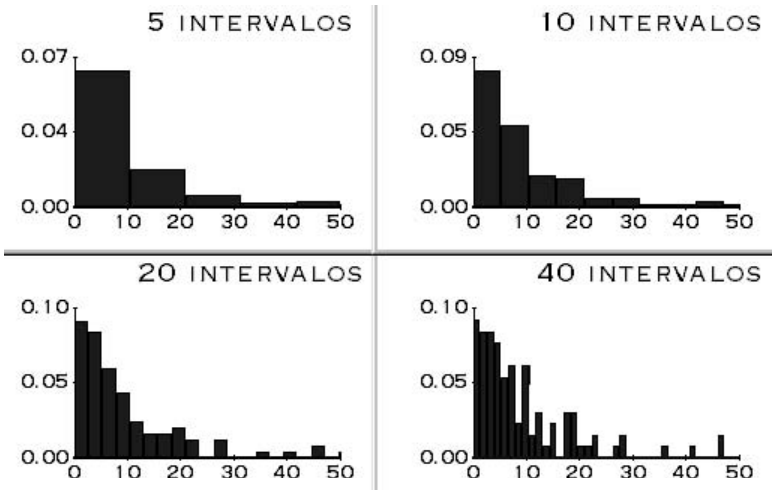


Figura 3.11 Histograma datos de la tabla 3.1

Para obtener el histograma, la muestra de valores continuos se agrupan en intervalos disjuntos y adyacentes $[i_0,i_1),[i_1,i_2),...,[i_{k-1},i_k)$. El número de intervalos en que deben agruparse los datos depende del número de datos de los que se dispone. En la figura 3.11 se han representado los histogramas de los datos experimentales de la tabla 3.1 agrupados en intervalos de diferentes rangos (5, 10, 20 y 40 intervalos). Como puede observarse, el número de intervalos en que se agrupan los datos experimentales es un parámetro muy sensible que puede determinar la selección de la función de distribución de probabilidad teórica que mejor represente el sistema en estudio.

© Los autores, 2002; © Edicions UPC, 2002.

Existen fórmulas que indican cuál podría ser el número óptimo de intervalos en que deberían agruparse *n* datos para determinar la función de distribución de probabilidad que mejor los represente. Una de estas fórmulas es la ecuación de Sturges [STUR26]:

$$k = [1 + \log_2 n] = [1 + 3.322 \log_{10} n]$$

donde las variables utilizadas tienen el siguiente significado:

- k: número de intervalos
- n: número de datos experimentales

De modo análogo, también se dispone de un conjunto de reglas empíricas para determinar el número de intervalos en que deberían agruparse los datos. En general, se suelen escoger entre 5 y 20 intervalos para agrupar los datos experimentales. Nótese que por debajo de 5 intervalos prácticamente no se puede extraer ninguna información para seleccionar una función de distribución de probabilidad que modele los datos en estudio, y por encima de 20 intervalos la representación gráfica que se obtiene es demasiado sensible a la muestra seleccionada.

En la tabla 3.2 se ha descrito la distribución de probabilidades de los datos experimentales de la tabla 3.1 agrupados en 10 intervalos. Dado que el número de observaciones es 100, la probabilidad en cada intervalo es igual al número de valores en el intervalo dividido por 100.

Intervalo	Nº de sucesos	Probabilidad
[0-4)	34	0.34
[4-8)	26	0.26
[8-12)	13	0.13
[12-16)	7	0.07
[16-20)	8	0.08
[20-24)	4	0.04
[24-28)	1	0.01
[28-32)	2	0.02
[32-36)	0	0.00
[36-40)	1	0.01
[>40)	4	0.04

Tabla 3.2 Frecuencia de aparición de los valores

Si los valores de la muestra son discretos, se emplea un diagrama de barras para representar la distribución de probabilidad de la muestra. Para cada posible valor x_i de la muestra, la altura de la barra correspondiente p_i será la proporción de valores de la muestra que son iguales a x_i.

A las distribuciones de probabilidad descritas en este apartado se las puede denominar *distribuciones de probabilidad empíricas* dado que han sido obtenidas directamente a partir de valores experimentales. Si las variables aleatorias son continuas, las distribuciones de probabilidad empíricas pueden adoptar otras formas más adecuadas a la naturaleza continua de la muestra de valores. Estas formas serán explicadas en un apartado posterior.

© Los autores, 2002; © Edicions UPC, 2002.

3.5 Función de densidad de probabilidad

Cuando el proceso estocástico que se quiere describir (dinámica de interés del sistema que se desea modelar) es continuo y no se encuentra limitado a valores discretos, la variable aleatoria utilizada para describir el proceso estocástico puede tomar un número infinito de valores posibles en el rango continuo de interés. Teniendo en cuenta que la probabilidad de obtener un valor cualquiera dentro del rango de valores para el cual ha sido definida la variable aleatoria es 1, la probabilidad de que la variable aleatoria asuma un cierto valor concreto $P(X= x_i)$ será 0. Este es uno de los motivos por los cuales las propiedades estadísticas de una variable aleatoria X se suelen describir mediante la función de densidad de probabilidad f(x). Así pues, en lugar de describir la probabilidad de que la variable aleatoria asuma un valor determinado, se describe la probabilidad de que la variable aleatoria asuma un valor dentro de un rango de valores (entre a y b):

$$P(a \leq X \leq b) = \int_a^b f(x)\,dx$$

Esta información puede obtenerse bien directamente a partir de la función de distribución de probabilidad empírica, bien a partir de una función de distribución de probabilidad teórica parametrizada según el caso particular del sistema en estudio.

Las funciones de distribución de probabilidad empíricas permiten mimetizar mejor los datos reales pero presentan las siguientes limitaciones [LAW99]:

- Son muy sensibles a las condiciones bajo las cuales se ha obtenido la muestra.
- No pueden generar valores fuera del rango de los datos observados. Esto puede representar un grave inconveniente dado que el comportamiento del modelo puede estar muy influenciado por la aparición de eventos de baja probabilidad.
- El empleo en los modelos de simulación de distribuciones de probabilidades empíricas puede resultar laborioso, especialmente si el número de distribuciones e intervalos es elevado.

En cambio, el uso de funciones de distribución de probabilidad teóricas, al aproximar éstas la distribución de probabilidad obtenida empíricamente mediante una función matemática conocida y parametrizable de acuerdo con los datos experimentales recogidos, permite eliminar (filtrar) aquellos resultados que han aparecido por las condiciones según las cuales se ha recogido la muestra, y no responden a la dinámica general del sistema.

En la figura 3.12 se ha representado el histograma de los datos obtenidos del sistema (tabla 3.1) junto con la función de densidad de probabilidad teórica que se le aproxima

$$f(x) = \frac{1}{9.8} e^{-x/9.8}$$

© Los autores, 2002; © Edicions UPC, 2002.

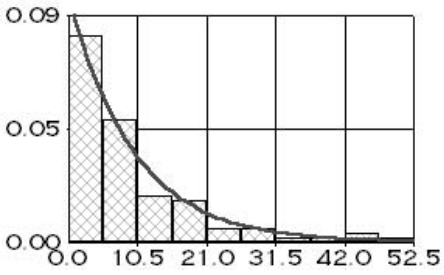


Figura 3.12 Distribución de probabilidad empírica y fdp ajustada teórica

Se puede observar que la función de densidad de probabilidad (fdp) obtenida suaviza la realidad. Esta característica es favorable en aquellos casos en que los datos experimentales se encuentren sometidos a ruido y la muestra de que se disponga sea pobre.

En numerosas situaciones la distribución de probabilidad empírica tiene una forma tal que no existe una distribución estándar que represente adecuadamente el comportamiento del proceso. Las opciones que pueden plantearse para su formalización son varias:

- Emplear directamente la distribución de probabilidad empírica con las ventajas e inconvenientes de esta opción descritas anteriormente.
- Rechazar los valores de la muestra claramente atípicos*:* esto es posible, siempre y cuando la pérdida de información pueda ser asumida en el contexto del proceso que se está modelando.
- Si el histograma tiene varias zonas dominantes se puede intentar separar y ajustar éste en varios casos. Es decir, se ajustará una distribución diferente en cada una de las zonas dominantes [LAW99]. Si p_j es la proporción de muestras en cada zona dominante y $f_j(x)$ la función de densidad de probabilidad en cada una de las zonas, la función de densidad de probabilidad global será

$$f(x) = \sum_{j=1}^{n} p_j f_j(x)$$

EJEMPLO 3.6 Modelado del peso de bobinas de acero

Muchas empresas siderúrgicas, especialmente las que producen acero para el sector del automóvil, trabajan con bobinas, es decir, su materia prima puede ser bobinas de acero sin procesar. En la factoría encontramos bobinas semiprocesadas y el producto final que se entrega al cliente son bobinas procesadas (producto final). Uno de los aspectos más importantes que se debe tener en cuenta para planificar las operaciones productivas es definir las características físicas de la bobina: ancho, longitud, espesor, peso,... La figura 3.13 es el histograma de una muestra del peso en kilogramos por milímetro de ancho de 1200 bobinas de acero. A partir de este valor se puede obtener el peso de la bobina multiplicando por su ancho, el diámetro de la bobina y su longitud, si se conoce su espesor.

© Los autores, 2002; © Edicions UPC, 2002.

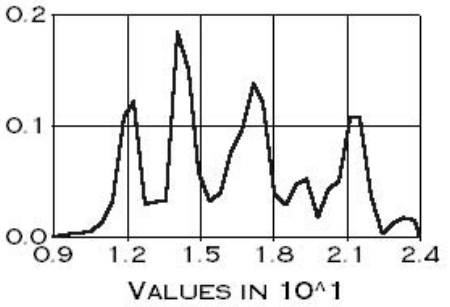


Figura 3.13 Histograma del peso en kg/mm de bobinas de acero proporcionadas por un suministrador (reproducido con el permiso de Siderúrgica del Mediterráneo S.A..)

No es posible obtener una función de densidad de probabilidad que se ajuste al histograma. Por tanto, se ha optado por obtener un ajuste diferente para cada una de las cuatro zonas dominantes. El resultado final ha sido:

Intervalo (kg/mm)	% de las muestras	Fdp ajustada
9 – 13.5	18.56	Weibull (22.52,12.48)
13.5 - 15.5	24.03	Lognormal (14.51,0.41)
15.5 - 18.5	28.78	Normal (17.29,0.74)
18.5 – 24	28.61	Normal (21.18,1.34)

Tabla 3.3 Conjunto de fdp ajustadas

En la figura siguiente se muestra el ajuste obtenido para una de las zonas dominantes.

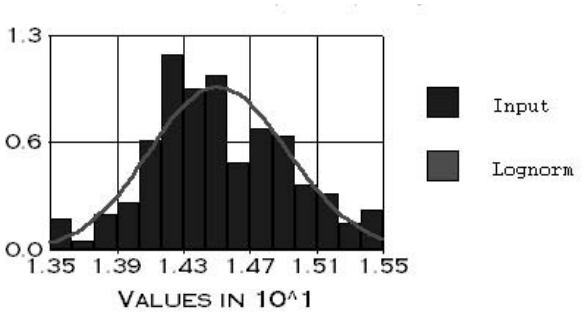


Figura 3.14 Histograma y fdp ajustada para la segunda zona dominante (reproducido con el permiso de la Siderúrgica del Mediterráneo S.A.)

3.6 Función de distribución acumulativa

A pesar de que las propiedades estadísticas de una variable aleatoria (es decir, del sistema a modelar) quedan totalmente especificadas a partir de la función de densidad de probabilidad, también se suelen utilizar en el modelado de sistemas orientados a eventos discretos otras funciones afines como es, por

© Los autores, 2002; © Edicions UPC, 2002.

ejemplo, la función de distribución acumulativa $F(x_i)$, la cual se define como la probabilidad de que la variable aleatoria X asuma un valor más pequeño o igual a un cierto valor dado x_i.

Para las variables aleatorias continuas, la función de distribución acumulativa teórica se determina integrando la función de densidad teórica

$$F(x)=\int_{-\infty}^{x_i} f(x)dx$$

De modo análogo, la función de distribución acumulativa de una variable aleatoria discreta puede calcularse a partir de la expresión

$$F(x)=\sum_{\forall x_i \le x} p(x_i)$$

Considerando, por ejemplo, el caso particular de la función de densidad de probabilidad exponencial de media 9.8

$$f(x)=\frac{1}{9.8}e^{-x/9.8}$$

la función acumulativa (figura 3.15) se obtiene integrando la ecuación anterior:

$$y=F(x)=\int_{-\infty}^{x}\frac{1}{9.8}e^{-x/9.8}dx=1-e^{-x/9.8}$$

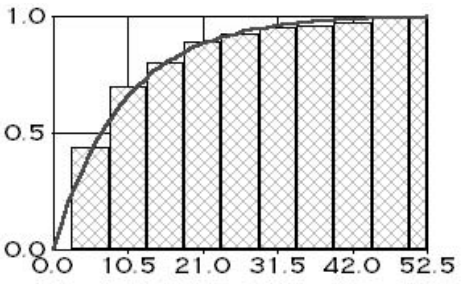


Figura 3.15 Distribución acumulativa de probabilidad para los datos empíricos y la función acumulativa teórica ajustada

A partir de la distribución de probabilidad empírica obtenida en el apartado anterior, se puede calcular la distribución acumulativa empírica añadiendo para cada intervalo la probabilidad que la variable aleatoria adquiera el valor de algún intervalo inferior o igual al que se está evaluando. En la tabla 3.4 se ha representado la distribución empírica de probabilidad acumulativa de los datos experimentales representados en la tabla 3.1 y en la figura 3.15.

© Los autores, 2002; © Edicions UPC, 2002.

Intervalo	Nº de sucesos	Probabilidad	Acumulativa
[0-4)	34	0.34	0.34
[4-8)	26	0.26	0.60
[8-12)	13	0.13	0.73
[12-16)	7	0.07	0.80
[16-20)	8	0.08	0.88
[20-24)	4	0.04	0.92
[24-28)	1	0.01	0.93
[28-32)	2	0.02	0.95
[32-36)	0	0.00	0.95
[36-40)	1	0.01	0.96
[>40)	4	0.04	1

Tabla 3.4 Distribución empírica de probabilidad acumulativa

3.7 Distribuciones empíricas para variables aleatorias continuas

Para variables aleatorias continuas, la distribución de probabilidad puede ser modificada de tal forma que los valores de la distribución sean obtenidos a partir de la interpolación lineal de los valores de la muestra o de su histograma. La forma final de la distribución empírica depende de si se dispone de la muestra original de observaciones X_1, X_2,...,X_n o sólo se tiene el número de muestras que hay en cada intervalo (histograma).

Si los valores originales de la muestra están disponibles, se puede definir una función acumulativa de probabilidad $F_1(x)$ lineal a tramos ordenando, en primer lugar, todos los valores de la muestra en orden creciente de forma que si el índice de orden i es menor que el índice de orden j se cumple que $X_i \leq X_j$. Entonces, la función $F_1(x)$ viene dada por

$$F_1(x) = \begin{cases} 0 & si \;\; x < X_1 \\ \dfrac{i-1}{n-1} + \dfrac{x - X_i}{(n-1)(X_{i+1} - X_i)} & si \;\; X_i \leq x \leq X_{i+1} \text{ para } i = 1,2,...,n-1 \\ 1 & si \;\; X_n \leq n \end{cases}$$

Si la muestra de datos está agrupada en intervalos $[i_0,i_1),[i_1,i_2),...,[i_{k-1},i_k)$, de tal forma que n_j es el número de muestras en el intervalo j y $n=n_1+n_2+...n_k$ es el número de muestras totales, la función $F_2(x)$ viene dada por

$$F_2(x) = \begin{cases} 0 & si \;\; x < i_0 \\ G(i_{j-1}) + \dfrac{x - i_{j-1}}{i_j - i_{j-1}} \left[G(i_j) - G(i_{j-1}) \right] & si \;\; i_{j-1} \leq x \leq i_j \text{ para j} = 1,2,...,k \\ 1 & si \;\; i_k \leq x \end{cases}$$

donde $G(i_0)=0$ y $G(i_j)=(n_1+n_2+...+n_j)/n$ para j=1,2,...,k.

© Los autores, 2002; © Edicions UPC, 2002.

Una de las desventajas de la distribución empírica es que los valores aleatorios generados a partir de ella no pueden estar fuera de los límites de la muestra. Esta deficiencia se puede compensar añadiendo valores adicionales en los extremos de la misma.

EJEMPLO 3.7 Modelado de averías

Una de las fuentes de azar que más impacto producen en un proceso de fabricación son las averías de los elementos productivos o de transporte y los tiempos posteriores de reparación. La tabla siguiente muestra un resumen (histograma) de los tiempos necesarios para reparar una instalación industrial.

Intervalo (min.)	Frecuencia	Frecuencia relativa	Frecuencia acumulada
$0 < x \leq 60$	40	0.4	0.4
$60 < x \leq 120$	10	0.1	0.5
$120 < x \leq 180$	30	0.30	0.8
$180 < x \leq 240$	15	0.15	0.95
$240 < x \leq 300$	5	0.05	1.00

Tabla 3.5 Resumen de los tiempos de reparación de una instalación industrial

La distribución empírica se muestra en la figura 3.16. Obsérvese que se ha obtenido a partir de la interpolación lineal de los puntos obtenidos a partir de la tabla anterior. Según esta distribución, es posible tener tiempos de reparación casi nulos. Si en la práctica se conoce que el tiempo de reparación mínimo es de, por ejemplo 20 minutos, se puede desplazar el primer punto (0,0) hasta el punto (20,0) para reflejar este aspecto.

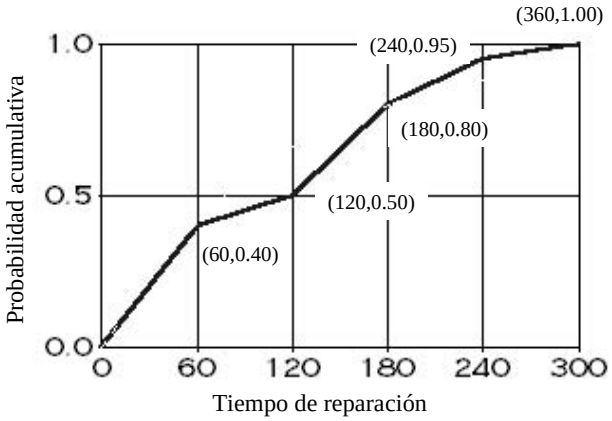


Figura 3.16 Distribución acumulativa empírica $F_2(x)$ de los tiempos de reparación

3.8 Análisis del ajuste de una distribución

Se ha comentado anteriormente que una alternativa al uso de la distribución de probabilidad empírica es el empleo de una función de distribución teórica. Para la determinación de la distribución de

© Los autores, 2002; © Edicions UPC, 2002.

probabilidad teórica se emplean los test de hipótesis. En este apartado no se explica como se escoge la función de distribución teórica que mejor representa el proceso aleatorio, ni tampoco se explican las técnicas para parametrizar las distribuciones. En la actualidad, estas técnicas están automatizadas, de manera que numerosas herramientas estadísticas son capaces de seleccionar un conjunto de distribuciones, parametrizarlas y priorizarlas por orden de calidad del ajuste. Como consecuencia, el objetivo principal de este apartado es introducir los conceptos fundamentales que permiten entender las técnicas básicas empleadas para evaluar la calidad del ajuste. Estas se pueden clasificar en dos grupos: técnicas estadísticas basadas en test de hipótesis y, técnicas heurísticas que emplean representaciones gráficas.

3.8.1 Test de hipótesis

Una hipótesis estadística es una suposición sobre el proceso o actividad que se está muestreando. Puede consistir, por ejemplo, en suponer que el proceso sigue una distribución de probabilidad teórica o que un parámetro del proceso tiene un valor determinado. *El test de hipótesis es simplemente una regla que permite aceptar o rechazar la hipótesis contemplada*. La decisión de rechazar o aceptar la hipótesis se basa normalmente en estadísticas obtenidas directamente de una muestra o conjunto de muestras. Este conjunto de estadísticas se denominan test de estadísticas si se emplean para diagnosticar la validez o no de las hipótesis. La región crítica de un test de estadísticas es el conjunto de valores que tienen como resultado el rechazo de la hipótesis contemplada.

El test de hipótesis se basa en estadísticas efectuadas sobre una muestra de valores. Dado que la muestra puede no ser representativa del comportamiento del proceso, las decisiones tomadas basándonos en la muestra pueden ser erróneas. Consideraremos dos tipos de errores en el test de hipótesis:

1. Rechazar la hipótesis H_0 (*hipótesis nula*) cuando ésta es cierta (error de tipo I). La probabilidad de este tipo error se denomina α.
2. Aceptar la hipótesis H_0 cuando en realidad es falsa (error de tipo II). La probabilidad de este tipo de error se denomina β.

El conjunto de decisiones posible se resume en la tabla siguiente.

Resultado del Test	*H_0 es verdadero*	*H_0 es falso*
H_0 aceptada	Decisión correcta	Error de tipo II (β)
H_0 rechazada	Error de tipo I (α)	Decisión correcta

Tabla 3.6 Decisiones posibles de un test de hipótesis

Evidentemente, uno de los objetivos deseados es minimizar la probabilidad de tomar una decisión incorrecta (minimizar los parámetros α y β). Desgraciadamente, no es posible reducir los dos parámetros al mismo tiempo: si se reduce una probabilidad la otra aumenta. En la mayoría de situaciones, se inicializa α a un valor aceptable y se formula la regla de decisión intentando minimizar β.

© Los autores, 2002; © Edicions UPC, 2002.

La secuencia de pasos que se deben realizar para efectuar un test de hipótesis es:

1. Formular la hipótesis que se pretende diagnosticar (H_0).
2. Determinar el riesgo aceptable de rechazar la hipótesis (inicializar el valor α).
3. Seleccionar el test estadístico apropiado para diagnosticar la hipótesis H_0.
4. Fijar la región crítica para la cual H_0 será rechazada. Es de destacar que 100α% de las veces H_0 se rechazara a pesar de ser cierta.
5. Seleccionar una muestra de valores de un determinado tamaño, ejecutar el test y evaluar la hipótesis.

La selección del test estadístico apropiado (punto 3) se realiza atendiendo al tipo de información de que se dispone sobre el proceso. En la tabla 3.7 se describen brevemente algunos de los test de hipótesis más empleados. Los tests clasificados como *paramétricos* son aquellos en los que la función de distribución a la que responde el proceso es conocida y se emplean para evaluar hipótesis sobre los parámetros de la función de distribución. Los test no-paramétricos se emplean para evaluar las hipótesis que hacen referencia al tipo de distribución que puede utilizarse para describir el proceso.

En el mercado existen herramientas informáticas que permiten ajustar automáticamente la mejor función de densidad de probabilidad teórica a una muestra empírica según los criterios estadísticos de ajuste mas extendidos, dos de los cuales se describen en los próximos apartados: el test de Chi-cuadrado y el de Kolgomorov-Smirnov.

Nombre del test	Tipo	Utilización
t-Student	Paramétrico	Diagnosticar la diferencia del valor medio μ entre dos muestras, suponiendo que responden a una función de distribución normal con media desconocida.
F	Paramétrico	Diagnosticar las diferencias entre dos muestras que responden a una función de distribución normal.
Parámetro de Poisson	Paramétrico	Diagnóstico del parámetro de una función de distribución de Poisson.
Kolgomorov-Smirnov	No paramétrico	Test para evaluar la bondad del ajuste de una función de distribución acumulativa para representar una muestra.
Chi-cuadrado	No paramétrico	Test para evaluar la bondad del ajuste de una función de de densidad de probabilidad para representar una muestra.

Tabla 3.7 Tests paramétricos y no paramétricos

EJEMPLO 3.8 Proceso de llegadas de pacientes a un hospital

Se desea parametrizar correctamente el proceso de llegadas de pacientes a un hospital. Para ello se supone que el ritmo de llegadas es de 20 pacientes cada hora. La hipótesis contemplada para diagnosticar esta presunción es

$$H_o : \mu = 20$$

© Los autores, 2002; © Edicions UPC, 2002.

donde H_0 se denomina *hipótesis nula*, lo cual implica una hipótesis alternativa H_1 que es cierta si H_0 es falsa. Para este caso particular, la hipótesis alternativa puede ser

$$H_1 : \mu \neq 20$$

Si suponemos que se rechaza la hipótesis nula cuando la muestra obtenida nos da un ritmo de llegadas inferior a 17 o superior a 23 pacientes cada hora, la región crítica es

$$\left| \overline{X}(n) - 20 \right| > 3$$

donde $\overline{X}(n)$ es la media de la muestra de llegadas por hora. Por tanto, si el resultado del cálculo anterior da un valor en la región crítica (valor superior a 3), se rechaza la hipótesis H_0.

Test de Chi-Cuadrado

Este test se emplea para validar si una cierta función de distribución puede utilizarse como patrón para generar valores con las mismas propiedades estadísticas que la muestra con la que se está trabajando. El test compara los datos de la muestra con los esperados.

La hipótesis de que los datos observados y la función de distribución sean los mismos se realiza evaluando el test estadístico formulado en la ecuación siguiente

$$\chi^2 = \sum_{i=1}^{r} \frac{(O_i - E_i)^2}{E_i}$$

donde O_i es la frecuencia observada de valores en el intervalo i, E_i es la frecuencia de observaciones esperada si la función de distribución fuese correcta, y r es el número de intervalos empleados. La secuencia de pasos que se deben realizar para evaluar la bondad del ajuste mediante el test de Chi-cuadrado son:

1. Construir la distribución de probabilidad (histograma) de la muestra de valores observados.
2. Calcular las frecuencias esperadas o teóricas de cada uno de los intervalos suponiendo que la función de distribución es la correcta. Los parámetros de la función de distribución se estiman a partir de los valores de la muestra.
3. Calcular el valor χ^2 empleando la expresión matemática descrita anteriormente.
4. Seleccionar un valor para α y evaluar la hipótesis. Esta se rechaza si

$$\chi^2 \geq \chi^2_{\alpha,r-p-1}$$

donde $\chi^2_{\alpha,r-p-1}$ se obtiene de la tabla Chi-cuadrado (ver apéndice C) y (r-p-1) es el número de grados de libertad, (p es el número de parámetros de la distribución teórica que hay que estimar; por ejemplo, para una distribución normal p es igual a 2 (media y variancia)).

© Los autores, 2002; © Edicions UPC, 2002.

La ventaja principal del test de Chi-cuadrado es su generalidad. Puede ser empleado para ajustar tanto distribuciones continuas como discretas empleando los datos de la muestra, de la densidad de probabilidad o los acumulativos. La desventaja principal de esta técnica es que no existe un mecanismo claro en la selección de los intervalos de trabajo para las distribuciones continuas. En algunas situaciones, las conclusiones alcanzadas pueden depender de los intervalos escogidos. Aunque no hay una regla general a seguir, [BANK96] propone el siguiente número de intervalos en función del tamaño de la muestra:

Tamaño de la muestra	*Número de intervalos*
20	No emplear el test de Chi-cuadrado
50	5 a 10
100	10 a 20
>100	$\sqrt{n}$ a n/5

Tabla 3.8 Número de intervalos recomendados para distribuciones continuas

EJEMPLO 3.9 Modelado del tiempo de reparación de una instalación industrial

En el ejemplo 3.5 se muestra el histograma de tiempos para reparar una instalación industrial. Se cree que la distribución triangular (triangular(0, 300, 30)) proporciona un buen ajuste. Dado que la distribución es continua, E_i se puede obtener integrando la función de densidad de probabilidad en el intervalo considerado

$$E_i = 100 \int_{i_{k-1}}^{i_k} f(x)dx = 100[F(i_k) - F(i_{k-1})]$$

donde i_{k-1} e i_k son los extremos del intervalo $[i_{k-1},i_k)$. La función de distribución triangular(a,b,c) es

$$F(x) = \begin{cases} 0 & x < a \\ \dfrac{(x-a)^2}{(b-a)(c-a)} & a \le x \le c \\ 1 - \dfrac{(b-x)^2}{(b-a)(b-c)} & c \le x \le b \\ 1 & b < x \end{cases}$$

donde a< c<b. La tabla siguiente muestra las frecuencias observadas y las esperadas.

Intervalo (mn)	Oi	E_i
$0 < x \le 60$	40	29
$60 < x \le 120$	10	31
$120 < x \le 180$	30	22
$180 < x \le 240$	15	13.5
$240 < x \le 300$	5	4.5

Tabla 3.9 Frecuencias observadas y esperadas

© Los autores, 2002; © Edicions UPC, 2002.

El valor calculado de χ^2 es 21.5 y, dado que $\chi^2_{\alpha,r-p-1}$ es igual a 6.63 para $\alpha = 0.01$ y un solo grado de libertad, se rechaza la hipótesis de que la distribución triangular es un buen ajuste.

Test de Kolmogorov-Smirnov

De manera similar al de Chi-cuadrado, este test también se emplea para evaluar la bondad con que la función de distribución acumulativa puede ser empleada para representar un proceso. La ventaja principal respecto al test de Chi-cuadrado es que trabaja directamente con la función de distribución empírica encontrada directamente a partir de los valores de la muestra y, como consecuencia, no requiere la agrupación de los valores en intervalos. La desventaja principal del test de Komogorov-Smirnov (K-S) es que su rango de aplicación es más limitado. Lo emplearemos exclusivamente para distribuciones continuas dado que los valores críticos para distribuciones discretas no se pueden obtener de una forma fácil. Adicionalmente, el test original de K-S solo es valido si todos los parámetros de la distribución son conocidos; es decir, los parámetros no pueden ser estimados a partir de la muestra. No obstante, en la práctica se ha seguido empleando este test a pesar de que los parámetros hayan sido estimados; excepto en las distribuciones normal, exponencial y Weibull para las que se han creado variantes de este test más apropiadas [LAW99].

La secuencia de pasos que se debe seguir es:

1. Construir la función de distribución acumulativa empírica S(x) a partir de los valores de la muestra.
2. Formular una hipótesis sobre la función de distribución acumulativa teórica F(x) que mejor puede representar los valores de la muestra.
3. Para cada uno de los intervalos en los que se ha descompuesto el rango de valores de la muestra, calcular el valor D según la expresión,

$$D = max_i \left| F(X_i) - S(X_i) \right|$$

4. Escoger un valor α, y rechazar la hipótesis si $D > D_{n,\alpha}$, donde $D_{n,\alpha}$ es el valor crítico [SMIR48].

El primer paso consiste en construir la función de distribución empírica. Se ha visto que existen muchas formas alternativas para definir esta función. Para el test de K-S se define la función de distribución empírica S(x) a partir de los n valores X_1, X_2, ..., X_n de la muestra

$$S(x) = \frac{\text{número de } X_i \leq x}{n}$$

Por tanto, S(x) es una función escalonada tal que si los valores Xi de la muestra están ordenados de menor a mayor valor, entonces

$$S(X_i) = i/n \qquad i = 1,2,\ldots,n$$

© Los autores, 2002; © Edicions UPC, 2002.

Si F(x) es la función de distribución teórica, D es la máxima distancia entre ambas funciones. Debido a que ambas funciones son monótonas crecientes, la máxima distancia se encontrará en los puntos de discontinuidad de la función de distribución empírica (escalones). En cada uno de estos puntos, será necesario calcular la distancia entre la función teórica y los escalones superior e inferior de la función empírica

$$D^+ = \max_{1\le i\le n}\left\{\frac{i}{n} - F(X_i)\right\}, \qquad D^- = \max_{1\le i\le n}\left\{F(X_i) - \frac{i-1}{n}\right\}$$

y, por tanto,

$$D = max_i\left|F(X_i) - S(X_i)\right| = max\left\{D^+, D^-\right\}$$

Cuanto mayor es D peor será el ajuste. Por tanto, se rechaza el ajuste si se excede el valor crítico $D_{n,\alpha}$. Si todos los parámetros de F(x) son conocidos el umbral anterior es equivalente a rechazar la hipótesis si

$$\left(\sqrt{n} - 0.12 + \frac{0.11}{\sqrt{n}}\right)D > c_\alpha$$

donde c_α se encuentra tabulado en

α	0.15	0.1	0.05	0.025	0.01
c_α	1.138	1.224	1.358	1.480	1.628

Tabla 3.10 Valores de c_α

EJEMPLO 3.10 Aplicación del test de Kolmogorov-Smirnov

En el ejemplo 3.9 se ha empleado el test de Chi-cuadrado para evaluar la bondad del ajuste triangular(0,300,30). En este ejemplo se supone que la muestra ordenada de valores (no el histograma) es 60, 74, 127, 136, 145, 162, 181, 205. Para cada uno de estos valores se puede calcular el valor $F(X_i)$ correspondiente a la distribución teórica y el valor $S(X_i)$ correspondiente a la empírica.

Xi	**$S(X_i)=i/n$**	**$F(X_i)$**	**D^+**	**D^-**
60	1/8	0.29	0.165	0
74	2/8	0.37	0.12	0.24
127	3/8	0.63	0.25	0.38
136	4/8	0.66	0.16	0.28
145	5/8	0.70	0.075	0.2
162	6/8	0.76	0.01	0.13
181	7/8	0.82	0.05	0.07
205	8/8	0.88	0.12	0.005

Tabla 3.11 Valores D^+ *y* D^-

© Los autores, 2002; © Edicions UPC, 2002.

Así, $D = max\{D^+, D^-\} = max\{0.25, 0.35\} = 0.35$, y si suponemos que todos los parámetros de la distribución triangular son conocidos,

$$\left(\sqrt{n} - 0.12 + \frac{0.11}{\sqrt{n}}\right) D = \left(\sqrt{8} - 0.12 + \frac{0.11}{\sqrt{8}}\right) 0.35 = 2.74 > 1.62 \quad \text{para } \alpha = 0.01$$

Por tanto, rechazamos la hipótesis del ajuste.

3.8.2 Técnicas heurísticas para determinar la calidad del ajuste

Los test de hipótesis anteriores dan como respuesta el rechazo o aceptación de la distribución teórica candidata para representar el comportamiento del proceso aleatorio. En simulación, no interesa tanto el rechazo o aceptación de la distribución candidata como que la distribución escogida represente razonablemente bien el comportamiento aleatorio del proceso modelado con independencia de la calidad estadística del ajuste. En la práctica, esto puede provocar el rechazo del mejor ajuste estadístico por no incluir algunos fenómenos aleatorios de especial interés. Por esta y otras razones, autores como [VINC98] destacan que los test de hipótesis no son los más apropiados en el contexto de la simulación y recomienda técnicas heurísticas basadas en el sentido común.

Si se emplea una herramienta estadística para ajustar las distribuciones teóricas candidatas y para priorizarlas por orden de calidad, es importante, antes de aceptar una distribución, evaluar cualitativamente la calidad del ajuste, especialmente la capacidad de la distribución candidata para modelar correctamente los valores extremos, que en muchos casos pueden provocar perturbaciones importantes en el proceso de simulación.

Las técnicas heurísticas que se presentan a continuación tienen como objetivo ayudar a evaluar cualitativamente la calidad del ajuste. La primera posibilidad es la comparación directa entre el histograma de la muestra y la función de densidad de probabilidad candidata. La figura 3.17 muestra la comparación entre la fdp lognormal y el histograma, y la fdp Weibull y el histograma. En este caso particular, el mejor ajuste lo consigue la distribución lognormal.

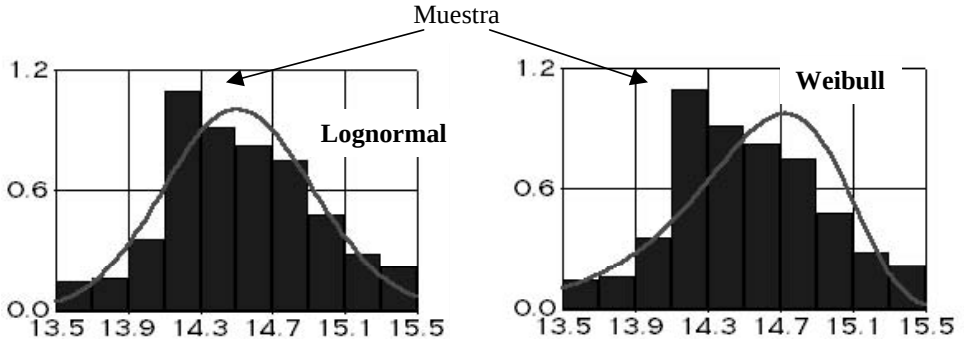


Figura 3.17 Función de densidad de probabilidad lognormal y Weibull superpuestas al histograma

© Los autores, 2002; © Edicions UPC, 2002.

El inconveniente principal de la comparación directa es que, a menudo, el histograma es muy sensible a la configuración de los intervalos. Es decir, pequeños cambios en la configuración del histograma pueden modificar de forma importante la forma de la gráfica.

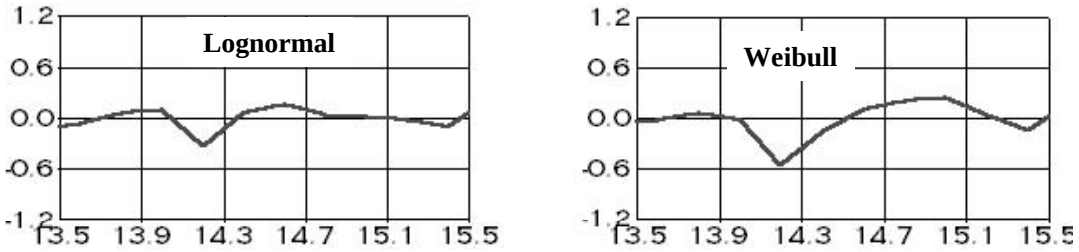


Figura 3.18 Función de diferencias para las distribuciones lognormal y Weibull

La heurística más simple consiste en representar gráficamente las diferencias entre las probabilidades acumulativas empíricas y las del modelo ajustado para cada valor de interés. El eje de las *x* es el mismo que en el caso anterior, pero el eje de las *y* ha sido modificado para reflejar las diferencias que forzosamente tienen que ser menores que 1. En general, el ajuste es mejor cuanto menor es la diferencia. La figura 3.18 muestra el gráfico de diferencias para la lognormal y la Weibull. Las dos gráficas tienen un comportamiento parecido pero la lognormal tiene una desviación menor respecto al cero.

Otra alternativa para evaluar la calidad del ajuste es el gráfico P-P (probabilidad-probabilidad). Permite comparar las probabilidades que se obtienen para la distribución empírica y la ajustada. Para cada observación x_k se representa [Abcisa = $S(x_k)$, ordenada = $F(x_k)$], donde $S(x_k)$ es la distribución de la muestra y $F(x_k)$ la ajustada.

Cuando el modelo se ajusta bien a la muestra los puntos de la gráfica estarán situados cerca de la diagonal. Por tanto, en general el ajuste es mejor cuanto más cerca este la gráfica de la diagonal. En la figura 3.19 se observa que el ajuste de la Lognormal es claramente superior al de la Weibull.

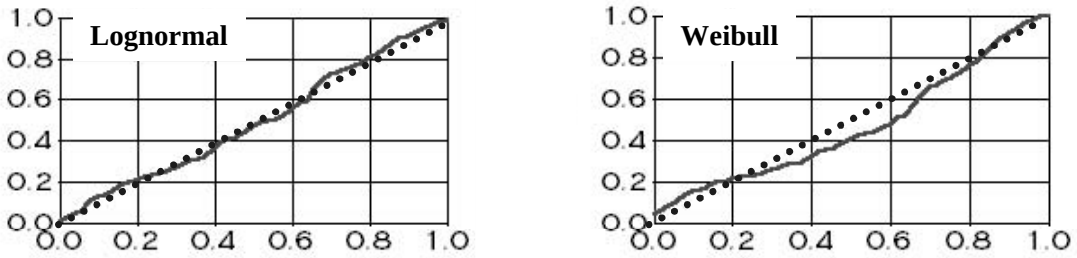


Figura 3.19 Gráficas P-P para las distribuciones Lognormal y Weibull

3.8.3 Qué hacer si no hay datos disponibles

Podemos considerarnos afortunados si disponemos de todas las muestras necesarias para ajustar todas las distribuciones al modelar el proceso de interés. En la práctica, parte o la totalidad de la información necesaria no podrá obtenerse debido a que: o el sistema no está disponible, o está

© Los autores, 2002; © Edicions UPC, 2002.

disponible pero no es posible obtener la información, o no tengamos suficiente tiempo para obtenerla. En muchos estudios de simulación, se invierte más tiempo en la obtención y análisis de los datos que en el diseño y la programación del modelo [MARK81]. En los casos en los que no se dispone de la totalidad de la información es necesario apoyarse en los operadores, diseñadores, vendedores (en general expertos) y/o consideraciones teóricas.

En general, pero muy especialmente cuando la información es incompleta, el modelador tiene que ser consciente de la madurez de toda la información disponible y, por tanto, del nivel de robustez de los resultados del estudio de simulación.

Una primera posibilidad es que sólo dispongamos de la media. En este caso, se puede aplicar directamente la media como un retardo constante. No obstante, esta posibilidad, sólo debe ser usada si la variabilidad real o esperada es muy pequeña. Si ésta no es despreciable, debe acudirse a la experiencia para decidir qué tipo de distribución emplear, por ejemplo, la distribución exponencial si el proceso es de llegadas de tipo *poissoniano*. Dado que los resultados finales estarán condicionados por las decisiones tomadas, es aconsejable mucha prudencia a la hora de interpretar aquellos. En algunos casos, se puede disponer de los valores extremos de la distribución, por ejemplo, el tiempo mínimo y máximo que se tardaría en efectuar una operación. En estos casos se suele emplear la distribución uniforme. A esta distribución se la ha llamado *distribución de la máxima ignorancia* dado que asume que todos los posibles resultados tienen igual probabilidad. Dado que en la realidad esta igualdad de probabilidades casi nunca ocurrirá, parece preferible emplear una distribución triangular, de punto central la moda. Si se conocen los valores extremos y el valor más probable, la distribución triangular es la más adecuada.

3.9 Generación de datos aleatorios a partir de un modelo estadístico

Uno de los objetivos al desarrollar el modelo estadístico de simulación de un sistema, es poder imitar el comportamiento de éste en un ordenador. Existen en la literatura una gran diversidad de algoritmos que permiten generar datos aleatorios con las mismas propiedades estadísticas que los datos obtenidos del sistema a modelar.

La generación de muestras aleatorias que siguen una determinada distribución de probabilidad se realiza generalmente en dos pasos:

1. El primero consiste en generar una variable aleatoria u_i que siga una distribución uniforme entre 0 y 1 (U[0,1]). Para la obtención de una secuencia de números aleatorios uniformemente distribuidos utilizamos *generadores de números aleatorios* (GNA).
2. El segundo paso consiste en convertir el valor u_i anterior en otro que siga la distribución escogida mediante un *generador de variables aleatorias* (GVA). El algoritmo más empleado en esta etapa es el método de la transformada inversa.

Aunque denominamos a los valores extraídos como *aleatorios*, en realidad no lo son dado que han sido obtenidos mediante algoritmos deterministas. Muchos autores denominan a estos valores *pseudo-aleatorios*. No obstante, en este libro y en general, en simulación, a estos valores se los denominan simplemente como *aleatorios*.

© Los autores, 2002; © Edicions UPC, 2002.

Los números aleatorios son utilizados en muchas ramas de la ciencia y de la técnica. Destacamos, a continuación, los principales ámbitos en los que suelen tener un papel relevante [NIED92]:

- Métodos de Monte Carlo
- Métodos de simulación
- Estadística computacional
- Implementación de algoritmos probabilísticos
- Computación científica con componentes estocásticas
- Pruebas de VLSI
- Criptografía
- Juegos

3.9.1 Requerimientos y utilización en simulación

En los proyectos en las áreas de aplicación mencionadas se utiliza, en general, una gran cantidad de números aleatorios para la realización del trabajo. Los requerimientos básicos que se exigen a dichas series de números generados son:

1. Estructurales

   - *Largo período*. Para un uso profesional de los números aleatorios es preciso construir generadores que permitan obtener un gran *período,* siendo éste el número máximo de valores no repetidos que es capaz de generar un método a partir de unos parámetros determinados. Dicha característica es importante para generar series (*streams)* independientes de gran dimensión. Los mayores GNA conocidos, los *Mersenne Twister* de Matsumoto y Kurita [MATS94], producen series del orden de 2^{19937}-1, un número del orden de 6000 dígitos decimales (algoritmo MT19937B en http://www.math.keio.ac.jp/~matumoto).

   - *Estructura reticular.* Evitar una posible cobertura estructura reticular incompleta o anómala. Se basa en el trabajo de Marsaglia [MARS68] en el que se identifica la propiedad de las k-tuplas de números aleatorios generados mediante determinados métodos a distribuirse en hiperplanos y cómo recubren éstos una estructura reticular. Una mala distribución de éstos de modo que la distancia entre hiperplanos sea excesiva, detectada mediante el test espectral [HELL98], puede llegar a representar un serio inconveniente para problemas de grandes dimensiones en los que se utilicen números aleatorios empaquetados en k-tuplas con k mayor que 3 (ver K. Entacher, en http://random.mat.sbg.ac.at/~charly/server/server.html).

   - *Reproductibilidad.* En este caso, el objetivo consiste en ser capaces de reproducir de forma fácil y sistemática cualquier serie de números aleatorios, mediante la utilización de métodos de generación (por ejemplo, *generadores congruenciales lineales* (GCL)) que lo permitan. El número base de la serie se denomina semilla. Esta cualidad tiene mucha importancia en la fase de experimentación, análisis de resultados y técnicas de reducción de la variancia.

© Los autores, 2002; © Edicions UPC, 2002.

2. Estadísticos

- *Uniformidad de la distribución.* Los números aleatorios generados deberán distribuirse de forma uniforme a lo largo de todo el rango de valores posibles que pueda tomar la serie, de forma que no aparezcan tramos no recubiertos ni agrupaciones no deseadas.
- *Densidad*. Esta propiedad tiene mucha relación con la dimensión del periodo y del ciclo de la serie de números generada. A mayor densidad, mayor precisión y finura en la definición de los distintos fenómenos. Así, por ejemplo, si se desea determinar aleatoriamente el próximo palet a ser extraído de un silo robotizado con 100.000 posiciones de almacenamiento, se debe utilizar un método de generación que proporcione, como mínimo, este número de valores distintos a partir de los que se obtiene el identificador de la posición mediante una distribución uniforme.
- *Independencia estadística.* Básicamente, una secuencia de números aleatorios es aquel conjunto de números sobre los que no puede establecerse ninguna relación relevante con los restantes números de la serie. Algunos de los métodos de generación más habituales son recursivos, por lo que no puede afirmarse el principio anterior ya que cada número es calculado mediante un algoritmo determinista aplicado a otro número que le ha precedido en la serie. Aunque en este caso los denominamos números *pseudo-aleatorios*, está claro que existen métodos de generación, convenientemente parametrizados, que producen series perfectamente utilizables, pues cumplen los requerimientos aquí especificados y que se acostumbran a asegurar mediante la superación de determinados test. En toda la literatura aparecen, en general, referenciados como números aleatorios.

3. Teóricos

- *Complejidad*. Los procedimientos de generación serán necesariamente simples, poco costosos en términos computacionales y, en consecuencia, la complejidad algorítmica de los mismos deberá estar convenientemente acotada.
- *Estabilidad del proceso.* Los algoritmos de generación deben ser estables y no degenerar a lo largo de su utilización en un experimento de simulación.

4. Computacionales

- *Facilidad de programación.* Posibilidad de realización de pruebas y depuración de errores. Implementación razonable, fiabilidad, mantenibilidad y capacidad de reutilización.
- *Requerimientos de memoria.* Aunque este no sea un requerimiento tan crítico como lo fue en el pasado, criterios de racionalidad en determinados problemas pueden aconsejar la utilización de algoritmos con baja ocupación de memoria.
- *Eficiencia.* En cuanto a tiempo de puesta a punto o inicialización y tiempo de ejecución. En algunos proyectos, como la aplicación de métodos de Monte Carlo en el ámbito de la física o de la biología, puede llegar a ser un factor fundamental.

© Los autores, 2002; © Edicions UPC, 2002.

Si nos referimos, específicamente, a la utilización de números aleatorios en simulación, debemos respetar los requisitos relacionados con las propiedades estadísticas que garanticen que la muestra generada sea independiente e idénticamente distribuida (IID), la densidad de valores generados, la eficiencia y la reproductibilidad.

3.9.2 Métodos históricos

A título puramente de curiosidad, diversos autores citan métodos, en general basados en tablas, y dispositivos de tipo físico que se utilizaron en tiempos pasados para generar valores aleatorios. Así, por ejemplo, se puede citar la caída de rayos cósmicos, el ruido térmico, fenómenos radioactivos, u otros más prosaicos, como la observación de tablas con los resultados de la lotería o del 6/49 a lo largo del tiempo, trabajar con una ruleta o utilizar una urna con bolas. Estos métodos simplemente no son útiles en simulación ya que no permiten reproducir fácilmente las series e introducen una gran complejidad en la depuración de errores y en la gestión de los experimentos. No obstante, Marsaglia ha elaborado y comercializa un CD con 60 ficheros de 10 MB de números aleatorios binarios utilizables para experimentación, alguno de ellos como combinación de GNA y dispositivos físicos y otras fuentes (¡incluso canciones *rap*!) (http://stat.fsu.edu/~geo/)

3.9.3 Métodos actuales

Probablemente, la introducción de métodos recursivos como los generadores congruenciales lineales por Lehmer [LEHM51] representó un cambio fundamental en los métodos de experimentación basados en números aleatorios. Hoy en día existe una abundante experiencia y bibliografía sobre el tema [PARK88, KLEI92, NIED92, LECU94, KNUT97, HELL98]. Para una referencia completa y actual de los GNA, aplicados básicamente a diversas áreas de la investigación científica y a la criptografía, existen dos referencias básicas: Pierre L'Ecuyer de la Universidad de Montreal, http://www.iro.umontreal.ca/~lecuyer/, y Peter Hellekalek, director del grupo del proyecto pLab de la Universidad de Salzburg, http://random.mat.sbg.ac.at/generators/. A continuación se describen algunos de los GNA más conocidos.

Generadores congruenciales lineales

Una mayoría de los GNA utilizados actualmente en simulación discreta son *generadores congruenciales lineales* (GCL). En éstos, introducidos por *Lehmer*, una secuencia de enteros $Z_i, Z_{i+1}, Z_{i+2}, \ldots$ es definida por la congruencia lineal

$$Z_i \equiv (aZ_{i-1} + c) (\mathrm{mod}\, m)$$

donde m (el *módulo*), a (el *factor multiplicativo*), c (el *factor aditivo*) y Z_0 (la *semilla* o *valor inicial*) son enteros no negativos. Z_i es el resto tras dividir ($aZ_{i-1} + c$) por m.

Adicionalmente a la no negatividad, los enteros m, a, c y Z_0 deben satisfacer $0 < m$, $a < m$, $c < m$ y $Z_0 < m$. De esta forma, se cumple que $0 \leq Z_i \leq m-1$, es decir, todo Z_i es menor que m. Por lo tanto, si en un determinado momento se repite un número, entonces se repite toda la serie. A la longitud de esta serie se la denomina *ciclo* del generador.

© Los autores, 2002; © Edicions UPC, 2002.

Obtenemos los números aleatorios u_i (i=1,2,...) en el intervalo [0,1) dividiendo Z_i por el módulo m

$$u_i = \frac{Z_i}{m}$$
$$u_i \sim U(0,1)$$

Podemos observar, por inducción, que para i = 1, 2,...,n, a partir de la expresión

$$Z_i - aZ_{i-1} \equiv c(\mathrm{mod}\, m)$$

podemos obtener

$$Z_n = \left[a^n Z_0 + \frac{c(a^n - 1)}{a - 1} \right] (\mathrm{mod}\, m)$$

con lo que todo Z_i está completamente determinado por m, a, c y Z_0. De todas formas, una elección cuidadosa de estos cuatro parámetros permite que los correspondientes U_i *parezcan* variables aleatorias uniformemente distribuidas U(0,1) y sean aceptadas si superan una variedad de tests empíricos.

EJEMPLO 3.11 Generadores GCL

En la tabla 3.12 se ha representado los 10 primeros valores de una secuencia de datos pseudoaleatorios que se obtienen utilizando el algoritmo de congruencias lineales parametrizado a los valores a = 9, c = 1, m = 17, y la semilla $X_0 = 5$.

n	X_n	X_{n+1}	X_{n+1}/m
0	5	12	0.7059
1	12	7	0.4118
2	7	13	0.7647
3	13	16	0.9412
4	16	9	0.5294
5	9	14	0.8235
6	14	8	0.4706
7	8	5	0.2941
8	5	12	0.7059
9	12	7	0.4118

Tabla 3.12 Datos Pseudoaleatorios

Los valores a,c,m escogidos dan un periodo de repetición muy bajo (cada 7 datos se repite la secuencia generada) lo cual las hace inservibles para poder extraer propiedades estadísticas.

Generadores congruenciales multiplicativos

Los generadores congruenciales multiplicativos (**GCM**) son aquellos en los que el parámetro aditivo c es igual a cero y el factor multiplicador a es mayor que 1. Son más eficientes que los GCL, pues no es

© Los autores, 2002; © Edicions UPC, 2002.

necesaria la suma del valor c, pero no alcanzan el periodo completo ya que no cumplen la propiedad de que m y c sean primos entre sí. De todos modos, es posible obtener un periodo p = m-1 si m y a son escogidos convenientemente.

La expresión general de un GCM es

$$Z_n = (aZ_{n-1})(\mathrm{mod}\, m)$$
$$m \neq 2^k$$

Algunos hechos que se deben tener en cuenta a la hora de utilizar un GCM son:

1. Utilizar como módulo el mayor número posible primo m.
2. Elegir convenientemente a para obtener p *full period* : p = m - 1.
3. Z_n nunca podrá ser cero, es decir, estará comprendida entre 1 y m - 1.
4. No todos los valores de a son convenientes.

Un GCM es de periodo completo si y sólo si el multiplicador a es una *raíz primitiva del módulo m*.[3] Lehmer propuso que en lugar de trabajar con m = 2^b, se escogiera m como el número primo más grande que fuera menor que 2^b. De esta forma, para m primo, se puede observar el periodo p = m - 1 si a es una raíz primitiva del módulo. Estos generadores se llaman multiplicativos módulo primo (o GCMMP).

EJEMPLO 3.12 Generadores GCM

A continuación se muestran tres ejemplos de GCM; dos de ellos presentan periodo completo y uno muestra cómo una mala elección de los parámetros del generador puede hacer que no se obtengan todos los valores posibles. Así, tenemos un primer generador

$$Z_n = 3Z_{n-1} \bmod 31$$

Con este GCM obtenemos un periodo completo al comenzar por el valor Z_0 = 1. Los valores obtenidos son: 1, 3, 9, 27, 19, 26, 16, 17, 20, 29, 25, 13, 8, 24, 10, 30, 28, 22, 4, 12, 5, 15, 14, 11, 2, 6, 18, 23, 7, 21 y 1, repitiéndose a partir de entonces los valores.

El segundo generador

$$Z_n = 5Z_{n-1} \bmod 31$$

obtiene los valores 1, 5, 25 y 1 (repetición a partir de entonces) al comenzar con el valor Z_0 = 1. Este hecho se produce ya que *a* = 5 no es una rpmm de 31 y por tanto no se cumple la propiedad de la definición dada anteriormente.

El tercer generador es uno de los más utilizados en ordenadores personales dado que utiliza el mayor número primo menor que 2^{31}

[3] a es una raíz primitiva del módulo m (rpmm) $\Leftrightarrow$ a^n (mod m) $\neq 1$, para n = 1,..., n-2. Esto es, el número entero más pequeño *t* por el cual a^t - 1 es divisible por m es t = m - 1

© Los autores, 2002; © Edicions UPC, 2002.

$$Z_n = (7^5 Z_{n-1})(\mathrm{mod}\, 2^{31} - 1)$$
$$Z_n = (16807 Z_{n-1})(\mathrm{mod}\, 2147483647)$$

donde $7^5 = 16807$ es una de las 534.600.000 rpmm de $2^{31} - 1$ (número primo de *Mersenne*).

El principal problema al utilizar estos GCM es el hecho de usar números enteros y no números reales con lo que la precisión de la máquina utilizada nos hará tener en cuenta un posible error de overflow. Para solucionar este problema, Schrage [BRAT87] ideó el método de la división simulada. Este método se basa en el hecho que la operación calculada por la fórmula recursiva del GCM se puede expresar como

$$ax(\mathrm{mod}\, m) = g(x) + mh(x)$$

teniendo en cuenta que los valores expresados por g(x) y h(x) son los siguientes

$$g(x) = a(x \bmod q) - r(x \,\mathrm{div}\, q)$$
$$h(x) = (x \,\mathrm{div}\, q) - (ax \,\mathrm{div}\, m)$$

Al aplicar las expresiones anteriores al cálculo de la expresión original se sabe que

$$r < q$$
$$q = (m \,\mathrm{div}\, a)$$
$$r = (m \bmod a)$$

Por tanto, para todo *x* perteneciente al intervalo [1, m - 1], se cumple que las expresiones implicadas en el cálculo de g(x) son menores que m - 1 y además

$$h(x) = 0 \vee 1$$
$$h(x) = 1 \Leftrightarrow g(x) < 0$$

Obviamente, no es necesario calcular h(x) en ningún caso, pues sus posibles valores (0,1) dependen únicamente del valor de g(x). El procedimiento gana así en eficiencia.

Es algo difícil encontrar buenos GCMMP para b=31, (en la mayoría de ordenadores personales PC actuales el tamaño de palabra disponible, teniendo en cuenta el bit para el signo, es $2^b=2^{31}$). No obstante, el número primo más grande menor que 2^{31} es, sorprendentemente, $m = 2^{31}-1 = 2147483647$, y dos valores conocidos (y ampliamente usados) para a son $a_1 = 7^5 = 16807$ (*Lewis*, *Goodman* y *Miller*)[4], y el a_2=630360016 [PARK88].

Otros tipos de GNA son los generadores tipo Tausworthe y su extensión *generalized feedback shift-register* (GFSR), los *lagged Fibonacci*, los generadores congruenciales inversivos [HELL98] y los generadores combinados [LECU94].

[4] El generador que emplea el simulador ARENA tiene como parámetros $m=2^{31}-1$, $a=7^5=16807$ y c=0 [KELT98]

© Los autores, 2002; © Edicions UPC, 2002.

3.9.4 Tests sobre la calidad en la generación de números aleatorios

Se evalúa si se cumplen determinadas propiedades que se cumplirían si las secuencias generadas fueran *realmente* aleatorias. Básicamente nos fijamos en la uniformidad de los valores de la secuencia y la no-correlación serie en la misma. Algunos tests trabajan con los valores enteros (0 a m-1) mientras otros lo hacen con los mismos números aleatorios $u_i \sim U(0,1)$. De hecho, una secuencia $u_i \sim U(0,1)$ de números aleatorios producida por un GNA no es aleatoria ya que el propio procedimiento de generación es determinista. Algunos generadores, además, funcionan especialmente mal (como era el caso de Randu en el IBM/SSP, que proporcionaba triples consecutivos completamente correlacionados). A este hecho, se le añade que la técnica estadística actual no está equipada para probar hipótesis que contengan procesos no aleatorios y la inadecuación de los tests de hipótesis de independencia en variables uniformemente distribuidas.

Tests teóricos y empíricos

Aunque no cabe duda de la importancia de los test teóricos y la calidad de sus predicciones como el test espectral o el de discrepancias, a continuación nos limitamos a presentar algunos de los recursos disponibles en la red en cuanto a pruebas de tipo empírico.

DIEHARD

George Marsaglia (http://stat.fsu.edu/~geo/diehard.html [MARS97]), ha desarrollado una batería de tests, *Diehard*, aplicables a cualquier gran secuencia de valores pretendidamente aleatorios a fin de probar la calidad de la misma. El objetivo del trabajo no se centra exclusivamente en la prueba de generadores determinísticos (típicamente todos los GNA para simulación, en especial los GCL) sino que trata también de evaluar el output de determinados dispositivos de tipo físico (y, en general, no reproducible).

Se trata de tests diseñados a lo largo de muchos años por el propio Marsaglia que pretenden superar lo que él mismo denomina *tests clásicos*, de entre los que sólo salva los *test run-up y run-down* (*runs test*), considerando el resto superado por sus propias 15 pruebas, recogidas en un disco flexible. Estos test son: *birthday spacings, overlapping 5-permutation, binary rank, binary rank 32X32, binary rank 6X8, bitstream, opso, oqso, dna, count-the-1'S, q5-q4, parking lot, minimum distance, 3d-spheres, sqeeze, overlapping sums, runs y craps.* Ofrece versiones en código fuente para DOS, UNIX y Linux y acoge muy favorablemente todo tipo de mejora, nueva parametrización o incorporación a los programas entregados. Recomienda probar ficheros con un mínimo de 80 millones de bits. Proporciona ayuda para comprobar GNA genéricos de tipo *congruencial*, *shift-register*, *lagged Fibonacci* y *multiply-with-carry* en palabras de ordenador de 32 bits.

Otros disponibles en la red y comprobaciones visuales

Otra batería de pruebas más clásica está disponible en http://www.fib.upc.es/simulacio/tests. Se trata de un módulo ejecutable que realiza los siguientes tests:

- test *de frecuencias o de chi-cuadrado $\chi 2$*. Comprueba la aleatoriedad global. Para funciones discretas o discretizables.
- test *serie*. Detecta zonas sin recubrir.

© Los autores, 2002; © Edicions UPC, 2002.

- test de *gap*. Detecta distribuciones no esperadas de pautas (en este caso, *gaps*)
- test de *yule-walker*. Compara la suma de 5 dígitos con la suma esperada. Detecta pautas.
- test de *distancias*. Comprueba la aleatoriedad global entre dígitos, trabajando con la distancia entre puntos (Ui, Ui+1) en el plano y aplica $\chi 2$.
- test de *autocorrelación*
- test *max-min*. Comprueba la uniformidad de valores máximos y mínimos y aplica $\chi 2$ sobre 100 subintervalos.
- test de *bit condicional*. Prueba la independencia entre bits y aplica $\chi 2$.
- test de *K-S*. Comprueba la aleatoriedad global y local. Para funciones continuas.
- test *run*. Aplica sólo a la prueba de independencia y no a la de aleatoriedad. Es de los calificados como *fuertes*.

Tal como se ha citado anteriormente, en la comprobación de la aplicabilidad de los números aleatorios para un problema particular se acepta concluir que las series tienen la calidad suficiente si superan *razonablemente* un conjunto de pruebas empíricas que consideramos suficientemente potentes. Para visualizar los resultados de una batería de test a la que hemos sometido un conjunto de series generadas por un GNA determinado, se construye una tabla de doble entrada en la que las filas especifican el tipo de test aplicado y las columnas el identificador de cada serie. La tabla se rellena con las respuestas de los distintos test, indicando, por ejemplo en su caso, un valor si/no. Se considera aceptable que una tabla tenga *unos pocos* elementos de resultado negativo aunque, en algunos casos, debe tenerse en consideración *cuán negativos* llegan a ser. No deberían aceptarse generadores que no superen algún test para todas las series, ni que generen alguna serie que no supere ninguno de los test (ver el ejemplo en la tabla 3.13).

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	S	S	S	S	S	S	N	S	S	
2	S	N	S	S	S	S	S	S	S	
3	S	S	S	S	S	S	S	S	S	
4	S	S	S	S	S	S	S	S	S	
5	S	S	N	S	S	S	S	S	S	
6	S	S	S	S	S	S	S	S	S	
7	S	S	S	S	S	S	N	S	S	
8	S	S	S	S	S	S	S	S	S	

a)

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	S	N	S	S	S	S	N	S	S	
2	S	N	S	S	S	S	S	S	S	
3	S	N	S	S	S	S	S	S	S	
4	S	N	S	S	S	S	S	S	S	
5	N	N	N	N	N	N	N	N	N	
6	S	N	S	S	S	S	S	S	S	
7	S	N	S	S	S	S	N	S	S	
8	S	N	S	S	S	S	S	S	S	

b)

Tabla 3.13 Resultados de pruebas de GNA a) pasa b) no pasa

© Los autores, 2002; © Edicions UPC, 2002.

Otras pruebas son los diagramas de dispersión (RIPL77) como el de la figura 3.20 que muestra el diagrama dispersión de una muestra de 1000 valores y algunos métodos de visualización tridimensional de resultados y muchos más tipos de test, como los de póker, el de permutaciones, el test de máximo-de-t, el test de colisión, y los tests en subsecuencias.

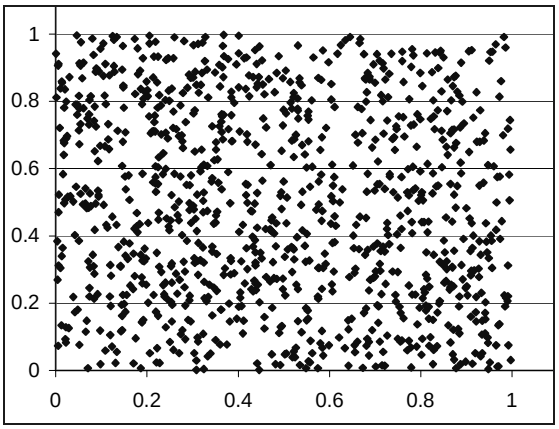


Figura 3.20 Diagrama de dispersión de una muestra de valores aleatorios U(0,1)

3.9.5 Métodos de generación de variables aleatorias: enumeración y principios básicos

Existen diversos métodos para la generación de variables aleatorias a partir de algoritmos que utilizan series de números aleatorios. A continuación, relacionamos los principales. En el apartado siguiente, y en la relación de las distintas distribuciones más conocidas, se incorpora a su ficha técnica algoritmos basados en dichos métodos. Obviamente, las variables aleatorias pueden ser generadas por más de un método, dependiendo su elección de criterios de eficiencia, calidad o rango aceptado de parámetros.

- *Transformada inversa:* aprovecha las posibilidades de inversión de la función de distribución. El más extendido, es eficiente y preciso y reduce el problema de la sincronización. Ejemplo de aplicación: distribuciones exponencial, geométrica, logística, Pareto, Weibull.

- *Composición*: La función de distribución puede ser expresada por una combinación convexa de otras funciones de distribución de más fácil muestreo, por ejemplo, en el método Box-Muller, y en el de Marsaglia para la distribución normal.

- *Convolución***:** Representación de la propia variable resultado como suma de otras variables aleatorias. Es el caso de la distribución m-erlang(b) generada como suma de m variables aleatorias que siguen una distribución exponencial de parámetro (b/m). El método también se usa para la generación de la distribución binomial como suma de variables de Bernouilli, o la de Pascal como suma de variables geométricas.

- *Aceptación/rechazo*: método sistemático para la generación de variables discretas y continuas, es utilizado cuando sólo se conoce la función de densidad de probabilidad y no se puede invertir la función de distribución acumulada. Se basa en la utilización de una fdp g(x) que envuelve tan próximamente como sea posible la fdp de interés. Su eficiencia depende del grado de aceptación de un valor en función de determinados criterios, por lo que no podemos garantizar el esfuerzo

© Los autores, 2002; © Edicions UPC, 2002.

que nos costará el muestreo al no poder predecir cuantos de los valores serán rechazados a lo largo de las iteraciones. Tampoco interesará si la generación de g(x) es compleja o costosa. Por ejemplo, es el caso de generación de la distribución normal por el método polar.

- *Caracterización*: aprovecha las propiedades estadísticas o matemáticas de la distribución como es el caso de la distribución lognormal, la de Poisson por el método multiplicativo, el método basado en la gamma para la distribución beta, y la exponencial por el procedimiento de von Neuman.
- *Comparación*: Este método se basa en la comparación entre series de U(0,1) iid., por ejemplo para la generación de la distribución exponencial.

Aplicación del método de la transformada inversa

La función de distribución acumulativa y = F(x) realiza una aplicación del rango de valores de la variable aleatoria X en el rango [0,1].

$$F : R \rightarrow [0,1]$$

La inversa de la función de distribución acumulativa, permitirá realizar una aplicación de un conjunto de valores definidos en el rango U[0,1] a valores en el rango en el que se encuentra definida la variable aleatoria X

$$F^{-1} : [0,1] \rightarrow R$$

Así pues, si se dispone de un conjunto de datos uniformemente distribuidos en el rango [0,1], será posible mapearlos mediante la inversa de la función de distribución acumulativa en un conjunto de datos en el rango para el cual se encuentra definida la variable aleatoria X, tales que presentarán las mismas propiedades estadísticas que la variable aleatoria X. El principal problema del método de la transformada inversa aparece cuando no es posible invertir la función de distribución acumulativa. En este caso, es necesario recurrir a otros métodos como los descritos para extraer los valores de la función inversa.

En general, cuando el modelo del sistema ha sido descrito mediante funciones de distribución teóricas conocidas, no es necesario implementar algoritmos de generación de datos aleatorios, puesto que la mayor parte de simuladores, así como entornos de programación, ofrecen primitivas de generación de valores aleatorios de las funciones de densidad de probabilidad más utilizadas. En cambio, cuando el sistema ha sido modelado con una función de distribución experimental, es necesario programar rutinas específicas para generar datos aleatorios de acuerdo a la función de distribución seleccionada.

EJEMPLO 3.13 Generación de variables aleatorias

En la tabla 3.13 se recoge una secuencia de datos uniformemente distribuidos [0,1] (columna 3) que han sido generados y normalizados mediante el algoritmo de congruencias lineales, así como un conjunto de datos (columna 4) que responden a una función de densidad de probabilidad exponencial de media 9.8, los cuales han sido generados a partir de la inversa de su función de distribución acumulativa.

$$X = F^{-1}(Y) = -9.8 \times \ln(1-Y)$$

© Los autores, 2002; © Edicions UPC, 2002.

n	y_n	y_n/m	x_n=-9.8 x ln(1-yn)
0	1333	0.0172	18.24
1	65390	0.8446	4.822
2	30086	0.3886	0.4353
3	3364	0.04345	1.3663
4	10075	0.1301	4.6496
56	29246	0.3777	7.0266
7	39621	0.5117	0.8011
8	6077	0.07849	18.416
9	65595	0.8472	7.159
10	40131	0.5183	5.0272
	31067	0.4012	10.6989

Tabla 3.14 Generación de una variable exponencial por el método de la transformada inversa

3.10 Funciones de distribución más utilizadas

La utilización de funciones de distribución teóricas en lugar de funciones de distribución empíricas es muy frecuente. En general, la muestra de que se dispone para la construcción del modelo suele ser demasiado pequeña para representar el comportamiento del sistema en cualquier circunstancia empleando la distribución empírica (los datos disponibles no recogen todos los aspectos).

3.10.1 Funciones de distribución continuas

Para cada una de las funciones presentadas se incluye una tabla resumen que incluye la función de densidad de probabilidad, la función de distribución acumulativa, la media y la variancia. La media (o valor esperado) y la variancia de una variable aleatoria continua X que sigue una función de densidad f(x), se calculan mediante

$$\mu = E(X) = \int_{-\infty}^{\infty} x f(x) dx$$

$$\sigma^2 = V(X) = E(X-\mu)^2 = \int_{-\infty}^{\infty} (x-\mu)^2 f(x) dx$$

Función de distribución Exponencial

Esta distribución se emplea para modelar el tiempo entre llegadas. También se emplea para modelar tiempos de servicio que son muy variables, por ejemplo, la duración de una llamada telefónica. Esta distribución está relacionada con la de Poisson, dado que si una tasa de llegadas (llegadas por unidad de tiempo = λ) sigue una distribución de Poisson, el tiempo entre llegadas sigue una distribución exponencial de parámetro $\beta = 1/\lambda$.

© Los autores, 2002; © Edicions UPC, 2002.

Exponencial	$Expo(\beta)$
Posible interés	Tiempo entre llegadas de clientes cuando la frecuencia media de llegadas es constante.
Densidad de probabilidad	$f(x)=\begin{cases}\frac{1}{\beta}e^{-x/\beta} & x\geq 0\\ 0 & x<0\end{cases}$
Distribución acumulativa	$F(x)=\begin{cases}1-e^{-x/\beta} & x\geq 0\\ 0 & x<0\end{cases}$
Media	β
Variancia	β^2

Tabla 3.15 Ficha técnica función de distribución exponencial

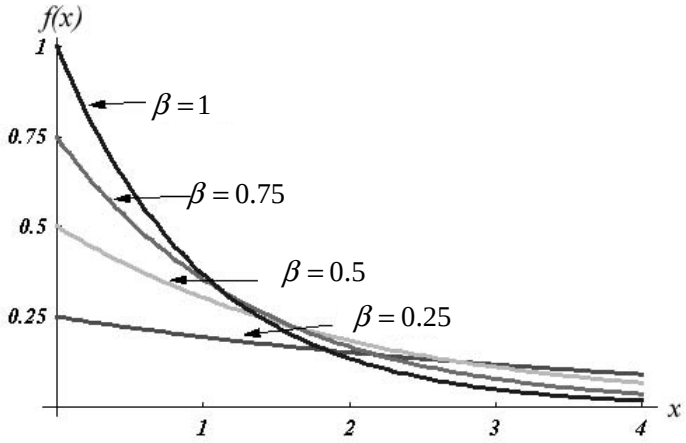


Figura 3.21 Función de distribución exponencial

EJEMPLO 3.14 Llegadas de clientes a una entidad bancaria

Una entidad bancaria tiene una tasa de llegadas al mostrador de dos clientes cada minuto. Por tanto, el tiempo medio entre llegadas es de 1/2 = 0.5 y la variancia es 0.5^2. Si el administrativo tarda 3 minutos en servir al primer cliente, ¿cuál es la probabilidad de que al menos un cliente esté esperando en la cola cuando finalice el servicio al primer cliente? El resultado se obtiene calculando la probabilidad de que el tiempo entre dos llegadas consecutivas sea menor que 3 minutos:

$$F(3)=1-e^{-x/0.5}=1-e^{-6}=0.9975$$

© Los autores, 2002; © Edicions UPC, 2002.

EJEMPLO 3.15 Modelo de llegada de barcos a una terminal portuaria

La figura siguiente muestra el histograma del tiempo entre llegadas de barcos a la terminal de graneles del Puerto de Barcelona. La curva continua muestra la distribución de densidad de probabilidad exponencial ajustada.

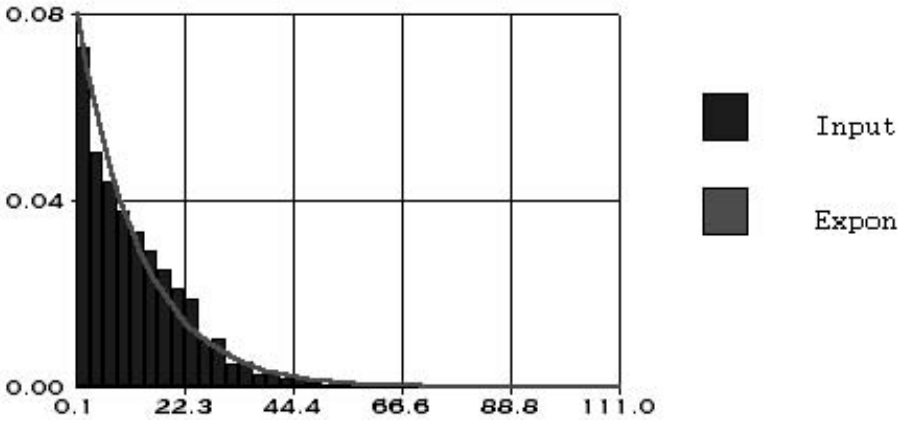


Figura 3.22 Distribución de llegadas de barcos a la terminal de graneles liguidos del puerto de Barcelona, exponencial (12.6) hores. (reproducido con la autorización de la Autoridad Portuaria de Barcelona)

Las funciones de distribución normal, lognormal y gamma, suelen ser las más utilizadas para modelar aquellas actividades tales que, en condiciones normales de operación, el tiempo consumido suele presentar variaciones (justificables físicamente) respecto a un valor promedio.

Función de distribución Gamma

En general, el tiempo que una unidad de producción requiere para realizar una operación repetitiva de procesamiento de materia prima, o bien el tiempo consumido en una actividad repetitiva de transporte de material entre 2 estaciones de trabajo, suele seguir un valor constante con pequeñas variaciones provocadas por ciertos aspectos físicos. Estos podrían ser modelados de modo determinista pero, con el objetivo de simplificar la tarea, se suelen describir como el resultado de una actividad aleatoria mediante modelos estadísticos.

En función de los parámetros de la fdp gamma, ésta presenta una gráfica muy similar a la de la fdp normal, pero con una cierta asimetría que responde a la presencia de datos con valores superiores al valor promedio. Esta asimetría permite modelar secuencias de actividades (por ejemplo, unidades de procesamiento o de transporte) que se realizan en paralelo, tales que cada una de ellas responde a una fdp Normal, pero el tiempo consumido en la secuencia de actividades presenta una asimetría sesgada hacia los valores superiores a la media.

© Los autores, 2002; © Edicions UPC, 2002.

Gamma	$Gamma(\alpha, \beta)$
Densidad de probabilidad	$f(x) = \begin{cases} \dfrac{\beta^{-\alpha} x^{\alpha-1} e^{-x/\beta}}{\Gamma(\alpha)} & x \geq 0 \\ 0 & x < 0 \end{cases}$ $\Gamma(\alpha)$ es la función gamma $\Gamma(\alpha) = \int_0^\infty t^{\alpha-1} e^{-t} dt$ si α es entero positivo $\Gamma(\alpha) = (\alpha - 1)!$
Distribución acumulativa	$F(x) = \begin{cases} 1 - e^{-x/\beta} \sum_{j=0}^{\alpha-1} \dfrac{(x/\beta)^j}{j!} & x \geq 0 \\ 0 & x < 0 \end{cases}$ si α es entero positivo; en caso contrario no hay formula cerrada
Media	$\alpha\beta$
Variancia	$\alpha\beta^2$

Tabla 3.16 Ficha técnica de la función de distribución gamma

Considérese, por ejemplo, la línea de producción representada en la figura 3.24, en la cual cada una de las operaciones de procesamiento puede ser realizada indistintamente por cualquiera de las unidades asignadas a la operación. Nótese que la avería de una máquina comportará un incremento en el tiempo total desde que una pieza espera para poder realizar la operación nº 1 hasta que finaliza la operación número n, puesto que el número de máquinas operativas ha disminuido. Estas variaciones en el tiempo de producción por avería en alguna de las máquinas, se traduce en una asimetría tal que el número de veces en los que se ha finalizado la secuencia de actividades en un tiempo mayor que el valor promedio, es superior al número de veces en las que se ha conseguido finalizar la secuencia de actividades por debajo del valor promedio.

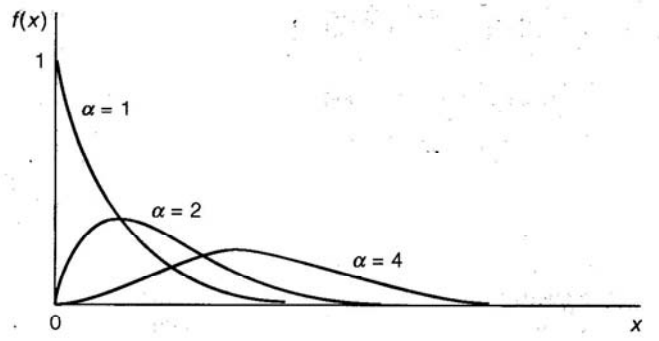


Figura 3.23 Función de distribución gamma (β =1)

© Los autores, 2002; © Edicions UPC, 2002.

La función de distribución gamma representa una herramienta de modelado estadística muy buena para modelar sistemas reales sometidos a la aparición de ciertos eventos (por ejemplo, probabilidad de avería de una máquina) que incrementan la aparición de los valores superiores al valor promedio.

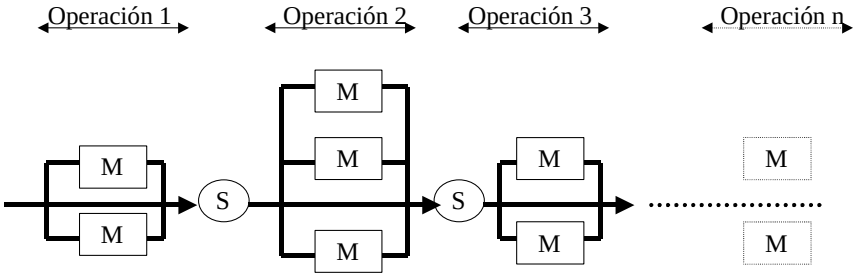


Figura 3.24 Línea de producción

Función de distribución Lognormal

Si se considera de nuevo la línea de producción representada en la figura 3.24 pero con una única máquina asignada a la realización de cada una de las operaciones, el tiempo total de producción en condiciones de operación ideales (sin averías) debería de responder al patrón de una fdp normal. Ahora bien, un modelo más realista, en el que se considerasen los efectos de las averías en las máquinas, respondería a una función de distribución lognormal, caracterizada por la aparición de valores muy superiores al valor promedio, con una baja frecuencia de aparición. Nótese que la avería de una máquina repercutirá en un paro *aguas abajo* de la línea de producción una vez que se hayan procesado las piezas almacenadas en los *stocks* locales (de capacidad finita), y en un incremento considerable del tiempo total de producción de algunas piezas.

En general, la función de distribución lognormal es utilitzada para modelar una secuencia multiplicativa (la avería de una máquina repercute en el paro del resto de máquinas) de operaciones. La función de distribución gamma se utiliza para modelar una secuencia aditiva de operaciones.

Lognormal	$LN(\mu,\sigma^2)$
Densidad de probabilidad	$f(x)=\begin{cases}\dfrac{1}{x\sqrt{2\pi\sigma^2}}\exp\left(\dfrac{-(\ln x-\mu)^2}{2\sigma^2}\right) & x>0\\ 0 & x\le 0\end{cases}$
Distribución acumulativa	No existe fórmula cerrada
Media	$e^{\mu+\sigma^2/2}$
Variancia	$e^{2\mu+\mu^2}\left(e^{\sigma^2}-1\right)$

Tabla 3.17 Ficha técnica de la función de distribución lognormal

© Los autores, 2002; © Edicions UPC, 2002.

Función de distribución Normal

Se utiliza para modelar sistemas tales que el 70% de los datos muestreados se encuentran a una distancia inferior a σ (desviación estándar) del valor promedio μ, y la frecuencia de aparición de los datos se encuentra distribuida simétricamente respecto al valor promedio. Un ejemplo de utilización de una función de distribución normal es el modelado del tiempo de producción de las máquinas cuando no se considera la posibilidad de fallos o errores de diversos tipos.

En la figura 3.25 se ha representado el histograma de una función de distribución normal en la cual, a diferencia de las fdp gamma y lognormal, los datos prácticamente no presentan grandes variaciones respecto a un valor promedio.

Normal	$N(\mu,\sigma^2)$
Densidad de probabilidad	$f(x)=\frac{1}{\sqrt{2\pi\sigma^2}}e^{-(x-\mu)^2/2\mu^2}$
Distribución acumulativa	No existe fórmula cerrada
Media	μ
Variancia	σ^2

Tabla 3.18 Ficha técnica función de distribución normal

La función de distribución acumulativa no puede ser calculada de forma exacta. Como consecuencia, se han empleado métodos numéricos para obtener tablas de la función. Dado que no es práctico obtener una tabla para todos los posibles valores de μ y σ^2, se ha construido una tabla (apéndice C) de la distribución normal estándar (de parámetros μ=0, σ^2=1). Si X es una variable aleatoria de distribución normal de valores de μ y σ^2, la variable aleatoria Z=(X-μ)/σ sigue una distribución normal de media 0 y variancia 1.

EJEMPLO 3.16 Mecanizado de cilindros de acero

Un proceso de mecanizado produce cilindros de acero, con un diámetro que sigue una distribución normal de media 1.005 cm y desviación estándar 0.01 cm. Las especificaciones delimitan el diámetro máximo en el intervalo [0.98, 1.02] cm. ¿Qué porcentaje de piezas no cumplirá las especificaciones?

$$Z=\frac{X-\mu}{\sigma}=\frac{X-1.005}{0.01}$$
$$X=0.98\Rightarrow Z=-2.5$$
$$X=1.02\Rightarrow Z=1.5$$

La probabilidad de exclusión es

$$P(X<0.98)+P(X>1.02)=P(Z<-2.5)+P(Z>1.5)=0.0062+0.0668=0.0730$$

Por tanto, el porcentaje de piezas que no cumplirá las especificaciones es el 7.3% del total.

© Los autores, 2002; © Edicions UPC, 2002.

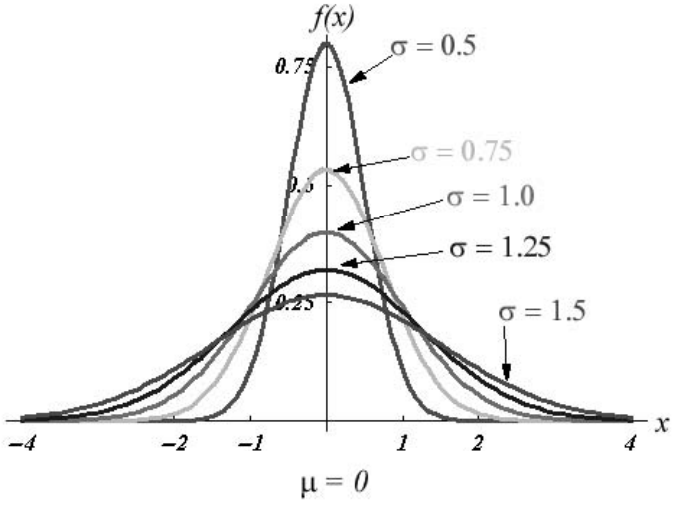


Figura 3.25 Función de distribución normal (μ =0)

Función de distribución Triangular

La distribución triangular proporciona una primera aproximación cuando hay poca información disponible (apartado 3.8.3). Esta distribución queda definida con el valor mínimo, el máximo y la moda. Se emplea para especificar actividades que tienen un tiempo mínimo, máximo y más probable.

Triangular	*trian*(a,b,c)
Densidad de probabilidad	$f(x)=\begin{cases}\dfrac{2(x-a)}{(b-a)(c-a)} & a\le x\le c\\ \dfrac{2(b-x)}{(b-a)(b-c)} & c<x\le b\\ 0 & \textit{resto}\end{cases}$
Distribución acumulativa	$F(x)=\begin{cases}0 & x<a\\ \dfrac{(x-a)^2}{((b-a)(c-a)} & a\le x\le c\\ 1-\dfrac{(b-x)^2}{(b-a)(b-c)} & c\le x\le b\\ 1 & b<x\end{cases}$
Media	$\dfrac{a+b+c}{3}$
Variancia	$\dfrac{a^2+b^2+c^2-ab-ac-bc}{18}$

Tabla 3.19 Ficha técnica de la función de distribución triangular

© Los autores, 2002; © Edicions UPC, 2002.

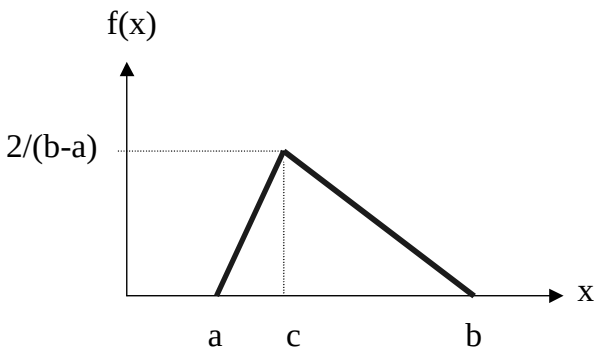


Figura 3.26 Función de distribución triangular

Función de distribución Uniforme

La distribución uniforme es una distribución continua que se emplea para especificar una variable aleatoria que tiene la misma probabilidad de tener su valor en cualquier punto de un rango de valores. Se define especificando un límite inferior a y un límite superior b del rango. La distribución uniforme no es, en general, una representación valida de un fenómeno aleatorio. Se emplea cuando no se conoce la distribución y sólo se tiene información de los valores extremos.

Uniforme	*U (a,b)*
Densidad de probabilidad	$f(x)=\begin{cases}\dfrac{1}{b-a} & \text{a} \le \text{x} \le \text{b} \\ 0 & \text{resto}\end{cases}$
Distribución acumulativa	$F(x)=\begin{cases}0 & x<a \\ \dfrac{x-a}{b-a} & a \le x \le b \\ 1 & b<x\end{cases}$
Media	(a+b)/2
Variancia	$\dfrac{(b-a)^2}{12}$

Tabla 3.20 Ficha técnica de la función de distribución uniforme

© Los autores, 2002; © Edicions UPC, 2002.

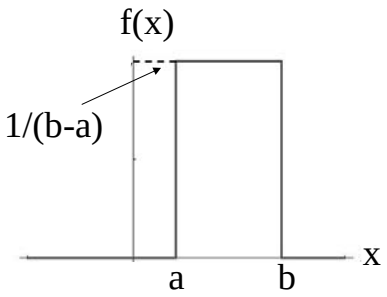


Figura 3.27 Función de distribución uniforme

Función de distribución Weibull

La función de distribución Weibull es parecida a la gamma. Se emplea para modelar tiempos de proceso. También se emplea para modelar la fiabilidad de un equipo al definir el tiempo que transcurre hasta que el equipo falla.

Weibull	*Weibull*(α, β)
Densidad de probabilidad	$f(x) = \begin{cases} \alpha\beta^{-\alpha} x^{\alpha-1} e^{-(x/\beta)^{\alpha}} & x \geq 0 \\ 0 & x < 0 \end{cases}$
Distribución acumulativa	$F(x) = \begin{cases} 1 - e^{-(x/\beta)^{\alpha}} & x \geq 0 \\ 0 & x < 0 \end{cases}$
Media	$\frac{\beta}{\alpha}\Gamma\left(\frac{1}{\alpha}\right)$
Variancia	$\frac{\beta^2}{\alpha}\left\{2\Gamma\left(\frac{2}{\alpha}\right) - \frac{1}{\alpha}\left[\Gamma\left(\frac{1}{\alpha}\right)\right]^2\right\}$

Tabla 3.21 Ficha técnica de la función de distribución Weibull

© Los autores, 2002; © Edicions UPC, 2002.

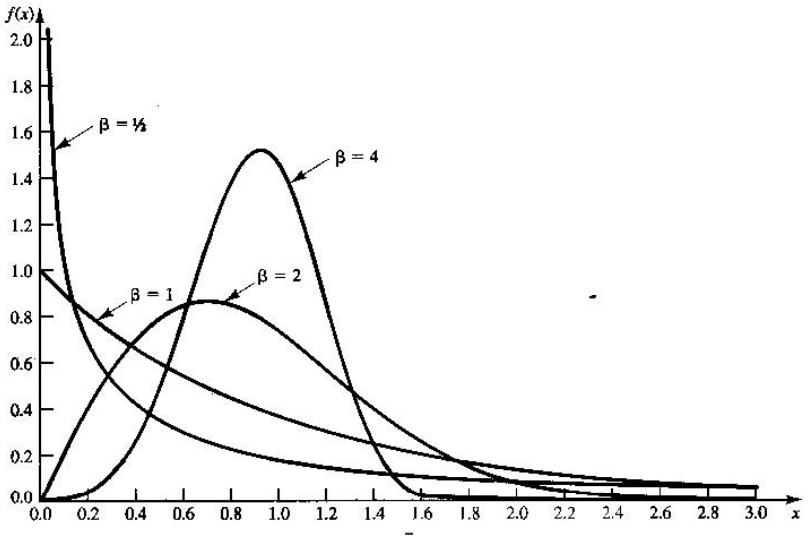


Figura 3.28 Función de distribución Weibull (α =1/2)

3.10.2 Funciones de distribución discretas

Para cada una de las funciones presentadas se incluye una tabla resumen con la función de probabilidad, la función de distribución acumulativa, la media y la variancia.

La media (o valor esperado) y la variancia de una variable aleatoria discreta X que sigue una función de densidad f(x) se calcula mediante

$$\mu = E(X) = \sum_x xp(x)$$

$$\sigma^2 = V(X) = E(X-\mu)^2$$

Función de distribución de Bernoulli

La distribución de Bernoulli se aplica en los casos en los que hay dos posibles estados. La probabilidad de un estado es p y la del otro estado p-1. Los fenómenos que pueden ser definidos pueden ser, entre otros,

- la pieza que sale de un proceso es defectuosa o no;
- una empleada se presenta a trabajar o no se presenta;
- una operación requiere un proceso secundario o una reoperación o no.

© Los autores, 2002; © Edicions UPC, 2002.

Bernoulli	*Bernoulli*(p)
Función probabilidad	$p(x) = \begin{cases} 1-p & \text{si } x = 0 \\ p & \text{si } x = 1 \\ 0 & \text{resto} \end{cases}$
Distribución acumulativa	$F(x) = \begin{cases} 0 & \text{si } x < 0 \\ 1-p & \text{si } 0 \le x < 1 \\ 1 & 1 \le x \end{cases}$
Media	P
Variancia	p(1-p)

Tabla 3.22 Ficha técnica de la función de distribución Bernoulli

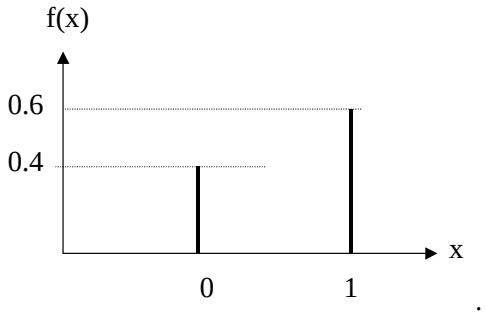


Figura 3.29 Función de probabilidad de Bernoulli (p=0.6)

Función de distribución Uniforme discreta

Se emplea cuando todos los valores en el intervalo [i,j] tienen igual probabilidad. Se emplea como un primer modelo cuando sólo se tiene información sobre los límites del intervalo.

© Los autores, 2002; © Edicions UPC, 2002.

Uniforme disc.	$UD(i, j)$
Función probabilidad	$p(x) = \begin{cases} \dfrac{1}{j-i+1} & \text{si } x \in \{i, i+1, \ldots, j\} \\ 0 & \text{resto} \end{cases}$
Distribución acumulativa	$F(x) = \begin{cases} 0 & \text{si } x < i \\ \dfrac{\lfloor x \rfloor - i + 1}{j-i+1} & \text{si } i \le x \le j \\ 1 & j < x \end{cases}$ donde $\lfloor x \rfloor$ indica el entero más grande $\le x$
Media	(i+j)/2
Variancia	$\dfrac{(j-i+1)^2 - 1}{12}$

Tabla 3.23 Ficha técnica de la función de distribución uniforme discreta

Función de distribución Binomial

La distribución binomial es una distribución discreta que expresa el resultado de n experimentos independientes. Es esencialmente la suma de n experimentos de Bernoulli. Supongamos que un experimento que tiene dos posibles resultados se realiza n veces (n>0). Supongamos también que la probabilidad de obtener un resultado determinado (lo llamamos resultado *a*) para cualquier experimento es p, y la probabilidad del otro resultado es q=1-p (lo llamamos resultado *b*). Entonces el resultado a puede aparecer un número de veces entre 0 y n, al igual que el resultado b. La distribución binomial especifica la probabilidad de que el resultado a ocurra k veces. Algunos fenómenos que pueden ser definidos con esta distribución son:

- el número de piezas defectuosas en un lote;
- el número de clientes de un tipo particular que entran en el sistema.

© Los autores, 2002; © Edicions UPC, 2002.

Binomial	*Bin(n,p)*
Función probabilidad	$p(x)=\begin{cases}\binom{n}{x}p^x(1-p)^{n-x} & \text{si } x\in\{0,1,\ldots,n\}\\ 0 & \text{resto}\end{cases}$ donde $\binom{n}{x}=\frac{n!}{x!(n-x)!}$
Distribución acumulativa	$F(x)=\begin{cases}0 & \text{si } x<0\\ \sum_{i=0}^{\lfloor x\rfloor}\binom{n}{i}p^i(1-p)^{n-i} & \text{si } 0\le x\le n\\ 1 & n<x\end{cases}$ donde $\lfloor x\rfloor$ indica el entero más grande $\le x$
Media	np
Variancia	np(1-p)

Tabla 3.24 Ficha técnica de la función de distribución binomial

EJEMPLO 3.17 Modelado de fusibles defectuosos

Supongamos que un lote de 1000 fusibles tiene un 10% de defectos. Cuatro fusibles son extraídos al azar del lote. Se desea determinar la probabilidad de que un solo fusible sea defectuoso y la probabilidad de que como mínimo un fusible sea defectuoso del total de cuatro del lote.

Dado que el tamaño del lote es muy elevado, la probabilidad de extraer un lote defectuoso será aproximadamente la misma (0.1) a pesar de que se hayan extraído uno, dos o tres fusibles (esta hipótesis no sería correcta si el tamaño del lote fuera pequeño). Por tanto, la distribución binomial se puede emplear para responder a las preguntas anteriores. Si denominamos Y al número total de defectos del lote extraído, entonces

$$p(1)=\binom{4}{1}\cdot 0.1\cdot 0.9^3=0.2916$$
$$P(Y\ge 1)=1-P(Y=0)=1-p(0)=0.3439$$

Función de distribución de Poisson

La frecuencia de aparición de eventos en un proceso de llegadas puede formalizarse bien especificando el tiempo entre 2 llegadas sucesivas, bien especificando el número de eventos de llegada por intervalo de tiempo.

© Los autores, 2002; © Edicions UPC, 2002.

- *Tiempo entre 2 eventos de llegada sucesivos*: en general, el tiempo entre dos eventos independientes de llegada consecutivos suele responder a una distribución exponencial.
- *Número de eventos de llegada por intervalo de tiempo*: en lugar de describir el tiempo entre eventos de llegada, se describe el número de eventos en un intervalo de tiempo constante. Nótese, por ejemplo, que no es posible describir mediante una distribución exponencial la llegada de material a una unidad de producción cuando ésta es transportada en palets con un número de piezas variables, ya que el tiempo entre la llegada de una pieza y la siguiente es 0. La distribución de Poisson es una de las distribuciones más utilizadas para describir este tipo de comportamiento. Esta distribución fue desarrollada originalmente para modelar las llamadas telefónicas a una central. Otros fenómenos que pueden ser modelados son:

  1. el número de entidades temporales que llegan por unidad de tiempo;
  2. el número total de defectos en una pieza;
  3. el número de veces que un recurso es interrumpido por unidad de tiempo.

Poisson	$Poisson(\lambda)$
Función probabilidad	$p(x)=\begin{cases}\dfrac{e^{-\lambda}\lambda^{x}}{x!} & \text{si } x \in \{0,1,\ldots,\} \\ 0 & \text{resto}\end{cases}$
Distribución acumulativa	$F(x)=\begin{cases}0 & \text{si } x<0 \\ e^{-\lambda}\sum_{i=0}^{\lfloor x \rfloor}\dfrac{\lambda^{i}}{i!} & \text{si } 0 \le x\end{cases}$
Media	λ
Variancia	λ

Tabla 3.25 Ficha técnica función de distribución binomial

EJEMPLO 3.18 Entrada de vehículos en un aparcamiento

El número de coches que entran en un parking es una variable aleatoria que sigue una distribución de Poisson de media 4 coches por hora. El aparcamiento tiene una capacidad máxima de 12 vehículos. Se pide determinar la probabilidad de que el aparcamiento se llene en la primera hora si suponemos que todos los vehículos están más de 1 hora. También se pide determinar la probabilidad de que lleguen menos de 12 vehículos en un periodo de horas.

$$P(Y \ge 12) = (1 - P(Y \le 11)) = 1 - e^{-4}\sum_{i=0}^{11}\frac{4^{i}}{i!} = 0.00092$$

$$F(11) = P(Y \le 11) = e^{32}\sum_{i=0}^{11}\frac{32^{i}}{i!} = 0.000017$$

Para responder a la segunda pregunta se ha considerado que la media es de 32 vehículos cada 8 horas.

© Los autores, 2002; © Edicions UPC, 2002.

Función de distribución Geométrica

La distribución geométrica describe el número de experimentos con probabilidad p de éxito que se debe que efectuar hasta obtener un resultado determinado. Algunos ejemplos de fenómenos que se pueden modelar con esta distribución son:

- el número de ciclos de máquina hasta que ésta falla;
- el número de piezas inspeccionadas hasta encontrar una con defectos;
- el número de clientes servidos hasta que se encuentra uno de un tipo determinado.

Geométrica	*Geom(p)*
Función probabilidad	$p(x) = \begin{cases} p(1-p)^x & \text{si } x \in \{0,1,\ldots,\} \\ 0 & \text{resto} \end{cases}$
Distribución acumulativa	$F(x) = \begin{cases} 1-(1-p)^{\lfloor x \rfloor+1} & \text{si } x \geq 0 \\ 0 & \text{resto} \end{cases}$
Media	(1-p)/p
Variancia	$(1-p)/p^2$

Tabla 3.26 Ficha técnica de la función de distribución binomial

3.11 Ejercicios

1. A partir de los valores de la tabla 3.26 obtén su diagrama de autocorrelación.

4	68	63	45	24	34	74	9	48	32	98	96	100	85	30
25	16	86	96	20	35	84	35	94	89	56	88	29	72	85

Tabla 3.27 Valores para un diagrama de dispersión

2. Busca dos procesos diferentes que generen secuencias de valores y representarlos mediante un diagrama de autocorrelación para que sean:

   a) independientes
   b) dependientes

   Recoge 100 valores y realiza también el diagrama de dispersión de éstos.

3. La tabla 3.27 muestra el tiempo de llegada en segundos de elementos para ser procesados. Estudia la estabilidad de esta distribución.

0	3	7	10	11	12	13	42	49	68	85	87	103	132	133	139
146	149	151	153	185	203	222	273	288	299	310	312	317	317	319	331
335	345	353	355	369	380	381	409	420	428	432	435	436	439	460	475
477	483	498	502	505	513	532	537	543	547	550	556	563	577	594	600
616	616	618	620	623	628	633	635	637	639	640	640	643	646	646	662

© Los autores, 2002; © Edicions UPC, 2002.

666	671	689	690	696	697	714	738	741	754	755	759	770	772	774	780
782	813	821	825	830	830	853	853	855	856	860	861	864	866	871	883
883	885	886	889	894	895	901	904	906	906	925	926	929	933	940	949
951	953	959	970	972	973	976	977	986	991	991	996	1004	1005	1018	1027
1027	1031	1032	1033	1038	1040	1043	1049	1049	1049	1052	1056	1057	1057	1059	1060
1061	1062	1062	1069	1071	1071	1072	1073	1079	1081	1083	1084	1089	1089	1090	1094
1100	1109	1112	1128	1130	1131	1135	1137	1138	1139	1140	1141	1145	1155	1158	1162
1162	1164	1170	1171	1173	1173	1175	1175								

Tabla 3.28 Tiempos de llegada

4. El problema de los filósofos se ha presentado en el apartado de ejercicios del capítulo 2. Si se supone que el tiempo de meditación sigue una distribución uniforme entre 20 y 210 minutos y el de comer sigue una distribución uniforme entre 10 y 30 minutos, obtén empleando el método de la transformada inversa diez valores aleatorios para cada una de las distribuciones uniformes.

5. La función de densidad de probabilidad de la figura representa el tiempo de proceso en minutos de un trabajador que realiza una operación de montaje. ¿Cuál es la probabilidad de que el tiempo de montaje sea menor o igual a 6.5 minutos?

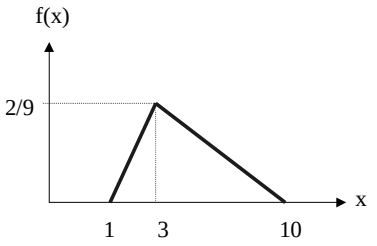


Figura 3.30 Tiempo de proceso

6. Realiza el histograma del tiempo entre llegadas de los valores de la tabla siguiente. Qué funciones de densidad de probabilidad podrían aproximar esta distribución?

2.73	2.82	2.64	2.22	2.44	2.41	3.79	3.06	3.53	3.50	2.44
2.73	2.56	2.54	3.97	3.57	3.61	4.39	3.50	3.33	3.33	2.66
3.30	3.20	2.51	3.53	3.35	2.44	3.98	3.41	3.22	2.94	2.79
3.06	3.64	2.93	2.71	3.29	3.78	3.12	3.07	3.02	2.77	3.19
1.91	2.74	2.68	2.95	3.04	3.12	2.82	2.65	2.75	2.48	1.65
3.12	3.86	2.50	3.25	2.64	3.85	4.16	2.86	3.71	2.61	3.08
4.50	3.45	3.13	3.46	3.85	2.40	2.96	3.05	2.95	1.95	2.63
2.31	2.77	3.84	3.58	2.74	3.17	3.64	3.76	3.16	3.73	3.38
2.90	2.84	2.25	3.74	3.08	3.63	2.70	2.80	3.30	2.72	2.89

© Los autores, 2002; © Edicions UPC, 2002.

7. Se han registrado los tiempos absolutos de llegadas de clientes a la gasolinera del km 207 de la Nacional II. El tiempo de valor 0 corresponde a las 6 de la mañana, instante en el que se ha iniciado la toma de tiempos. Los valores registrados de los tiempos entre llegadas, expresados en minutos, son:

14.1	15.1	18	32.4	37.9	46.6	52.6	55.7	73	73.8	76	76.5
79	92.3	103.8	127.2	133.2	137.4	159.3	160.7	163.5	164.3	172.7	181.4

Tabla 3.29 Tiempos absolutos de llegada a una gasolinera

¿Qué distribución emplearías para simular el proceso de llegadas?

8. En la tabla siguiente aparece una muestra de 23 valores de tiempos de servicio consecutivos registrados para modelar un proceso. Evalúa si las observaciones son independientes y están idénticamente distribuidas (IID).

105	28	98	44	128	46
68	105	48	52	173	52
54	69	93	69	84	69
42	128	42	18	33	

Tabla 3.30 Observaciones consecutivas del tiempo de servicio

9. A partir de los datos del ejercicio anterior construye la distribución empírica acumulativa mediante interpolación lineal.

10. Una técnica para generar números aleatorios uniformemente distribuidos entre 0 y 1 consiste en, a partir de una primera semilla de n dígitos, elevar el número al cuadrado y extraer del nuevo valor los n dígitos centrales para obtener el valor aleatorio (*mid-square method*) [BILE96]. Por ejemplo, dado $X_0 = 3187$ ($R_0 = 0.3187$),

$$X_0^2 = (3187)^2 = 10156969 \rightarrow X_1 = 1569 \rightarrow R_1 = 0.1569$$
$$X_1^2 = (1569)^2 = 2461761 \rightarrow X_2 = 4617 \rightarrow R_2 = 0.4617$$
$$X_2^2 = (4617)^2 = 21316689 \rightarrow X_3 = 3166 \rightarrow R_3 = 0.3166$$

Empleando este generador, escoger una semilla, obtén los 100 primeros valores y evaluar la calidad del generador analizando la independencia de los valores y la uniformidad de la distribución.

11. Mediante el método de generación de números aleatorios que utiliza la herramienta de simulación ARENA ($m=2^{31}-1$, $a=7^5=16807$ y $c=0$), calcula los 100 primeros valores partiendo de una semilla con valor 862. Analiza la independencia de los datos y realiza el diagrama de dispersión.

© Los autores, 2002; © Edicions UPC, 2002.

12. Implementa el algoritmo de Schrage utilizando los valores siguientes para los parámetros del generador congruencial multiplicativo :

$a = 16807$ $m = 2147483647$ $q = 12773$ $r = 2836$

Para realizar la comprobación del algoritmo verifica que el valor Z_{100000} es 1.043.618.065, si partimos de la semilla $Z_0 = 1$.

13. Una de las primeras aplicaciones de los números aleatorios fue el cálculo de integrales. Dada la función g(x), se desea calcular

$$\theta = \int_0^1 g(x)dx$$

dada la función de densidad de probabilidad uniforme entre 0 y 1, el valor de θ se puede expresar como θ = E[g(U)]. Adicionalmente conocemos que

$$\sum_{i=1}^{k} \frac{g(U_i)}{k} \to E[g(U)] = \theta \quad \text{si } k \to \infty$$

Por tanto, se puede calcular la integral de forma aproximada a partir del promedio de valores de $g(U_i)$. A esta técnica se la denomina *método de Monte Carlo*. Este método fue desarrollado por S.Ulma y Von Neumann (ver su biografía en http://www.lanl.gov/worldview/science/bios.shtml) durante la Segunda Guerra Mundial, y fue empleado para resolver problemas relacionados con el desarrollo de la bomba atómica [ULAM91, DUBI00, LIU01]. Este método tiene su mayor interés en las integrales multidimensionales. Si suponemos que g es una función n-dimensional, entonces

$$\theta = \int_0^1 \int_0^1 \cdots \int_0^1 g(x_1, x_2, \cdots, x_n)dx_1 dx_2 \cdots dx_n = E[g(U_1, U_2, \cdots U_n)]$$

Programa el generador de números aleatorios congruencial lineal de parámetros m=2147483647, a=16807 y c=0, y los algoritmos adicionales necesarios para calcular

1. $\int_0^1 e^{x+x^2} dx$

2. $\int_0^1 \int_0^1 e^{(x+y)^2} dxdy$

Evalúa también la calidad de los resultados en función del número de muestras tomadas.

© Los autores, 2002; © Edicions UPC, 2002.

4 Simulación de sistemas orientados a eventos discretos

Aunque a primera vista podría parecer que los mecanismos de funcionamiento interno de los programas de simulación tan sólo tendrían que importar a aquellos usuarios interesados en el desarrollo de los mismos, en realidad, no es así. En la mayor parte de proyectos de simulación, el usuario final tan sólo precisa conocimientos sobre aspectos externos (interfase) del simulador. Estos conocimientos le permiten construir, simular y analizar su modelo. Pero, de hecho, esta visión no es la correcta, ya que un buen conocimiento del funcionamiento interno del simulador es, en muchas situaciones, imprescindible para la obtención de los resultados perseguidos y la comprensión y corrección de los problemas de programación del modelo [SCHR00]. El conocimiento de los mecanismos internos de un simulador facilitará:

- el modelado de situaciones complejas
- la detección, comprensión y corrección de errores.

En este capítulo se introducen los mecanismos básicos de funcionamiento de un simulador orientado a eventos discretos con el objetivo de hacer entender y familiarizar al lector con las diferentes estrategias de simulación.

EJEMPLO 4.1 Subproceso de descarga de *palets* de un camión

En la figura 4.1 se muestra la RdP del subproceso de descarga de palets de un camión en el contexto de un proceso logístico.

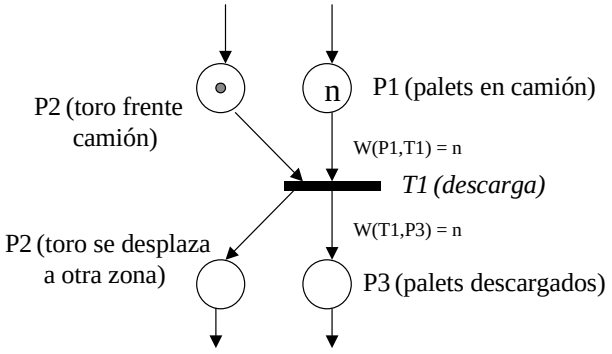


Figura 4.1 RdP de la descarga de palets *de un camión*

© Los autores, 2002; © Edicions UPC, 2002.

En este contexto concreto, se supone que el tiempo de descarga de un palet es nulo (despreciable respecto al resto de tiempos del modelo), y se consideran tan sólo dos condiciones que deben cumplirse para que pueda producirse una operación de descarga:

- presencia de un toro frente al camión
- existencia de *palets* en el camión para ser descargados.

Las función de arco $W(P_1,T_1)=n$ indica que el toro descargará todos los palets (marcas) del camión (lugar P_1) cuando se produzca el evento de descarga (transición T_1) sin consumir tiempo entre descarga de *palets* consecutivos.

La figura 4.2 muestra una forma de programar esta operación en un entorno orientado a procesos, en el cual tan sólo son necesarios 2 bloques (funciones) que se ejecutan cada vez que reciben una entidad (marca) a través de la conexión de entrada al bloque (punto de conexión izquierdo de la caja). Al llegar el toro se envía una entidad al primer bloque ("mirar si cola vacía"), el cual comprueba si el camión no está vacío mediante la evaluación del número de elementos almacenados en una cola de espera (la carga del camión se modela como una cola de espera), en cuyo caso se extrae un *palet* del camión (entidad temporal de la cola) y se repite esta operación hasta que se vacía completamente el camión (cola).

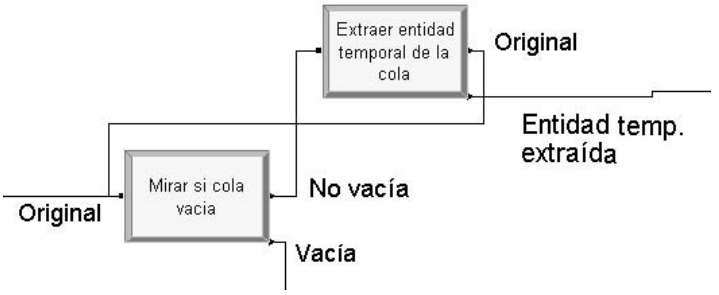


Figura 4.2 Código para la extracción de todas los elementos de una cola

Aunque la secuencialización de las operaciones "mirar si cola vacía" y "extraer entidad temporal de la cola" debería permitir vaciar la cola de *palets* asociada al camión, en realidad, si se ejecutan dichas funciones en un simulador, el resultado que se obtiene es que el camión nunca se vacía y el modelo entra en un bucle infinito debido a que se generan infinitos eventos que deben ser procesados sin poder avanzar el tiempo del simulador. El problema en este ejemplo concreto aparece porque en la salida del bloque "extraer entidad temporal de la cola" aparecen dos entidades temporales: el *palet* extraído (entidad temporal extraída) el cual se envía a otra parte del modelos, y el toro (entidad original que ha sido enviada al bloque) el cual se envía de nuevo al bloque "mirar si cola vacía" para repetir el ciclo de descarga en el caso que todavía hayan *palets* en el camión. El simulador, que ejecuta secuencialmente ambas operaciones en el mismo instante de tiempo de simulación t_x (ninguno de ambos bloques consume tiempo de simulación), genera primero la salida de la entidad temporal original (toro) del bloque "extraer entidad- temporal de la cola" y después quita un elemento de la cola. Al generar primero la entidad que representa el toro se activa otra vez el bloque "mirar si cola

© Los autores, 2002; © Edicions UPC, 2002.

vacía", el cual vuelve a enviar el toro al bloque "extraer entidad temporal de la cola" sin que el ciclo anterior haya quitado la primera entidad de la cola. Por tanto, el programa se queda bloqueado en un bucle infinito.

Hay varias alternativas para resolver este problema: una de ellas es, por ejemplo, dar un tiempo no nulo, tan pequeño como se desee, a la operación de descarga de *palets*. Sin embargo, el aspecto realmente importante es que es necesario entender los mecanismos internos de un simulador para poder detectar el problema y posteriormente corregirlo.

Son muchas las situaciones en las que "un simulador no hace lo que nosotros queremos", o bien "el simulador se *cuelga* porque tiene *bugs*". En algunos casos, es posible que el modelo desarrollado no contemple algunos aspectos, por lo que será necesario su modificación. Esta tarea se facilita considerablemente si el modelo ha sido especificado previamente en un formalismo como es el de las RdPC. En cambio, también pueden aparecer problemas al codificar un modelo conceptual correcto, debido a una insuficiente funcionalidad ofrecida por el simulador o a los mecanismos internos implementados en el simulador para la gestión de eventos y actualización del tiempo de simulación.

4.1 Elementos de un modelo de simulación

En el capítulo 1 se han introducido los elementos más significativos de un modelo de simulación. En este apartado, se repasan y amplían las principales características y propiedades de dichos elementos, los cuales son de interés para introducir los mecanismos que gestionan los eventos de los modelos de simulación. Para facilitar la comprensión de estos conceptos se utilizará el formalismo de redes de Petri a pesar de que dicho formalismo no sea propio a las distintas metodologías que utilizan los simuladores para ejecutar los modelos de simulación.

4.1.1 Entidades, entidades temporales y recursos

Las *entidades* son el conjunto de componentes del sistema, tales como máquinas, piezas, equipos de transporte o clientes. En la red de Petri las entidades corresponden a las marcas. Las entidades se pueden agrupar en dos tipos:

- *Recursos* o *entidades permanentes*: tal y como su nombre indica, su principal característica es que son elementos estáticos en cuanto que su número no aumenta ni disminuye a lo largo de la simulación. En general suelen utilizarse para describir los medios gracias a los cuales se pueden ejecutar las actividades. Los recursos definen quién o qué ejecuta la actividad, y suelen parametrizarse por características tales como capacidad, velocidad o tiempo de ciclo. Ejemplos de recursos son las máquinas, los elementos de transporte o los operadores.
- *Entidades temporales*: su principal característica es que se crean y destruyen a lo largo de la simulación. En general suelen utilizarse para describir los objetos que se procesan en el sistema, como por ejemplo las piezas, los clientes o los documentos. Las entidades temporales son los objetos que llegan, se procesan y salen del sistema. En la bibliografía, a menudo se emplea el concepto de *entidad* para referenciar sólo a las entidades temporales, en este libro el concepto es más amplio dado que también incluye los recursos para mantener la coherencia con los múltiples formalismos empleados.

© Los autores, 2002; © Edicions UPC, 2002.

En la red de Petri no se diferencian de forma explícita los recursos y las entidades temporales aunque no constituye ninguna dificultad diferenciarlas a partir de la topología de la red.

EJEMPLO 4.2 Proceso simple

La RdP de la figura 4.3 se representa un proceso simple compuesto básicamente por una estación de trabajo con dos máquinas, el cual ha sido descrito en el ejemplo 2.1. Las piezas que llegan para ser procesadas se esperan en una única cola FIFO (*first-in-first-out*), y una vez procesadas abandonan el sistema.

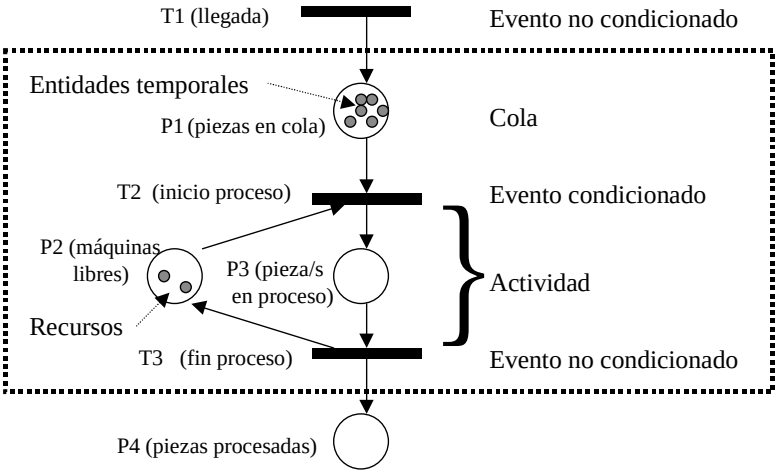


Figura 4.3 RdP de la estación de trabajo

Nótese que las marcas que representan las entidades permanentes (máquinas) siempre están en la red; en cambio, las marcas que representan las entidades temporales (piezas) fluyen a través de los recursos del sistemas, y desaparecen cuando salen de éste.

4.1.2 Atributos

Los atributos (colores en el formalismo de las RdPC) permiten caracterizar a las entidades; cada atributo corresponde a una propiedad (por ejemplo, precio, prioridad, tamaño), así como toda aquella información que fluye en el sistema junto con las entidades temporales (posición donde se encuentra la pieza, resultados de los controles de calidad, etc). Los atributos son imprescindibles para el control del flujo de entidades en el sistema, así como también para la extracción de entidades de una cola, por ejemplo, la extracción se puede hacer por orden del atributo prioridad.

4.1.3 Actividades

Las actividades son las tareas o acciones que tienen lugar en el sistema. Casi todas las actividades involucran a más de un tipo de entidad y, por tanto, el inicio de la actividad está condicionado a la presencia de todos los tipos de entidades involucrados. Un aspecto esencial de las actividades es su duración, la cual, a pesar de no ser un valor constante (proceso estocástico), tiene que ser conocida para que el simulador pueda determinar el instante en el cual la actividad finalizará. Nótese que en el

© Los autores, 2002; © Edicions UPC, 2002.

caso de actividades cuya duración no es constante, el instante de tiempo de finalización de la actividad puede determinarse a partir de la distribución de probabilidad que modela la duración prevista. Ejemplos de actividades son la reparación de una máquina, el procesado de una pieza o el transporte de un cliente.

En el ejemplo 4.2 anterior se muestra la RdP de una simple estación de trabajo. La actividad se inicia cada vez que en el sistema hay una entidad permanente (recurso máquina) libre y un entidad temporal (pieza) disponible para ser procesada. Nótese que el inicio y/o fin de una actividad corresponde con el disparo de una transición en la red de Petri.

4.1.4 Eventos

En los modelos de eventos discretos las variables de estado (toda la información asociada al sistema necesaria para evaluar su comportamiento) tan sólo pueden cambiar de valor en instantes de tiempo ligados a la aparición de un evento. Por tanto, un evento se puede definir como una acción instantánea (que no consume tiempo) y que puede cambiar el valor de una variable de estado del sistema modelado.

Normalmente, más de una actividad se inicia o finaliza con cada evento. Una primera clasificación nos permite diferenciar dos tipos de eventos:

- *Eventos condicionados*: para que se activen es necesario que se cumplan una o más condiciones.
- *Eventos no condicionados*: eventos que están planificados para su ejecución y que no dependen de condiciones.

Los eventos también se pueden clasificar en:

- *Eventos endógenos o internos*: son eventos causados por condiciones en el modelo, por ejemplo, la finalización de una operación.
- *Eventos exógenos o externos*: eventos externos al modelo, como por ejemplo la llegada de una pieza al modelo.

En la RdP del ejemplo 4.2 se distinguen tres eventos:

1. Asociado a la transición T1: evento no condicionado y externo que provoca la llegada de piezas al modelo.
2. Asociado a la transición T2: evento condicionado e interno cuya activación provoca el inicio de la actividad ligada al proceso de la pieza. Para ejecutarse este evento es necesario que una de las dos máquinas esté libre y que haya una o más piezas en la cola. Por tanto, después de la llegada de una pieza (evento ligado a la transición T1) o después del evento de fin de proceso (transición T3) se evaluará si se puede ejecutar el evento condicionado de inicio de proceso (transición T2). En la RdP aparecen claramente definidos los eventos condicionados: son los ligados a las transiciones con más de un arco de entrada
3. Asociado a la transición T2: evento no condicionado e interno cuya activación finaliza la actividad tiempo de proceso. Este evento no está condicionado, dado que cuando se inicia la actividad (transición T2) se planifica el tiempo de finalización.

Es muy importante darse cuenta de que tanto el estado global del modelo, como el estado de cada una de sus entidades (atributos), tan sólo pueden cambiar con la presencia de eventos. Esta característica

© Los autores, 2002; © Edicions UPC, 2002.

es la que permite a los simuladores avanzar el tiempo de simulación y gestionar los eventos de tal modo que la evolución de los valores de las variables de estado describan el comportamiento del sistema en estudio.

En la figura 4.4 se ha representado gráficamente la evolución del número de piezas en la cola y el número de máquinas libres (variables de estado) del sistema descrito en el ejemplo 4.2. Puede observarse que el número de piezas en la cola de espera tan sólo puede variar ante la presencia de eventos de tipo T1 (llegada de piezas al sistema), los cuales se producen en los instantes t_3, t_4 y t_9, y ante la presencia de eventos de tipo T2 (inicio de proceso), el cual se ha producido en los instantes t_5 , t_6 y t_{10}. Por otro lado, el número de máquinas libres tan sólo puede variar ante la presencia de un evento de tipo T2 (inicio de proceso), o bien de tipo T3 (fin de proceso).

Nótese que las variables de estado permanecen constantes entre eventos (el estado del sistema permanece constante entre los instantes), y su valor tan sólo puede cambiarse como efecto de la aparición de un evento.

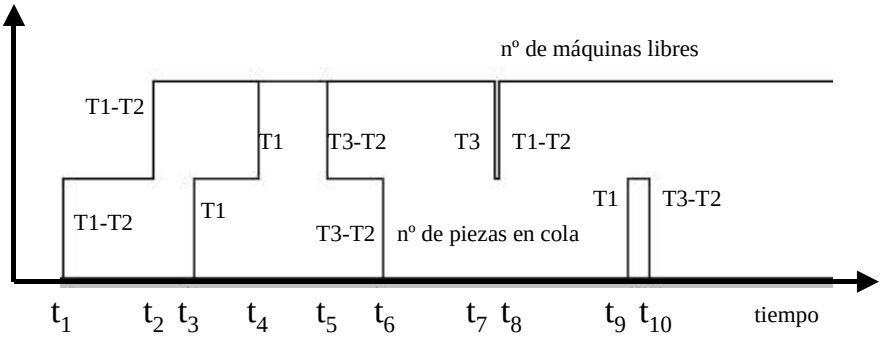


Figura 4.4 Evolución de las variables de estado del sistema

4.1.5 Colas

Estas estructuras quedan determinadas a partir de una colección de entidades (en general entidades temporales), ordenadas de una forma lógica. Por ejemplo, clientes en una cola de espera, ordenadas de forma FIFO. Las entidades que están en las colas sufren un retardo de duración indeterminada. En las redes de Petri no aparecen las colas de forma explícita pero, en general, aquellos lugares ocupados por entidades temporales que están a la espera de recursos restringidos suelen codificarse como colas de espera. En el ejemplo 4.2 el lugar P_1 se puede asociar a una cola.

EJEMPLO 4.3 Línea de fabricación flexible

En la figura 4.5 se presenta el sistema descrito en en ejemplo 2.4 del capítulo 2. El sistema está formado por las siguientes unidades de producción: dos robots, R_1 y R_2 ; tres máquinas CNC, M_1, M_2 y M_3, y dos unidades de almacenamiento, S_1 y S_2.

© Los autores, 2002; © Edicions UPC, 2002.

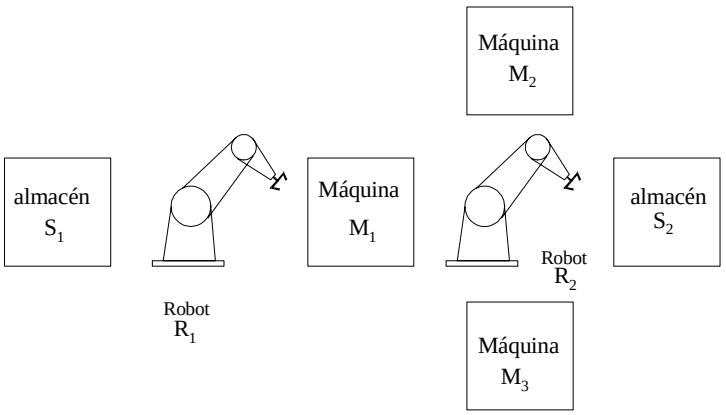


Figura 4.5 Línea flexible de fabricación

Recordemos el funcionamiento de la línea de producción:

5. El robot R_1 realiza el transporte de las piezas almacenadas en S_1 y las carga en la máquina M_1.
6. La máquina M_1 realiza la primera operación sobre las piezas.
7. Una vez la máquina M_1 ha finalizado la primera operación, el robot R_2 transporta la pieza procesada en M_1 y la carga en la máquina M_2 o en la máquina M_3.
8. Una vez la máquina M_2 o la máquina M_3 ha finalizado la segunda operación, el robot R_2 transporta la pieza finalizada al almacén de salida S_2.

La RdP del modelo se representa en la figura 4.6. En está se pueden identificar los siguientes elementos:

- *Entidades temporales y recursos*: las entidades temporales son las piezas que inicialmente están en el lugar P_6 y que fluyen hasta el almacén de salida (P_{18}). En este caso concreto, al no considerarse un evento de llegada de piezas al sistema, ni tampoco un evento de salida de piezas ya procesadas, el número de entidades temporales en el modelo se mantiene constante a lo largo de toda la simulación. Los recursos son los robots R1 y R2, y las máquinas M1, M2 y M3. Estos recursos aparecen explícitamente cuando están libres en los lugares P_3, P_8, P_1, P_7 y P_{13}.

- *Actividades*: aparecen en los lugares P_2, P_4, P_9, P_{10}, P_{11}, P_{12}, P_{15}, P_{16}, P_{17}. Los lugares P_2, P_9, P_{12}, P_{14} y P_{17} son actividades que involucran a los robots, y los lugares P_4, P_5, P_0, P_{11}, P_{15} y P_{16} son actividades que involucran a las máquinas.

- *Eventos*: no aparece en el modelo ningún evento externo. Los eventos condicionados son los ligados a las transiciones T_1, T_4, T_7, T_9 y T_{12}. El resto de eventos no son condicionados y corresponden al final de las actividades.

- *Colas*: corresponden a los lugares P_5, P_6, P_{18}, en donde es posible que las piezas se esperen por falta de recursos disponibles (libres).

© Los autores, 2002; © Edicions UPC, 2002.

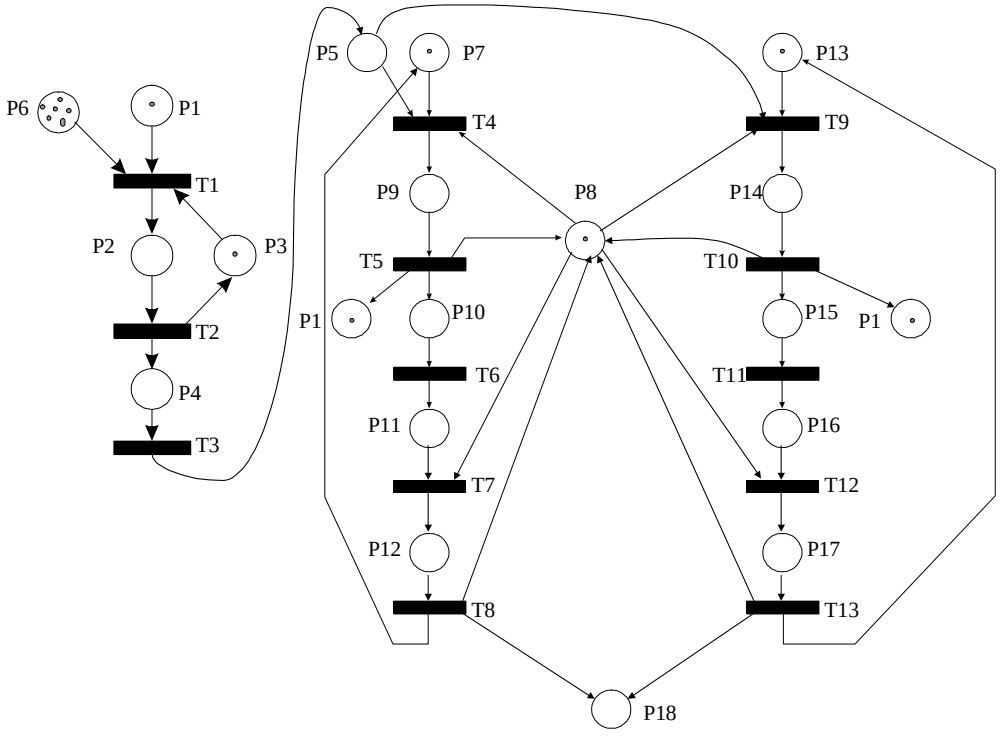


Figura 4.6 RdP de la linea de producción

4.2 Simulación manual y medidas de comportamiento

En este apartado se presentan los mecanismos básicos que debe implementar un simulador, a través de una simulación manual. El ejemplo que vamos simular corresponde a la estación de trabajo compuesta por dos máquinas descrito en el ejemplo 2.1 y en el 4.2 de este capítulo. Los elementos más importantes del modelo son:

- *Entidades temporales y recursos*: las entidades temporales son las piezas y los recursos son las dos máquina idénticas.
- *Actividades*: la única actividad del modelo es la asociada al lugar P_3. En la literatura, a veces también se asocia una actividad al proceso de llegadas representado por la transición T_1.
- *Eventos*: los eventos no condicionados son los asociados a las transiciones T_1 y T_3. El evento asociado a la transición T_2 es condicionado.
- *Colas*: encontramos una cola asociada al lugar P_1.

Se desea simular el proceso suponiendo que el tiempo entre llegadas sigue una distribución exponencial de media 10 minutos (transición T1) y el tiempo de proceso sigue una distribución uniforme ente 15 y 25 minutos (U[15,25]).

© Los autores, 2002; © Edicions UPC, 2002.

Para poder simular el proceso es necesario generar una muestra de valores de las distribuciones de probabilidad anteriores. Los simuladores extraen un valor de la distribución cada vez que lo necesitan pero, para simular a mano, es mucho más cómodo obtener una muestra suficientemente grande de valores de cada distribución.

Tanto si la simulación se realiza manualmente como si la realiza un simulador, debe preservarse el concepto de repetitividad, el cual consiste en generar los mismos valores aleatorios y, en consecuencia, generar los mismos resultados cada vez que se realiza la misma simulación. Para ello, en lugar de generar los datos a partir de funciones de distribución, los datos aleatorios se generarán en 2 etapas: en la primera se genera, mediante el algoritmo de congruencias lineales un valor que sigue una distribución U[0,1], y a continuación se aplica el método de la transformada inversa a la distribución de interés.

La función de distribución acumulativa exponencial de media β es

$$Y = F(x) = \begin{cases} 1 - e^{-x/\beta} & x \geq 0 \\ 0 & \text{x} < 0 \end{cases}$$

y la inversa de la función de distribución exponencial acumulativa

$$x = F^{-1}(Y) = -\beta * \ln(1-Y) = -10 * \ln(1-Y)$$

Así pues, asignando valores U[0,1] a Y, es posible obtener valores que siguen la distribución exponencial empleando la función inversa tal como muestra la tabla 4.1, en la cual se recogen: en la primera fila, los valores aleatorios U[0,1] generados mediante el algoritmo de congruencias lineales; en la segunda fila, el tiempo entre llegadas, el cual responde a una distribución exponencial y ha sido generado mediante la ecuación anterior. Finalmente, la tercera fila recoge el tiempo absoluto de llegada de once piezas sucesivas, obtenido directamente a partir de los tiempos entre llegadas de la fila anterior, suponiendo que la primera pieza llega en el instante 0.

U[0,1]		0.10	0.74	0.32	0.35	0.87	0.28	0.36	0.11	0.26	0.93
Tiempo entre llegadas		1.5	14.2	3.4	3.7	19.5	2.9	3.9	1.5	2.7	25.8
Tiempo de llegada	0	1.5	15.7	19.1	22.8	42.3	45.2	49.1	50.6	53.3	79.1

Tabla 4.1 Tiempo de llegada de piezas (evento T1)

De manera equivalente, la función de distribución uniforme U[a,b] acumulativa es:

$$Y = F(x) = \begin{cases} 0 & x < a \\ \dfrac{x-a}{b-a} & a \leq x \leq b \\ 1 & b < x \end{cases}$$

© Los autores, 2002; © Edicions UPC, 2002.

y la inversa de la función de distribución acumulativa es:

$$x = F^{-1}(Y) = a + Y(b-a) = 15 + Y * 10$$

A partir de una muestra de valores U[0,1] y empleando la ecuación anterior se obtiene la siguiente lista de tiempos de proceso.

U[0,1]	0.52	0.44	0.51	0.10	0.45	0.91	0.31	0.42	0.67	0.05
Tiempo de proceso	20.2	19.4	20.1	16	19.5	24.1	18.1	19.2	21.7	15.5

Tabla 4.2 Tiempo de proceso (actividad P3)

La tabla 4.3 muestra la simulación manual del proceso a partir de la información anterior. En la primera columna se recogen los instantes de tiempo en los que se producen la aparición de los eventos en orden cronológico:

Reloj	Eventos	P1	P2	P3	P4	Tiempo P3 empieza (T2)	Duración P3	Tiempo P3 finaliza (T3)
0	T1-T2	0	1	1	0	0	20.2	20.2
1.5	T1-T2	0	0	2	0	1.5	19.4	20.9
15.7	T1	1	0	2	0			
19.1	T1	2	0	2	0			
20.2	T3-T2	1	0	2	1	20.2	20.1	40.3
20.9	T3-T2	0	0	2	2	20.9	16	36.9
22.8	T1	1	0	2	2			
36.9	T3-T2	0	0	2	3	36.9	19.5	56.4
40.3	T3	0	1	1	4			
42.3	T1-T2	0	0	2	4	42.3	24.1	66.4
45.2	T1	1	0	2	4			
49.1	T1	2	0	2	4			
50.6	T1	3	0	2	4			
53.3	T1	4	0	2	4			
56.4	T3-T2	3	0	2	5	56.4	18.1	74.5
66.4	T3-T2	2	0	2	6	66.4	19.2	85.6

Tabla 4.3 Simulación manual basada en la programación de eventos.

En la segunda columna aparecen los eventos que se activan. La aparición de dos eventos en el mismo campo es debido a que la ejecución de un evento no condicionado activa un evento condicionado. Por ejemplo T_1-T_2 indica que la llegada de una pieza al sistema provoca el inicio del proceso, el cual no había podido activarse a pesar de tener 1 ó 2 máquinas libres, dado que no había ninguna pieza en la cola. La columna P_1 recoge el número de piezas en la cola pendientes de ser procesadas; P_2 el número de máquinas libres; P_3 el número de máquinas trabajando y P_4 el número de piezas procesadas. Las tres columnas siguientes dan el tiempo de inicio de la actividad proceso de la máquina, su duración y el tiempo de finalización.

© Los autores, 2002; © Edicions UPC, 2002.

A esta estrategia de simulación se la conoce como “simulación basada en la programación de eventos” dado que su enfoque consiste en mostrar cómo evoluciona el modelo a medida que se van activando cronológicamente todos los eventos. Otra estrategia de simulación distinta consiste en mostrar la evolución de las entidades temporales en el modelo. Por tanto, adopta el punto de vista de la entidad que se mueve y sus interacciones con el resto de entidades temporales del modelo. Esta segunda estrategia es conocida como “simulación basada en la interacción de procesos” y se llama *ciclo de vida o proceso de una entidad temporal* a toda su evolución a través del modelo.

No. de pieza	T1	T2	Duración P3	T3	Retardo en P1 = T2-T1
1	0	0	20.2	20.2	0
2	1.5	1.5	19.4	20.9	0
3	15.7	20.2	20.1	40.3	4.5
4	19.1	20.9	16	36.9	1.8
5	22.8	36.9	19.5	56.4	14.1
6	42.3	42.3	24.1	66.4	0
7	45.2	56.4	18.1	74.5	11.2
8	49.1	66.4	19.2	85.6	17.3
9	50.6	74.5	21.7	96.2	23.9
10	53.3	85.6	15.5	101.1	32.3

Tabla 4.4 Simulación manual basada en la interacción de procesos.

En la tabla 4.4 se muestra la simulación manual basada en la interacción de procesos. Se muestra para cada pieza el tiempo de llegada (T_1), el tiempo de inicio de la actividad proceso de la pieza (T_2), la duración del procesado de la pieza (P_3), cuándo finaliza el proceso (T_3) y el retardo de la pieza en la cola (P_1).

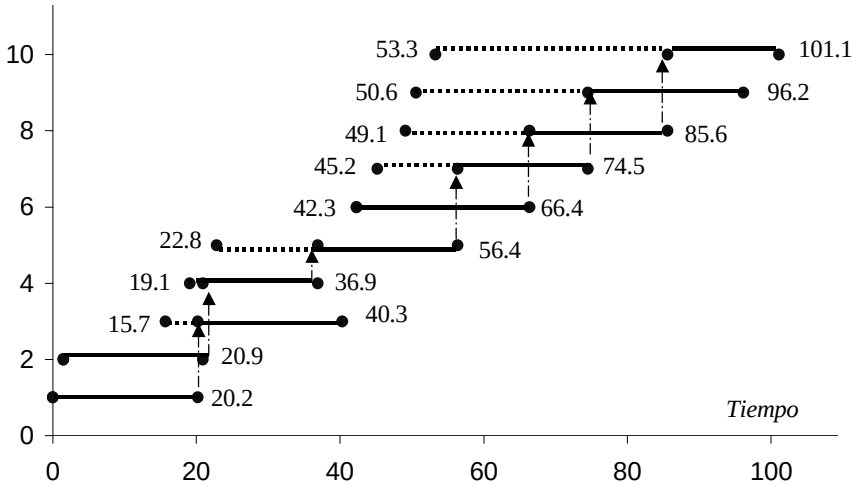


Figura 4.7 Simulación manual basada en la interacción de procesos.

© Los autores, 2002; © Edicions UPC, 2002.

En la figura 4.7 se muestra un diagrama en el que aparece la evolución de las entidades temporales y las interacciones entre ellas. Por ejemplo, la finalización del proceso de la pieza número 6 permite que se inicie el procesado de la pieza número 8.

El objetivo final de un estudio de simulación no es simular el proceso. La simulación, de por sí, no tienen ninguna utilidad. Es, por tanto, necesaria la obtención de estadísticas que permitan evaluar o medir cómo se comporta el modelo de simulación y ayudar en la toma de decisiones. En este ejemplo particular, posibles datos de interés que se podrían estimar serían, por ejemplo, el retardo promedio de los clientes en la cola, el tamaño promedio de la cola o el grado de utilización de las máquinas. A estas medidas de comportamiento se las denomina *estimadores* dado que su valor depende del comportamiento aleatorio del modelo. Los resultados de simulaciones diferentes (cambiando la semilla del algoritmo de generación de números aleatorios) generarán estimaciones diferentes, de la misma forma que el comportamiento real de un proceso sujeto al azar no es exactamente el mismo dos días diferentes. Por tanto, estos estimadores también son variables aleatorias [LAW91].

4.2.1 Retardo promedio en la cola

Dada una simulación con *n* clientes procesados (piezas en el ejemplo anterior), un estimador del retardo promedio esperado es

$$r(n) = \frac{\sum_{i=1}^{n} R_i}{n}$$

donde R_i corresponde al tiempo de espera del cliente i. En realidad, el concepto de retardo promedio esperado es aquel que se encontraría si el número de clientes en el sistema es infinito (muy grande en la práctica). El retardo promedio estimado de las piezas del ejemplo anterior se puede obtener a partir de los retardos de cada una de las piezas, observado en la tabla 4.4 o en la figura 4.7,

$$r(10) = \frac{\sum_{i=1}^{10} R_i}{n} = \frac{0 + 0 + 4.5 + 1.8 + 14.1 + 0 + 11.2 + 17.3 + 23.9 + 32.3}{10} = 10.5 \text{ minutos}$$

Este estimador constituye un ejemplo de una estadística discreta en el tiempo (estimador discreto) dado que se define a partir de una colección indexada (Ri, i=1,...,n) de valores aleatorios. El estimador del retardo promedio en la cola tiene un interés especial en aquellos sistemas en los que la entidad que se maneja son personas. Por ejemplo, cuando se simula la calidad del servicio de un restaurante, un estimador de la calidad puede ser el retardo promedio para ser servido. Ejemplos parecidos se pueden encontrar en la mayoría de sistemas de servicios, en donde prima la calidad del servicio a las personas.

4.2.2 Tamaño promedio de la cola

Otro estimador de interés es el del tamaño promedio de la cola. El comportamiento de la cola tiene interés para el análisis de la mayoría de modelos orientados a eventos discretos. En una instalación productiva, una cola con muchas entidades suele indicar la saturación de un conjunto de procesos, y una cola pequeña, puede ser un indicio de recursos infrautilizados.

© Los autores, 2002; © Edicions UPC, 2002.

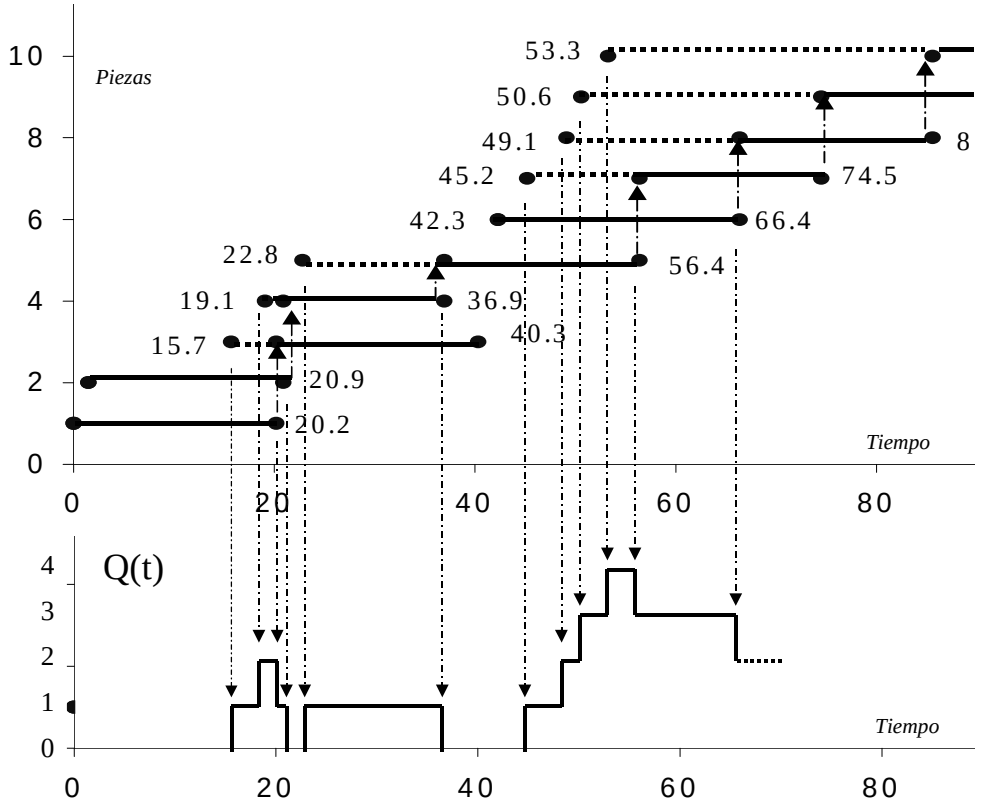


Figura 4.8 Evolución de la cola.

La figura 4.8 muestra la evolución Q(t) de la cola del proceso simulado. El estimador del tamaño promedio de la cola es

$$q(n) = \frac{\int_0^{T(n)} Q(t)dt}{T(n)}$$

donde T(n) puede ser, por ejemplo, el tiempo en el que el cliente número n sale del sistema. Dadas las características de Q(t), la ecuación anterior se puede calcular de manera equivalente como

$$q(n) = \frac{\int_0^{T(n)} Q(t)dt}{T(n)} = \frac{\sum_{i=0}^{\infty} iT_i}{T(n)}$$

donde T_i es el tiempo total que la cola tiene un tamaño i, es decir, $T(n)=T_0+T_1+T_2+T_3+T_4+...$ (ya que la suma de tiempos totales de la cola con tamaño i es el tiempo total de simulación). Este estimador

© Los autores, 2002; © Edicions UPC, 2002.

constituye un ejemplo de una estadística continua (estimador continuo) dado que la variable aleatoria Q(t) está definida sobre el intervalo de los números reales positivos.

Siguiendo con el ejemplo anterior, el tamaño promedio en la cola q(6) del ejemplo considerado es

$$T_0 = (15.7 - 0) + (45.2 - 36.9) = 24$$
$$T_1 = (19.1 - 15.7) + (20.9 - 20.2) + (36.9 - 22.8) + (49.1 - 45.2) = 22.1$$
$$T_2 = (20.2 - 19.1) + (50.6 - 49.1) = 2.6$$
$$T_3 = (53.3 - 50.6) + (66.4 - 56.4) = 12.7$$
$$T_4 = (56.4 - 53.3) = 3.1$$
$$q(n) = \frac{\sum_{i=0}^{4} iT_i}{T(n)} = \frac{0 \times 24 + 1 \times 22.1 + 2 \times 2.6 + 3 \times 12.7 + 4 \times 3.1}{66.4} = 1.17$$

Por tanto, el tamaño promedio de la cola es de 1.17 piezas y la ocupación máxima es de 4 piezas.

4.2.3 Grado de utilización de los recursos

Otro estimador continuo de interés es el grado de utilización de las máquinas (servidores en general). La información aportada es complementaria a la del tamaño promedio en la cola. Una ocupación elevada puede ser un indicio de colas de espera con muchas entidades y, consecuentemente, de la necesidad de incorporar más máquinas. Una ocupación baja puede indicar un exceso de máquinas.

La figura 4.9 muestra la evolución U(t) del número de máquinas ocupadas. El estimador del grado de utilización de m máquinas es

$$u(n) = \frac{\int_0^{T(n)} U(t)\,dt}{T(n)} = \frac{\sum_{i=0}^{m} iU_i}{T(n)}$$

donde U_i es el tiempo en el que el conjunto de recursos tiene i recursos ocupados.

Siguiendo con el ejemplo anterior, el grado de utilización u(6) del ejemplo considerado con 2 máquinas (m=2) es:

$$T_0 = 0$$
$$T_1 = (1.5 - 0) + (42.3 - 40.3) = 3.5$$
$$T_2 = (40.3 - 1.5) + (66.4 - 42.3) = 62.9$$
$$u(6) = \frac{0 \times 0 + 1 \times 3.5 + 2 \times 62.9}{66.4} = 1.95$$

© Los autores, 2002; © Edicions UPC, 2002.

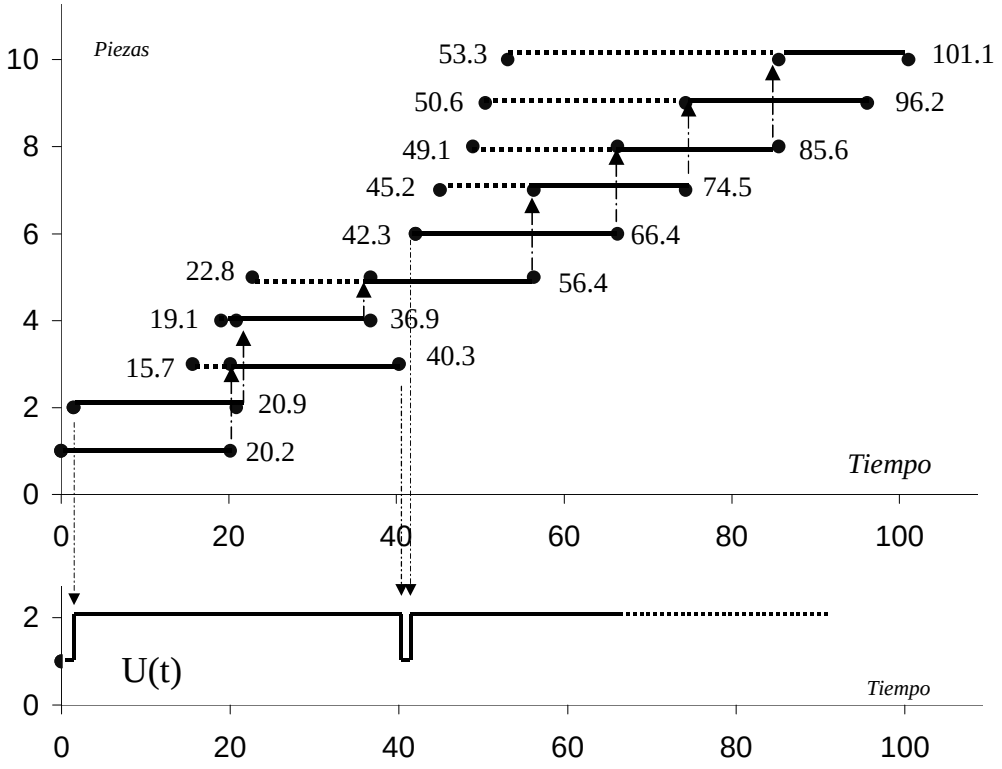


Figura 4.9 Utilización de las máquinas.

4.2.4 Otras estadísticas de interés

Adicionalmente a los estimadores presentados, existe un conjunto importante de datos que suelen utilizarse para evaluar distintos grados de rendimiento del sistema que se está simulando, con el objetivo de poder determinar aquellos factores (cuellos de botella) que están apartando el comportamiento real del sistema del comportamiento óptimo para el cual había sido diseñado. Otras medidas de comportamiento de interés en el ámbito de la fabricación son, por ejemplo:

- *Throughput* del sistema de producción: número de piezas producidas por unidad de tiempo.
- Tiempo de permanencia en el sistema de las piezas: se considera el tiempo desde que entran al sistema hasta que salen (*manufacturing lead time*).
- Número de piezas que se encuentran dentro del sistema de producción (bien en unidades de transporte, bien en colas de espera, bien en unidades de producción) en un instante determinado (*work in process*).
- Utilización de máquinas: relación del tiempo durante el cual una máquina está procesando respecto al tiempo total de simulación.
- Tiempo total que las piezas se esperan para el transporte.
- Tiempo total de las piezas en las colas.
- Proporción de piezas entregadas fuera de plazo.

© Los autores, 2002; © Edicions UPC, 2002.

4.3 Elementos de un simulador

Aunque existen diversas estrategias que pueden implementar los simuladores para ejecutar los modelos de simulación orientados a eventos discretos, la mayor parte tienen implementados los siguientes módulos:

- *Generador de números aleatorios U[0,1]*: es el módulo base para la generación de valores aleatorios uniformemente distribuidos [0,1].
- *Generadores de variables aleatorias*: a partir de valores aleatorios U[0,1] y, en determinados casos, de la transformada inversa de la función de distribución de interés, se obtienen los valores aleatorios que siguen las distintas funciones de distribución.
- *Reloj de la simulación*: variable que almacena el tiempo de la simulación.
- *Lista de eventos futuros (LEF):* lista que almacena los instantes de tiempo cuando deben activarse los eventos no condicionados. Los eventos de la lista están ordenados por tiempo de activación.
- *Módulos para la generación de estadísticas:* conjunto de variables y rutinas para obtener las medidas de comportamiento de interés.
- *Módulo de inicialización*: ejecuta el conjunto de tareas necesarias para la inicialización del modelo y del simulador. En general, suele inicializar la lista de eventos, el reloj del simulador, las variables de estado, así como las variables del módulo de estadísticas.
- *Rutina de actualización del reloj del simulador*: subprograma que determina el instante de tiempo en el que debe activarse el siguiente evento, así como el evento que debe ser disparado. Esta información la obtiene de la lista de eventos futuros.
- *Rutina de tratamiento de evento*: existe una rutina de tratamiento de evento para cada tipo de evento (para cada transición de la RdP), la cual actualiza los valores de algunas de las variables de estado y puede generar nuevas entradas a la lista de eventos futuros.
- *Código ejecutivo*: controla la ejecución de la simulación. En el apartado siguiente se detallan las responsabilidades del código ejecutivo en función de la estrategia de simulación adoptada.
- *Estado del sistema*: colección de variables necesarias para describir el sistema en cualquier instante de tiempo. Cuando el modelo ha sido formalizado mediante redes de petri coloreadas, las variables de estado suelen almacenar toda la información asociada tanto a las entidades permanentes como temporales.

En la figura 4.10 se ilustra el flujo de control entre los módulos de un simulador orientado a eventos discretos.

© Los autores, 2002; © Edicions UPC, 2002.

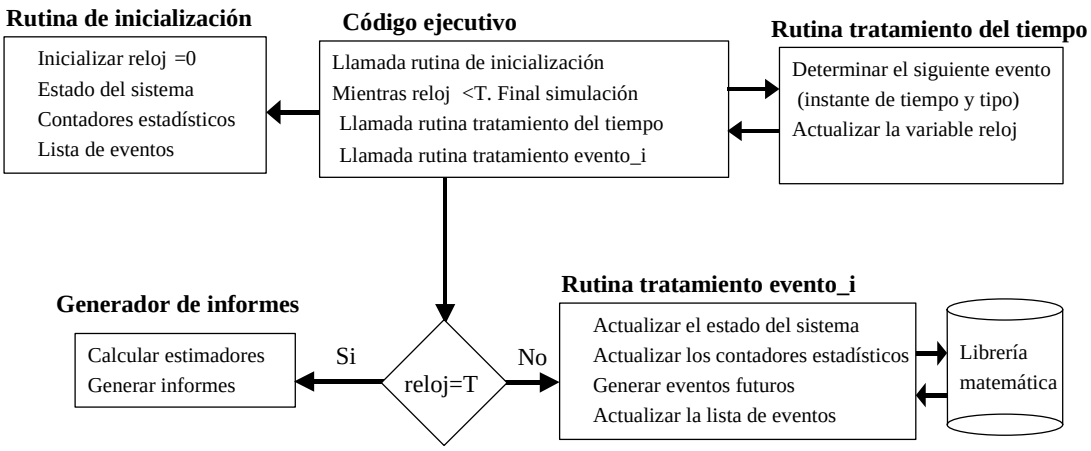


Figura 4.10 Flujo de control entre los elementos de un simulador orientado a eventos discretos

Al inicio de cada simulación, el código ejecutivo pasa el control a la rutina de incialización, para que ésta inicialice el reloj del simulador, las variables de estado del sistema, la lista de eventos, así como los contadores estadísticos. A partir de este momento, el código ejecutivo pasa el control a la rutina de tratamiento de tiempo para que ésta evalúe el tipo del siguiente evento a ejecutar, así como el instante de tiempo en que debe ser ejecutado (actualización del reloj del simulador). Una vez finalizada la rutina de tratamiento del tiempo, el código ejecutivo pasa el control a la rutina de tratamiento del evento, la cual actualiza las variables de estado de acuerdo con el tipo de acción que se realizaría sobre el sistema físico, genera los instantes de tiempo de los futuros eventos que se desencadenan como efecto de la aparición del presente evento, y almacena dicha información en la lista de eventos futuros. Una vez se ha finalizado la ejecución del código asociado al evento, el código ejecutivo comprueba si el reloj del simulador ha superado el tiempo máximo de simulación, en cuyo caso se pasa el control al generador de informes y se finaliza la simulación. Por otro lado, si el reloj del simulador es inferior al tiempo máximo de simulación, el código ejecutivo vuelve a pasar el control a la rutina de tratamiento del tiempo y a la rutina de tratamiento del evento de manera cíclica, hasta superar el tiempo máximo de simulación.

Cabe notar que la actualización del reloj de simulación al siguiente instante de tiempo se realiza sin consultar ni modificar ninguna variable de estado. Una característica básica de los sistemas orientados a eventos discretos es que el estado del sistema tan sólo puede presentar cambios como efecto de la aparición de un evento y permanece constante el resto de tiempo.

Para ilustrar este concepto, considérese un proceso de llegadas determinista (cada 8 minutos) a un servidor con un tiempo de proceso determinista de 10 minutos. En la figura 4.11 se han representado los instantes de tiempo en los cuales se actualizará el reloj del simulador y se ejecutará la rutina de tratamiento de evento asociada al proceso de llegada (evento e_1) y al de inicio de proceso (evento e_2). Puesto que se trata de un sistema orientado a eventos discreto, el estado del sistema permanece constante entre eventos, lo cual evita tener que actualizar el reloj del simulador con incrementos regulares.

© Los autores, 2002; © Edicions UPC, 2002.

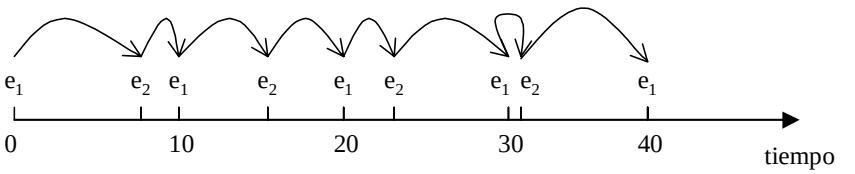


Figura 4.11 Instantes de tiempo en los cuales debe actualizarse el reloj del simulador

En la siguiente sección se presentan las tres principales estrategias para actualizar el reloj del simulador, teniendo en cuenta que los tiempos en los que aparecen los eventos no son deterministas, sino que se modelan utilizando variables aleatorias.

4.4 Estrategias de simulación

Una de las características más determinante de las herramientas de simulación es su *visión del mundo* [BARC96] o estrategia para la selección del siguiente evento. En el apartado 4.2 se han introducido dos de las tres estrategias que aquí se presentan:

- Programación de eventos (*Event Scheduling*)
- Exploración de actividades (*Activity Scanning*)
- Interacción de Procesos (*Process Interaction*)

En la actualidad, la mayoría de simuladores implementan varias estrategias de simulación con el objetivo de aprovechar el potencial de cada una de ellas.

4.4.1 Programación de eventos (*Event Scheduling*)

Los eventos son acciones instantáneas que pueden cambiar el estado del modelo. Estos instantes coinciden normalmente con el inicio o final de las actividades (por ejemplo, el inicio del transporte de una pieza y el final del transporte). La principal característica de esta estrategia de simulación es que tan sólo pueden codificarse las rutinas de tratamiento de eventos no condicionados. Puesto que los sistemas que se quieren simular presentan tanto eventos no condicionados como eventos condicionados, el código asociado a estos últimos se deberá incluir en el código de tratamiento de eventos no condicionados.

Después de inicializar el modelo, el código ejecutivo explora todos los futuros eventos no condicionados y mueve el reloj de la simulación hasta el evento no condicionado más cercano en el tiempo (el próximo evento). A continuación, llama al código asociado al evento para efectuar los cambios en el modelo provocados por el evento. Estas acciones incluyen efectuar los cambios derivados del evento, que pueden corresponder al inicio de una actividad y a la planificación de un evento no condicionado asociado al final de la actividad.

El mecanismo para garantizar que todos los eventos se ejecutan en orden cronológico se basa en el mantenimiento de la lista de eventos futuros (LEF). Esta lista contiene una entrada para todos los eventos no condicionados que tienen planificada su ejecución en el futuro. En la realidad, la mayoría de eventos no condicionados no se planifican, simplemente ocurren: por ejemplo la rotura de una

© Los autores, 2002; © Edicions UPC, 2002.

máquina o el fin del servicio a un cliente a una entidad bancaria; en el modelo, estos eventos representan el fin de una actividad cuya duración está representada por una distribución estadística.

La longitud y el contenido de la LEF cambian continuamente a medida que la simulación progresa y, por tanto, una buena gestión de la lista es fundamental para asegurar la eficiencia de la simulación. Las operaciones principales que se realizan en la lista son la eliminación del evento no condicionado que se acaba de ejecutar (el primero de la lista), la adición de un evento no condicionado y, ocasionalmente, la eliminación de un evento no condicionado.

Algoritmo

En la tabla siguiente se muestra el pseudocódigo ejecutivo para un algoritmo de programación de eventos. El primer bloque de código corresponde a la inicialización de la simulación. Los aspectos principales que se deben inicializar son:

- El tiempo de inicio y final de la simulación respecto al tiempo 0 de referencia.
- El reloj de la simulación al tiempo de inicio.
- Cálculos para obtener el estado inicial.
- La LEF, almacenando en ella los eventos no condicionales iniciales.
- Extraer el primer evento de la LEF y avanzar el reloj hasta el tiempo de este evento.

El segundo bloque corresponde al código repetitivo consistente en ejecutar el código asociado al evento extraído, extraer el nuevo evento más inminente de la LEF y en avanzar el reloj de la simulación hasta el tiempo de este nuevo evento.

```
Inicio ejecutivo
 Inicio módulo de inicialización
              Inicializar tiempo_inicio, tiempo_fin;
              Inicializar el reloj de la simulación al tiempo_inicio;
              Inicializar el estado inicial y los contadores estadísticos;
              Almacenar los eventos no condicionales iniciales en la LEF;
              Extraer el evento más inminente de la LEF;
              Avanzar el reloj hasta el tiempo del evento extraido;
       Fin inicialización;
       Inicio bucle de simulación
              Mientras el reloj no ha alcanzado el tiempo_fin
                     Ejecutar el código asociado al evento extraido
                     Extraer el evento más inminente de la LEF;
                     Avanzar el reloj hasta el tiempo del evento extraido;
              Fin mientras;
       Fin bucle de simulación;
 Generar un informe con las estadísticas de interés;
Fin ejecutivo;
```

Tabla 4.5 Código ejecutivo de la programación de eventos

Las tareas usuales que ejecuta el código para cada tipo de evento son:

© Los autores, 2002; © Edicions UPC, 2002.

- Actualizar el estado del sistema.
- Obtener información sobre el comportamiento del sistema para actualizar las estadísticas.
- Generar nuevos eventos no condicionados, actualizar la LEF y activar eventos condicionados. Los eventos no condicionados pueden ser, por ejemplo, el final de la actividad. Estos eventos precisan, generalmente, la obtención del tiempo futuro en que ocurrirá el evento a partir de las distribuciones de probabilidad. Si se activan eventos condicionados, la ejecución del evento se puede hacer desde este mismo bloque de código o se puede llamar a otro bloque de código (rutina) que maneja el evento condicionado.

```
Inicio código de evento
       Actualizar el estado del modelo;
       Actualizar los contadores estadísticos;
       Generar futuros eventos no condicionados e incluirlos en la LEF;
       Activar eventos condicionados en este instante de tiempo;
Fin código de evento;
```

Tabla 4.6 Código de evento de la programación de eventos

EJEMPLO 4.4 Ejemplo de la programación de eventos

Considérese de nuevo el sistema descrito en el ejemplo 4.2. En la figura 4.12 se muestra el organigrama del código asociado al evento de llegadas en el instante t (transición T_1 de la RdP). Nótese que como efecto de un proceso de llegada en el instante t, es posible evaluar el instante de tiempo en el que se producirá la siguiente llegada, puesto que el tiempo entre llegadas ha sido modelado estadísticamente mediante una fdp. Así pues, como efecto de la activación del evento no condicional T_1, es posible programar el instante de tiempo (variable "tel") en el que deberá ejecutarse de nuevo el código de la rutina de tratamiento de evento T_1.

Por otro lado, también puede observarse que en la rutina de tratamiento del evento T_1, también se ha incluido el código de la rutina de tratamiento del evento condicional T_2. Así pues, si como efecto de un proceso de llegadas (evento T_1) es posible (condición: existe una máquina libre) iniciar la actividad de procesamiento de la pieza (evento T_2), se ejecuta el código asociado a dicho evento, el cual consiste en incrementar la variable de estado que representa el número de máquinas ocupadas (variable "no"), y evaluar el instante de tiempo (variable "tes") en que finalizará la actividad, el cual se calcula mediante una llamada a la función de densidad de probabilidad U[15,25]. La primera acción, o la última, que se puede efectuar es calcular el tiempo de la siguiente llegada (evento no condicionado), resultado de sumar el tiempo actual al valor aleatorio Xl, obtenido éste a partir de la distribución de probabilidad del tiempo entre llegadas.

© Los autores, 2002; © Edicions UPC, 2002.

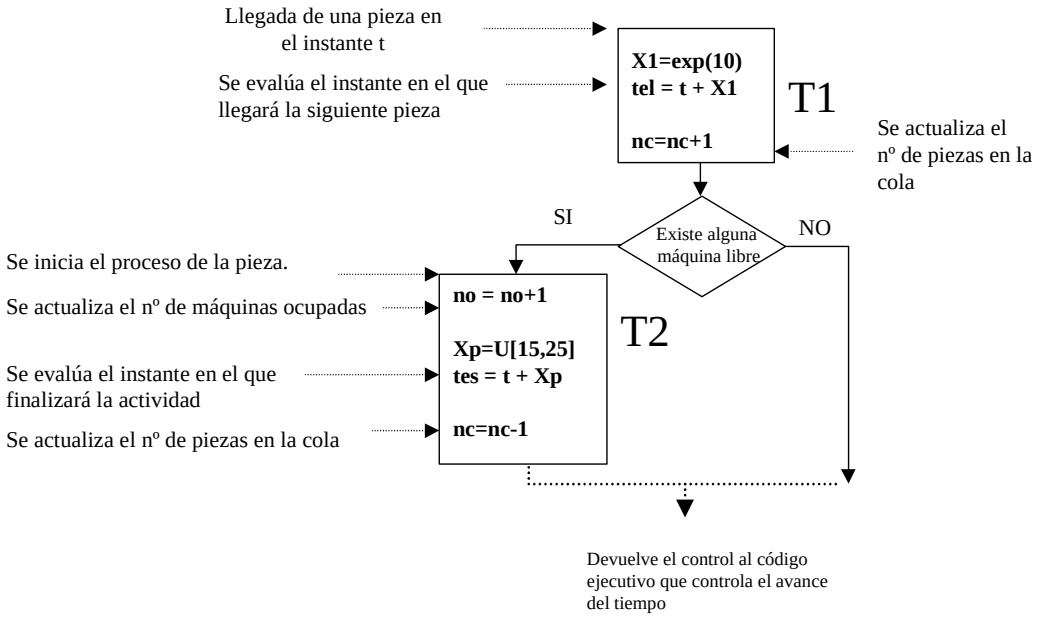


Figura 4.12 Ejecución del evento de llegadas (T_1)

Si las dos máquinas están ocupadas, el evento condicionado de inicio de proceso asociado a la transición T_2 no se puede ejecutar. Por el contrario, si hay una o más maquinas libres, la pieza puede pasar directamente a procesarse, el numero de máquinas ocupadas "no" se incrementa en 1 y se planifica el evento no condicionado, asociado a la transición T_3, de finalización del proceso.

En la figura 4.13 se ha representado el organigrama del código asociado al evento de salida (fin de proceso) en el instante t (transición T_3 de la RdP). Si todavía hay piezas en la cola, el tamaño de la cola se reduce en una unidad y se inicia el proceso de la pieza extraída de la cola (evento condicionado asociado a la transición T_2). También se planifica el siguiente evento no condicionado asociado al fin de proceso (T_3). Si no hay piezas en la cola, el número de máquinas trabajando se decrementa en una unidad.

En la figura 4.14 se ha representado la evolución de la lista de eventos (FEL), así como la evolución de las variables de estado (no, nc) y del reloj del simulador durante los primeros 4 eventos, suponiento que los tres primeros valores generados por la fdp exp(10) han sido {5,4,7}, y que los dos primeros valores generados por la U[15, 25] han sido {17, 20}.

Así pues, como puede observarse, inicialmente (reloj=0) hay programado (lista de eventos) un evento de llegada, la ejecución del cual genera dos nuevos eventos incondicionales: un evento de llegada en el instante 5 y un evento de finalización de actividad en el instante 17, y actualiza la variable de estado "no" al valor 1.

© Los autores, 2002; © Edicions UPC, 2002.

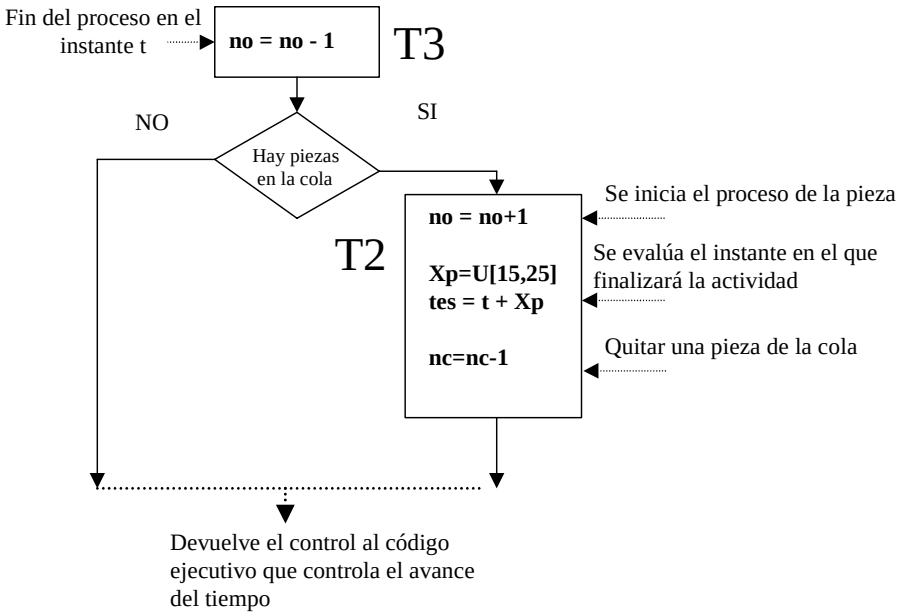


Figura 4.13 Ejecución del evento de salidas (T_3)

Una vez se ha finalizado la ejecución de la rutina de tratamiento del evento de llegadas T_1, el código ejecutivo pasa el control a la rutina de tratamiento del tiempo, la cual indica que el siguiente evento vuelve a ser un proceso de llegada (evento T_1) que se activará en el instante 5, por lo cual la propia rutina actualiza la variable reloj al valor 5. De modo análogo, el código ejecutivo vuelve a pasar el control a la rutina de tratamiento del evento T_1, la cual vuelve a generar dos nuevos eventos incondicionales: un evento de llegada en el instante 9 (Xp=4, tel=t+4) y un evento de finalización de actividad en el instante 25 (Xp=20, tes=t+20), y actualiza la variable de estado "no" al valor 2.

De nuevo, el código ejecutivo pasa el control a la rutina de tratamiento del tiempo, la cual indica que el siguiente evento vuelve a ser un proceso de llegada (evento T_1) que se activará en el instante 9, por lo cual la propia rutina actualiza la variable reloj al valor 9. De modo análogo, el código ejecutivo vuelve a pasar el control a la rutina de tratamiento del evento T_1, la cual genera tan sólo un nuevo evento incondicional, un evento de llegada en el instante 16 (Xp=7, tel=t+7), y actualiza la variable de estado "nc" al valor 1, ya que las 2 máquinas se encuentran ocupadas y la nueva pieza se almacena en la cola de espera.

Como conclusión puede decirse que la estrategia de programación de eventos facilita un buen control de todas las operaciones a realizar, así como facilita la rutina de tratamiento de tiempo, puesto que tan sólo es necesario acceder a la FEL para determinar el siguiente evento que debe ejecutarse. Por otro lado, presenta el inconveniente que la codificación de las rutinas de eventos incondicionales suele

© Los autores, 2002; © Edicions UPC, 2002.

aumentar su complejidad y su dificultad de reutilización[5] en función del número de eventos condicionales asociados a cada evento incondicional.

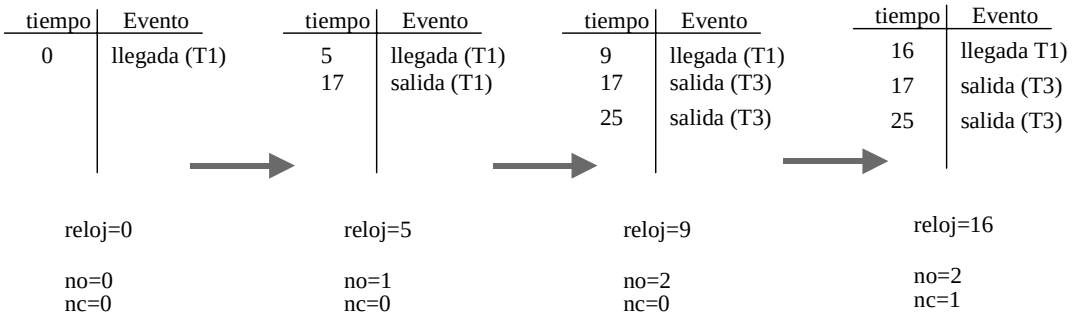


Figura 4.14 Evolución de las variables de estado

4.4.2 Interacción de procesos (*Process Interaction*)

La interacción de procesos se basa en tomar el punto de vista de la entidad que fluye por el proceso. Esta estrategia se centra en los procesos que sufre la entidad a medida que fluye por el sistema. Con esta estrategia se diferencian claramente las entidades temporales (por ejemplo, piezas o clientes) de las entidades permanentes, también denominadas recursos (por ejemplo, máquinas o vehículos de transporte).

En la Interacción de Procesos el analista define el modelo de simulación desde el punto de vista de las entidades temporales, su ciclo de vida a medida que fluyen por el sistema demandando múltiples recursos de capacidad limitada. Las entidades temporales se esperan en las colas si estos recursos no están disponibles. El código ejecutivo mantiene un registro de cada una de las entidades temporales indicando en que estado está dentro del proceso, y cuándo sale de este estado, si es que se conoce. En esta estrategia, se entiende por proceso todo el ciclo de vida de la entidad. Más concretamente, un proceso es una secuencia de eventos, actividades y retardos, incluyendo demanda de recursos, que definen el ciclo completo de vida de una entidad temporal a medida que fluye por el modelo.

La popularidad de esta estrategia reside en que permite al modelador describir el modelo de simulación con bloques de alto nivel en los cuales la interacción entre procesos es gestionada automáticamente por la herramienta de simulación. Internamente, estas herramientas manejan los eventos no condicionados en la LEF, tal como se realizaba en la programación de eventos. Las entidades temporales son almacenadas en listas cuando sufren retardos, finalizan o inician la ejecución en función de las características del flujo.

En la interacción de procesos los modelos son más fáciles de entender a costa de una estructura interna del simulador más complicada. En esta estrategia también se emplean la mayoría de los mecanismos descritos para la programación de eventos en el apartado anterior [KREU86]. Se emplea también una LEF pero, en lugar de referenciar cada evento no condicionado a su módulo de código

[5] Debe recordarse que los modelos de simulación suelen modificarse con mucha frecuencia para la realización de distintos experimentos bajo diferentes hipótesis de operación.

© Los autores, 2002; © Edicions UPC, 2002.

(rutina) correspondiente, los eventos apuntan al proceso. Dado que el proceso tiene memoria del estado de la entidad, el evento no condicionado provoca el cambio de estado de la entidad temporal y un avance en el proceso (ciclo de vida). Como consecuencia de un evento no condicionado sobre una entidad temporal, se pueden activar eventos condicionados sobre otras entidades temporales. Por ejemplo, cuando una entidad temporal finaliza el mecanizado en una máquina, la primera entidad temporal que está en la cola de la máquina puede iniciar su proceso de mecanizado. No confundir el proceso de mecanizado que efectúa la máquina con el concepto de proceso o ciclo de vida de la estrategia de interacción de procesos.

En esta estrategia los modelos se describen como diagramas de bloques y las entidades temporales fluyen por estos bloques a medida que avanzan en su ciclo de vida. Una entidad temporal se puede encontrar en diferentes estados a lo largo de su ciclo de vida [BARC96, SCHR00]:

- *Activa*: es la entidad temporal que en un determinado instante de tiempo se mueve a través de los bloques del modelo. Es decir, avanza en su ciclo de vida.
- *Demorada*: es una entidad que se encuentra demorada en algún bloque pero tiene un tiempo previsto para su activación. La activación se producirá por un evento no condicionado.
- *Detenida*: entidad temporal retenida en algún bloque (por ejemplo, una cola) pero sin tiempo previsto de activación. Su activación depende de un evento condicionado.

No se incluye en este apartado el algoritmo detallado pero el siguiente ejemplo ilustra su funcionamiento.

EJEMPLO 4.5 Ejemplo de la interacción de procesos

La figura 4.15 muestra el modelo de simulación del sistema descrito anteriormente. Se observa que los modelos de simulación desarrollados con esta estrategia tienen una estructura muy parecida a los modelos de flujos descritos en el capítulo 2.

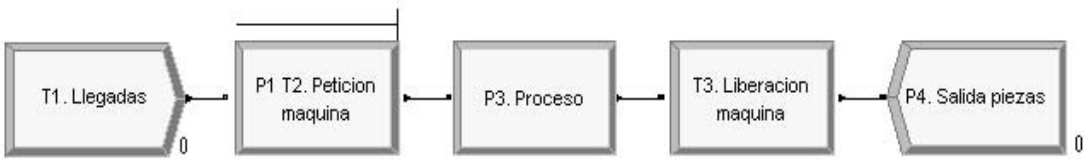


Figura 4.15 Interacción de procesos

La figura 4.16 se muestra el ciclo de vida de las cuatro piezas que llegan primero al sistema y la interacción entre ellas (interacción de procesos). Al llegar la primera pieza (transición T_1 de la RdP) no hay ninguna máquina ocupada y, por tanto, la petición de máquina es aceptada directamente y se inicia el proceso (actividad ligada en el lugar P_3). Al finalizar el proceso se libera la máquina (transición T_3) y la pieza sale del sistema. El ciclo de vida de la tercera pieza es diferente dado que al llegar (T_1) se encuentra con que la petición de máquina no puede ser servida dado que todas están ocupadas. Por tanto, la pieza tiene que permanecer en la cola de espera hasta que una de las máquinas quede libre.

© Los autores, 2002; © Edicions UPC, 2002.

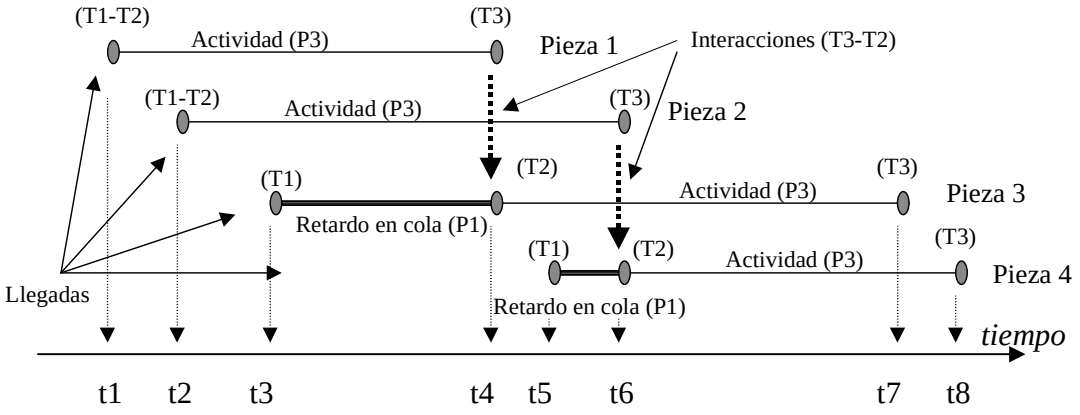


Figura 4.16 Interacción entre procesos

La tabla 4.7 muestra el flujo de entidades temporales por el diagrama de bloques de la figura 4.15 para cada instante de reloj (evento no condicionado).

Tiempo de reloj	Entidad temporal activa	Bloques por los que se mueve	Estado final
t1	Pieza 1	T1 – P1 T2 – P3	Demorada en P3
t2	Pieza 2	T1 – P1 T2 – P3	Demorada en P3
t3	Pieza 3	T1 – P1 T2	Detenida en P1 T2
t4	Pieza 1	P3 – T3 – P4	Fin ciclo de vida
t4	Pieza 3	P1 T2 – P3	Demorada en P3
t5	Pieza 4	T1 – P1 T2	Detenida en P1 T2
t6	Pieza 2	P3 – T3 – P4	Fin ciclo de vida
t6	Pieza 4	P1 T2 – P3	Demorada en P3
t7	Pieza 3	P3 – T3 – P4	Fin ciclo de vida
t8	Pieza 4	P3 – T3 – P4	Fin ciclo de vida

Tabla 4.7 Ciclo de vida de las entidades temporales

También muestra el estado final en que queda la entidad temporal cuando no puede seguir avanzando más. Por ejemplo:

- En el instante t1 se contempla la llegada de la pieza 1 al bloque T_1. Dado que no hay ninguna máquina ocupada, la pieza 1 puede seguir avanzando hasta el bloque P_3 en el que la pieza queda demorada hasta el tiempo previsto de fin de mecanizado, que se encuentra a partir de la distribución de probabilidad del tiempo de mecanizado. En el instante t4 se activa otra vez la pieza en el bloque P_3 y avanza por el P_3 hasta el P_4, donde sale del modelo (finaliza el ciclo de vida de la pieza).
- En el instante t3 llega la pieza 3 al bloque T_1 y avanza hasta el bloque P_1 T_2 para solicitar el recurso máquina. Dado que todas las máquinas están trabajando, la pieza queda detenida hasta

© Los autores, 2002; © Edicions UPC, 2002.

que una máquina quede libre (evento condicional). En el instante t4 queda una máquina libre, por lo que la pieza puede avanzar desde el bloque P_1 T_2 hasta el bloque P_3, donde queda demorada. En el instante t7 se activa otra vez la pieza en el bloque P_3 y avanza por el P_3 hasta el P_4, donde sale del modelo (finaliza el ciclo de vida de la pieza).

Obsérvese que el reloj de la simulación no se incrementa cuando la entidad temporal se desplaza entre bloques y sólo avanza cuando las entidades quedan demoradas o detenidas. En definitiva, el funcionamiento del algoritmo de control de procesos, es similar a una persona lleve de la mano a su hijo. El código ejecutivo "lleva de la mano" a una entidad hasta que ésta queda demorada en algún proceso. Mientras tanto, el código ejecutivo va añadiendo a la LEF todos los eventos que generaba esta entidad mientras es "llevada de la mano". Es por este motivo que en el ejemplo 4.1 el simulador entraba en un bucle infinito, ya que la entidad nunca quedaba demorada en ningún proceso. Todas las acciones que generaba esta entidad quedaban registradas en la LEF, pero nunca llegaban a ejecutarse.

4.4.3 Exploración de actividades (*Activity Scanning*)

En la interacción de procesos los modelos son generalmente programados desde el punto de vista de las entidades que se mueven a través del sistema. Estas entidades generalmente llegan, tienen algún tipo de transformación o proceso para los cuales precisan a menudo capturar y liberar recursos, y a continuación salen. Por tanto, una característica importante de estos modelos es la clasificación explícita entre las entidades temporales (entidades que se mueven por el modelo) y las entidades que sirven (que denominamos entidades permanentes o recursos), que son generalmente escasas. En la exploración de actividades, los modelos son desarrollados desde el punto de vista del conjunto de actividades que hay que ejecutar y el enfoque radica en la identificación de la naturaleza de las actividades, del conjunto de entidades (ya sean entidades temporales o permanentes) y del conjunto de condiciones que permiten que suceda la actividad [IOAN99]. Desde el punto de vista del modelado, el formalismo de las redes de Petri es muy parecido al de la exploración de actividades. En la red de Petri tampoco hay distinción explícita entre las entidades que fluyen por el sistema y los recursos necesarios. En la red de Petri tanto las entidades temporales como los recursos se manejan como entidades (marcas) que fluyen por el sistema (observar, por ejemplo, el paralelismo entre los modelos basados en la exploración de actividades desarrollados por [CARR92] y sus redes de Petri equivalentes). La forma de pensar y modelar con esta estrategia, y también con las redes de Petri, es de elevado interés en aquellos sistemas en los que no se pueden separar de forma estricta las entidades temporales y los recursos. Por ejemplo, en una fábrica las máquinas son generalmente recursos restrictivos pero, en caso de avería, puede ser necesario manejarlas como entidades temporales: máquinas que están en cola para su reparación.

En la programación de eventos y la interacción de procesos el avance del tiempo es variable. El reloj de la simulación siempre avanza hasta el próximo evento no condicionado de la LEF. La exploración de actividades emplea un incremento de tiempo fijo y en cada instante de tiempo explora todas las actividades para decidir si alguna de las actividades puede empezar. Con este método, el modelador se concentra en el conjunto de condiciones que permiten iniciar una actividad. Las condiciones para todas las actividades son evaluadas en cada avance del tiempo para decidir cuáles pueden empezar y cuáles no. Una de las deficiencias principales de esta estrategia es la necesidad de tener un incremento de tiempo reducido dado que, en caso contrario, eventos que están separados en el tiempo pueden aparecer simultáneamente en el simulador. Por tanto, el incremento de tiempo tiene que ser lo

© Los autores, 2002; © Edicions UPC, 2002.

suficientemente pequeño para que dos eventos separados en el tiempo también aparezcan separados en la simulación [POOC00]. Esta reducción forzosa del incremento de tiempo incrementa a menudo, y de forma significativa, el coste en tiempo de la simulación. Por todo ello, esta estrategia ha evolucionado hacia la denominada estrategia de *tres fases*, que combina el salto variable en el tiempo propio de la programación de eventos con la exploración de actividades [BANK95, CARR92]:

1. Extraer de la LEF el evento más inminente, avanzar el reloj de la simulación hasta este instante. Extraer el resto de eventos de la LEF que tienen el mismo tiempo.
2. Ejecutar todos los eventos extraídos, en el paso anterior, de la LEF. Recordar que los eventos de la LEF son eventos no condicionados. En esta fase podemos liberar recursos y/o cambiar el estado del sistema.
3. Explorar todas las actividades asociadas a eventos condicionados para detectar y ejecutar aquellas actividades que cumplen todas las condiciones. La exploración se repite cíclicamente hasta que no se encuentra activada ninguna actividad adicional.

En resumen, esta estrategia basada en tres fases separa o desacopla claramente las actividades ligadas a eventos no condicionados de las ligadas a eventos condicionados, de forma que las primeras son todas ejecutadas en primer lugar y, a continuación, se efectúa un barrido para ejecutar las actividades asociadas a eventos condicionados que cumplen las condiciones. Esta estrategia evita la evaluación continuada de los eventos condicionados, que se puede disparar como consecuencia de un evento no condicionado. Obsérvese, que la estrategia basada en la exploración de actividades tiene interés para resolver problemas complejos en los que recursos de diferentes tipos y diferentes combinaciones de recursos son necesarios para efectuar las diferentes tareas.

Algoritmo

En la tabla siguiente se muestra el pseudocódigo ejecutivo para un algoritmo de exploración de actividades basado en las tres fases descritas anteriormente. Las inicializaciones y el mecanismo de avance de tiempo son parecidos a la estrategia de programación de eventos. La diferencia principal es que para cada instante de tiempo, al cual se ha llegado después de avanzar el tiempo al tiempo de evento no condicionado más inminente, se ejecutan primero todos los eventos no condicionados activos y, a continuación, se barren repetidamente todas las actividades dependientes de eventos condicionados para evaluar si se pueden iniciar.

```
Inicio ejecutivo
       Inicio módulo de inicialización
              Inicializar tiempo_inicio, tiempo_fin;
              Inicializar el reloj de la simulación al tiempo_inicio;
              Inicializar el estado inicial y los contadores estadísticos;
              Almacenar los eventos no condicionales iniciales en la LEF;
              Extraer de la LEF el evento más inminente y todos lo que tienen el
                     mismo tiempo (fase 1);
              Avanzar el reloj hasta el tiempo del evento extraido;
       Fin inicialización;

       Inicio bucle de simulación
              Mientras el reloj no ha alcanzado el tiempo_fin
```

© Los autores, 2002; © Edicions UPC, 2002.

```
                    Ejecutar el código asociado a los eventos no condicionados
                           extraidos (fase 2)
                    Barrer repetidamente todos los eventos condicionados para
                           detectar y ejecutar las actividades que cumplen
                           todas las condiciones (fase 3)
                    Extraer de la LEF el evento más inminente y
                           todos lo que tienen el mismo tiempo (fase 1);
                    Avanzar el reloj hasta el tiempo del evento extraido;
             Fin mientras;
       Fin bucle de simulación;
 Generar un informe con las estadísticas de interés;
Fin ejecutivo;
```

Tabla 4.8 Código ejecutivo de la exploración de actividades

El código asociado a cada uno de los eventos no condicionados es más simple que en la estrategia de programación de eventos dado que no es necesario preguntarse si, como consecuencia de éste, se cumplen todas las condiciones para activar eventos condicionados.

```
Inicio código de evento no condicionado
       Actualizar el estado del modelo;
       Actualizar los contadores estadísticos;
       Generar futuros eventos no condicionados e incluirlos en la LEF;
Fin código de evento no condicionado;
```

Tabla 4.9 Código de evento no condicionado de la Exploración de Actividades

Así pues, con la estrategia de exploración de actividades, el modelo de simulación permite programar tanto eventos incondicionales como eventos condicionales, los cuales tienen que llevar especificadas todas las condiciones que permiten su aparición.

```
Inicio código de evento condicionado
       Si se cumplen todas las condiciones necesarias para iniciar la actividad;
              Actualizar el estado del modelo;
              Actualizar los contadores estadísticos;
              Generar futuros eventos no condicionados e incluirlos en la LEF;
       Fin si;
Fin código de evento condicionado;
```

Tabla 4.10 Código de evento condicionado de la exploración de actividades

En la figura 4.17 se muestra el organigrama de los tres eventos de la RdP de la figura 4.3 del ejemplo 4.2 mostrado anteriormente. Como se puede observar, para cada transición de la RdP hay una rutina de tratamiento de evento asociada. Las transiciones con más de una arco de entrada corresponden a las transiciones asociadas a eventos condicionados y, por tanto, su código asociado tendrá que ser explorado para evaluar que todas las condiciones de entrada se cumplen.

© Los autores, 2002; © Edicions UPC, 2002.

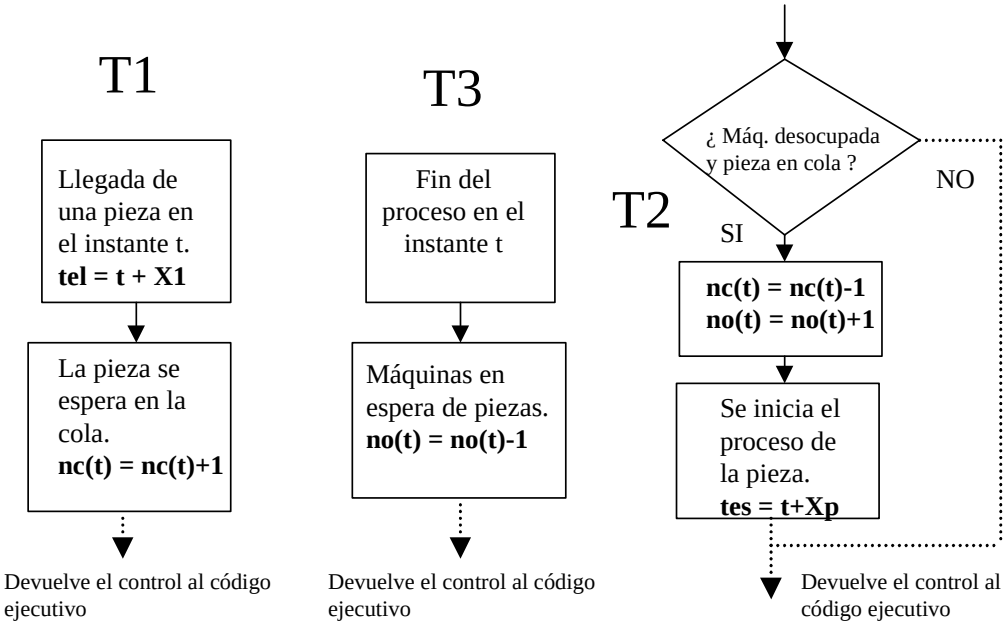


Figura 4.17 Código asociado a los eventos no condicionados y condicionados.

EJEMPLO 4.6 Ejemplo de la exploración de actividades

Un problema inherente a la interacción de procesos está originado por la reserva de recursos cuando dos o más recursos independientes son necesarios para efectuar una operación [CARR92]. En la figura 4.18 se muestra una estación de trabajo con una máquina.

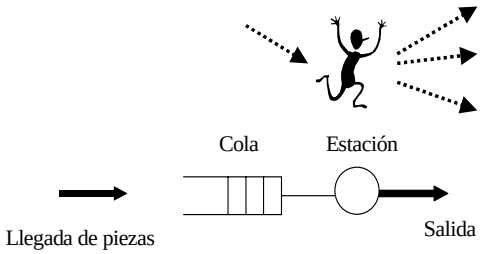


Figura 4.18 Estación de trabajo

Dado que las operaciones contempladas para ser realizadas son muy delicadas, es necesaria la supervisión de un trabajador. Por tanto, para que la operación se pueda efectuar es necesario que el recurso máquina esté libre y que el recurso trabajador también se encuentre libre. Esta máquina realiza también otras operaciones en las que no es necesario la supervisión del trabajador. El trabajador, aparte de supervisar esta célula cuando las operaciones que se deben realizar son delicadas, supervisa otras instalaciones de la factoría.

© Los autores, 2002; © Edicions UPC, 2002.

La estrategia de exploración a de actividades nos asegura que la actividad sólo se llevará a cabo si todas las condiciones se cumplen y sólo en este instante se asignarán los recursos necesarios a la actividad. Por tanto, en este caso particular, sólo se inicia la actividad cuando están libres el operador y la máquina. En el instante de inicio se reservan ambos recursos. Esta estrategia, y para este caso particular, es equivalente a la RdP de la figura 4.19a, la transición T_1 se dispara si tanto la máquina como el trabajador están libres y hay piezas en la cola (lugar P1).

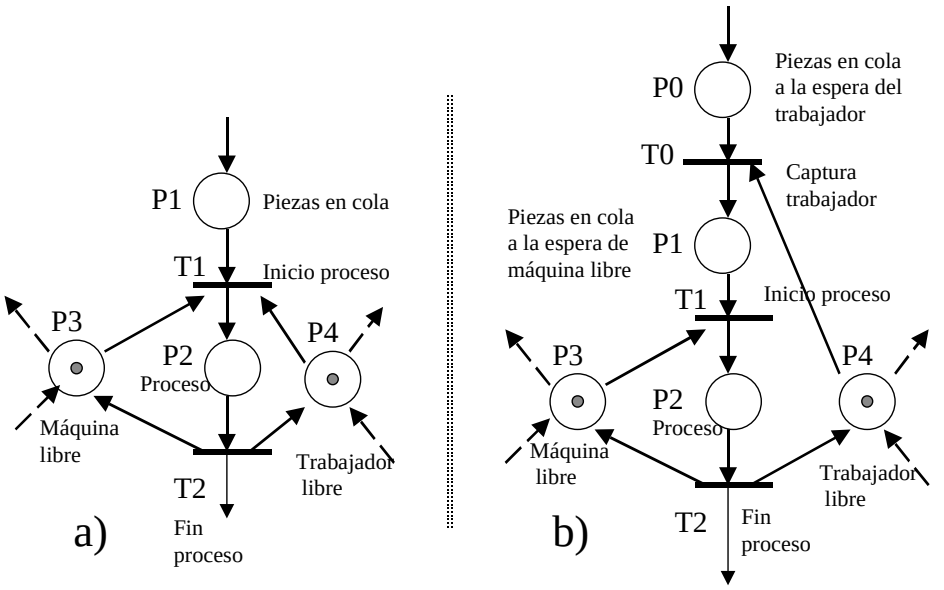


Figura 4.19 Estación de trabajo

En la interacción de procesos, el comportamiento del modelo puede depender de la estructura del código. Una programación poco cuidadosa nos podría dar como resultado un modelo incorrecto equivalente a la RdP de la figura 4.19b. Se observa que primero se captura el trabajador sin saber si la máquina está libre o ocupada. Nos podríamos encontrar que el trabajador está un tiempo elevado sin hacer nada a la espera de que la máquina quede libre, dado que la máquina está efectuando operaciones con una pieza que no necesita supervisión. Adicionalmente, y durante este tiempo, otra instalación puede precisar del recurso trabajador para poder avanzar.

EJEMPLO 4.7 Sistema de producción

Con el objetivo de ilustrar la facilidad de reutilización de código de la estrategia de exploración de actividades, así como ilustrar la codificación de un sistema de producción formalizado en RdPC, considérese el sistema descrito en la figura 4.20 el cual consiste en un proceso de llegadas de dos tipos diferentes de piezas a y b, sobre las que se realiza una primera operación en la máquina 1, y una segunda operación en la máquina 2 si se trata de una pieza de tipo a, o en la máquina 3 si se trata de una pieza de tipo b. El robot de la figura realiza el transporte de las piezas a y de las piezas b a la máquina 2 y máquina 3 respectivamente si y sólo si hay espacio disponible en el almacén asignado a cada una de las máquinas. Dichos almacenes se encuentran inicialmente vacíos y tienen una capacidad máxima de 5 piezas.

© Los autores, 2002; © Edicions UPC, 2002.

En la tabla 4.11 se resume el significado de cada una de las transiciones y de los nodos lugar

Nodo	Comentarios
T_1	Llegada de una pieza de tipo a
T_2	Llegada de una pieza de tipo b
T_3	Inicio de la operación de procesamiento en la máquina M1
T_4	Transporte de la pieza de tipo a de la máquina M1 al almacén de la máquina M2
T_5	Transporte de la pieza de tipo b de la máquina M1 al almacén de la máquina M3
T_6	Fin de la operación de procesamiento en la máquina M1
P_1	Almacén asociado a la máquina M1
P_2	Máquinas libres. Inicialmente se encuentran libres las 3 máquinas.
P_3	Pieza procesándose en la máquina M1.
P_4	Espacios libres en los almacenes asociados a las máquinas M2 y M3
P_5	Piezas en el almacén asociado a la máquina M2
P_6	Piezas en el almacén asociado a la máquina M3
P_7	Pieza procesada en la máquina M1 esperando a ser retirada por el robot.

Tabla 4.11 Descripción de los nodos de la RdPC del sistema de producción

Con el objetivo de ilustrar de modo simplificado los aspectos de codificación de las rutinas de tratamiento de eventos, se ha considerado que el robot está dedicado al transporte de las piezas desde M1 a M2 y M3, y el tiempo de transporte es considerablemente inferior al tiempo de operación en la máquina M1. Esta característica permite suponer que el robot siempre se encuentra libre, por lo que no ha sido su estado precondición de los eventos T_4 y T_5. Por el mismo objetivo de simplificación, tampoco se han considerado las operaciones de las máquinas M2 y M3, se puede suponer que pertenecen a otra subred.

En la figura 4.21se describen las rutinas de tratamiento de los eventos T_1 a T_6 con la estrategia de exploración de actividades. En la tabla 4.12 se comenta el significado de las variables de estado utilizadas en la descripción del sistema de producción.

Variable	Comentarios
na	Número de piezas de tipo a almacenadas en la cola asociada a la máquina M1
nb	Número de piezas de tipo b almacenadas en la cola asociada a la máquina M1
M1	Estado de la máquina M1: libre, procesando_a, procesando_b, ocupada_a, ocupada_b
M2_nc	Número de piezas en el almacén asociado a la máquina M2
M3_nc	Número de piezas en el almacén asociado a la Máquina M3

Tabla 4.12 Variables de estado

© Los autores, 2002; © Edicions UPC, 2002.

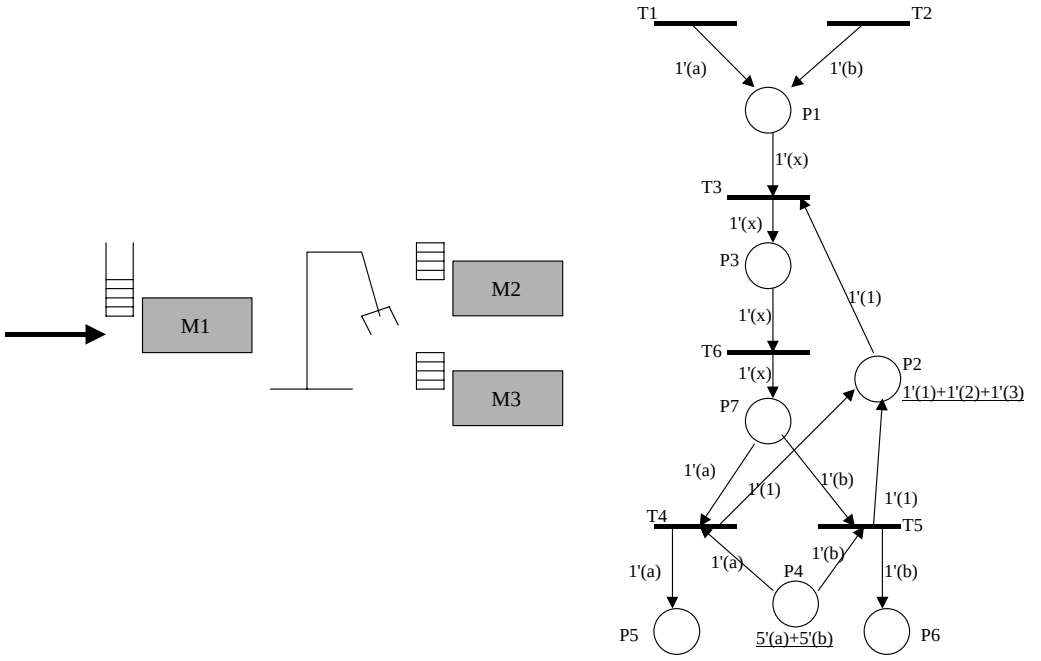


Figura 4.20 RdPC de un sistema de producción

Nótese que las variables descritas en la tabla 4.12 especifican completamente el estado del sistema de producción. En la tabla 4.13 se especifican las condiciones asociadas a los eventos condicionales T_3, T_4 y T_5.

Condición	Evento
(M1 == libre) & ((na>0) || (nb>0))	T_3
(M2_na <5) & (M1=procesando_a)	T_4
(M2_nb <5) & (M1=procesando_b)	T_5

Tabla 4.13 Condiciones de los eventos T_3, T_4 y T_5

Así pues, para que pueda activarse el evento T_3 (inicio de proceso en la máquina M1), debe haber como mínimo una pieza en la cola de espera, y la máquina M1 debe encontrarse libre. Las precondiciones del evento de transporte de M1 a M2 (evento T4) requieren que no hayan más de 5 piezas en el almacén asociado a la máquina M2, y que la máquina M1 haya finalizado el procesamiento de una pieza de tipo a. De modo análogo, las precondiciones del evento de transporte de M1 a M23 (evento T_5) requieren que no haya más de 5 piezas en el almacén asociado a la máquina M3, y que la máquina M1 haya finalizado el procesamiento de una pieza de tipo b.

© Los autores, 2002; © Edicions UPC, 2002.

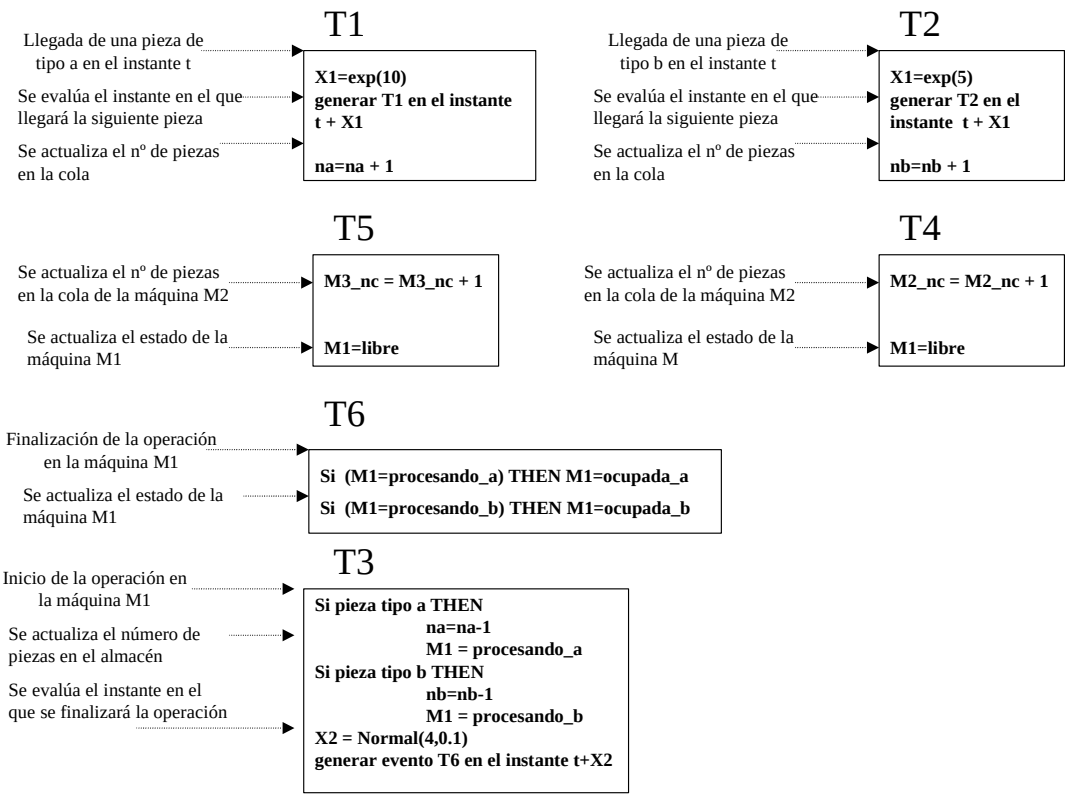


Figura 4.21 Rutinas de tratamiento de los eventos T1... T6, siguiendo la estrategia de exploración de actividades

En la figura 4.22 se ha representado la evolución de la lista de eventos (FEL), así como la evolución de las variables de estado (na,nb,M1,M2_nc,M3_nc) y del reloj del simulador durante los primeros 4 eventos, suponiento que los dos primeros valores generados por la fdp exp(10) han sido {5,9}, el primer valor generado por la fdp exp(5) ha sido {10}, y que el primer valor generado por la N(8,0.1) ha sido {8}.

Así pues, como puede observarse, inicialmente (reloj=0) hay programado (lista de eventos) un evento de llegada de pieza de tipo a y un evento de llegada de pieza de tipo b. La ejecución del código asociado al evento T_1 en el instante reloj=0 genera un nuevo evento incondicional: un evento de llegada de piezas tipo a en el instante 5. Como efecto del evento T_1, el valor de la variable de estado se actualiza al valor 1. Una vez finalizado el tratamiento de la rutina de evento T_1 en el instante 0, el código ejecutivo, antes de pasar el control a la rutina de tratamiento de tiempo, evalúa si hay algun evento condicional que pueda activarse en el mismo instante, y observa que el evento T_3 puede ser disparado, por lo que actualiza los valores de las variables de estado na y M1, y genera un nuevo evento incondicional T_6 que se añade a la lista FEL.

© Los autores, 2002; © Edicions UPC, 2002.

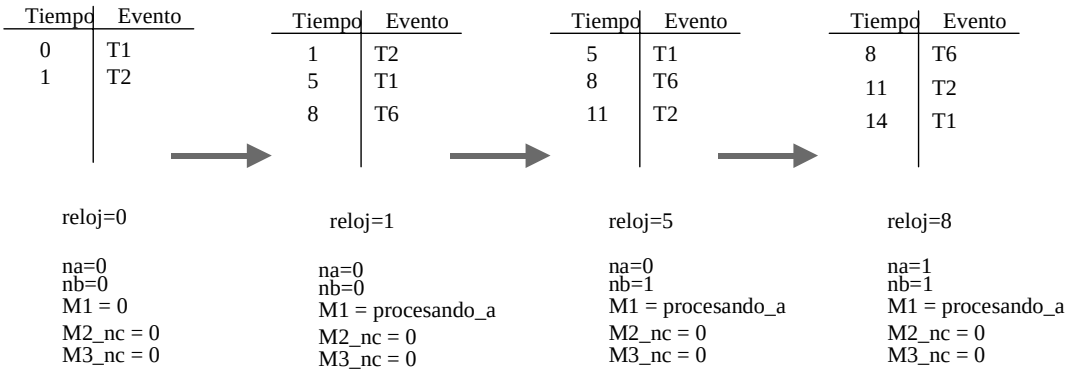


Figura 4.22 Evolución de las variables de estado

Una vez se ha finalizado la ejecución de la rutina de tratamiento del evento de llegadas T_1, el código ejecutivo pasa el control a la rutina de tratamiento del tiempo, la cual indica que el siguiente evento es un proceso de llegada de piezas tipo b (evento T_2) que se activará en el instante 1, por lo cual la propia rutina actualiza la variable reloj al valor 1. De modo análogo, el código ejecutivo vuelve a pasar el control a la rutina de tratamiento del evento T_2, la cual genera 1 nuevo evento incondicional: un evento de llegada de piezas de tipo b en el instante 11 (t+10), y actualiza la variable de estado nb al valor 1.

De nuevo, el código ejecutivo pasa el control a la rutina de tratamiento del tiempo, la cual indica que el siguiente evento vuelve a ser un proceso de llegada (evento T_1) que se activará en el instante 5, por lo cual la propia rutina actualiza la variable reloj al valor 5. De modo análogo, el código ejecutivo vuelve a pasar el control a la rutina de tratamiento del evento T_1, la cual genera tan solo 1 nuevo evento incondicional: un evento de llegada de piezas de tipo a en el instante 14 (t+9) y actualiza la variable de estado na al valor 1.

4.5 *Software* de simulación

Aunque cualquier lenguaje de programación podría ser utilizado para simular el comportamiento de un sistema orientado a eventos discretos, existen un conjunto de características inherentes tanto a los proyectos de simulación, como a la evolución que presentan los sistemas que se desean simular que aconsejan el uso de herramientas propias del campo de la simulación.

Las herramientas de simulación ofrecen los módulos necesarias para programar de forma ágil un modelo, pueden destacarse: generación de números aleatorios que responden a ciertas funciones de distribución particulares, gestión automatizada del tiempo de simulación, gestión automatizada de las rutinas de tratamiento de los eventos que determinan el comportamiento del sistema, algoritmos de análisis de los resultados, y generación de informes entre otros. Todas estas características repercuten en una reducción considerable del tiempo de programación del modelo. Adicionalmente, el mantenimiento de un modelo de simulación se ve enormemente simplificado cuando éste ha sido programado con una herramienta simulación.

© Los autores, 2002; © Edicions UPC, 2002.

4.5.1 Criterios para la evaluación de las herramientas informáticas

Para seleccionar los criterios de evaluación se toma como referencia la norma ISO 9126. Los criterios que propone son (figura 4.23), funcionalidad, fiabilidad, facilidad, eficiencia, mantenibilidad y portabilidad. Otros dos criterios adicionales de interés son la calidad de la empresa que desarrolla y vende la herramienta de simulación y los costes de la herramienta y otras opciones [PIER02].

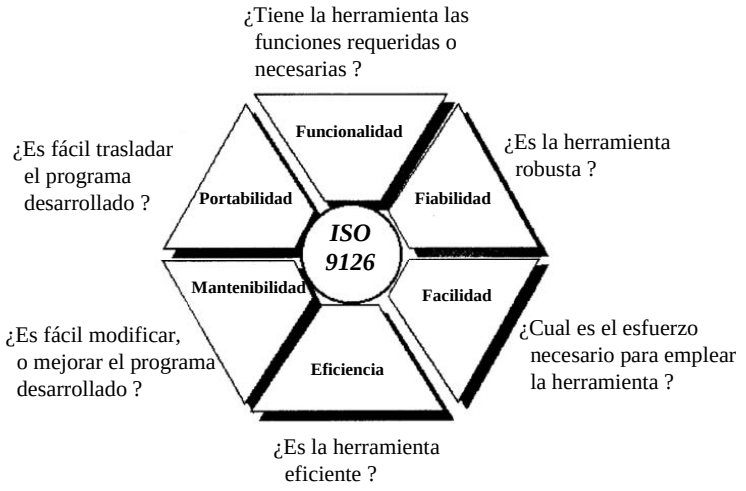


Figura 4.23 Criterios de evaluación de las herramientas informáticas

Hay que recordar, sin embargo, que aunque la correcta selección de la herramienta de simulación es importante, no garantiza el éxito del proyecto de simulación. *La mejor garantía para el éxito reside en la experiencia, calidad y formación del personal que trabaja en el proyecto*.

Funcionalidad

Aunque los simuladores actuales suelen ofrecer las herramientas necesarias para la formalización de cualquier sistema orientado a eventos discretos, el tiempo invertido en el desarrollo del modelo, así como en su mantenimiento, depende considerablemente de las funciones predefinidas en el entorno, las cuales, en muchos casos, han sido desarrolladas para campos de aplicación particulares. Es fácil darse cuenta que las funciones disponibles en un simulador de sistemas financieros serán radicalmente diferentes a las ofrecidas en un simulador de sistemas de tráfico. Se recomienda la evaluación de simuladores orientados a problemas del campo de aplicación al que pertenece el sistema que se desea estudiar, teniendo en cuenta el nivel de detalle que se desea alcanzar.

La funcionalidad de una herramienta de simulación puede ser evaluada atendiendo a los aspectos siguientes:

- *Construcción del modelo*: la *descomposición jerárquica* del modelo permite estructurar mejor el modelo. La posibilidad de *llamar programas o módulos fuera del modelo* escritos en lenguajes convencionales de programación permite modelar o incluir aspectos no contemplados por el simulador. Por ejemplo, es a menudo necesario poder conectar el simulador a herramientas externas de optimización y planificación. *La metodología de modelado y/o la estrategia de*

© Los autores, 2002; © Edicions UPC, 2002.

simulación puede facilitar la tarea de modelado y construcción posterior del modelo. La capacidad de soportar *submodelos continuos* es de interés para analizar modelos combinados. Un entorno de simulación tiene que disponer de *funciones potentes y flexibles* para representar de forma elegante la complejidad inherente a los sistemas reales. Estas funciones tienen que permitir modelar los elementos físicos de interés (por ejemplo, servidores, cintas y transportadores), sus restricciones, y las reglas de gestión y planificación. Es generalmente necesario que el entorno de simulación tenga mecanismos de programación equivalentes a los de los lenguajes de programación de alto nivel: if-then-else, while, ... que permitan el control del flujo de la simulación de una manera parecida a la de los lenguajes de programación convencionales.

- *Simulación y animación*: la posibilidad de *animar* la evolución del modelo facilita la depuración de errores y la presentación de resultados. Para facilitar la construcción de la animación es de interés la posibilidad *de importar gráficos CAD*. Es muy importante disponer de *mecanismos de depuración interactiva* para analizar el comportamiento del sistema y depurar sus errores. Es muy interesante la capacidad de ejecutar el modelo (no modificar) *sin necesidad de una licencia de desarrollo*. Esto permite distribuirlos sin necesidad de que el receptor adquiera la licencia de desarrollo.

- *Entradas y salidas*: Para parametrizar el modelo de simulación es en general necesario obtener, a partir de los datos disponibles, las distribuciones de probabilidad que modela el comportamiento de los diferentes elementos. Por ejemplo, frecuencia de llegadas y tiempos de proceso o averías. Esto *requiere un módulo o función que permita obtener la mejor distribución a partir de los datos conocidos*. Si el entorno de simulación no tiene esta función, se pueden adquirir herramientas informáticas que hacen este ajuste. Un buen entorno de simulación tiene que permitir definir y analizar *estadísticas tanto continuas como discretas*. Es conveniente que el entorno permita analizar el comportamiento del modelo en *régimen permanente*. En consecuencia, tiene que ser posible definir un tiempo de calentamiento para eliminar las estadísticas recogidas en el transitorio. Debe ser posible efectuar múltiples simulaciones *empleando diferentes secuencias de números aleatorios.* Dado que la mayoría de los posibles usuarios de los modelos de simulación están familiarizados con el uso de hojas de cálculo, es muy útil que la entrada de parámetros o la salida de resultados se efectúe a través de una *hoja de cálculo*. De esta forma el usuario final no tiene que estar familiarizado con los detalles del entorno de simulación. Cuando una parte significativa de la información del modelo tiene que ser empleada o compartida con otras aplicaciones (por ejemplo, un planificador), puede ser conveniente almacenar la información compartida en una *base de datos*. También es necesaria la lectura o escritura a *archivos.*

Facilidad

La mayoría de vendedores de entornos de simulación hacen énfasis en la potencia pero, en algunos casos, su elevada complejidad dificulta notablemente la obtención de resultados positivos en un tiempo razonable. La tendencia actual va hacia los entornos gráficos tipo Windows, que reducen la cantidad de información que hay que introducir en el teclado. La capacidad de definir gráficamente los elementos del modelo y sus relaciones también contribuyen a que los entornos sean más fáciles de usar. Los aspectos que facilitan que el entorno de simulación sea fácil de emplear son: capacidad de interacción del ratón con el entorno; calidad de la documentación, manual interactivo; menús y

© Los autores, 2002; © Edicions UPC, 2002.

opciones simples; entradas gráficas siempre que sea posible; generación automática de estadísticas; ayuda sensitiva al contexto; opciones de depuración y seguimiento; mecanismos de detección de errores y ejemplos de referencia.

Eficiencia

La velocidad de ejecución es un factor importante en la medida en que uno de los costes más importantes de un proyecto de simulación es el tiempo del analista. La velocidad de simulación esta penalizada muchas veces por el número de entidades y elementos del modelo. Muchos paquetes tienen varias formas de ejecutar un modelo:

- *Con animación*: permite visualizar el comportamiento dinámico del modelo.
- *A marcha rápida*: permite avanzar más deprisa hasta el instante de interés, dado que no actualiza la información gráfica (sin animación). A partir del instante de interés se puede ejecutar el modelo con animación.
- *Proceso en lotes*: ejecuta todo el modelo sin animación. Es el modo que permite la ejecución más veloz.

Fiabilidad

La fiabilidad del entorno de simulación es uno de los aspectos más complicados de evaluar debido a que las limitaciones o errores generalmente no aparecen con los primeros ejemplos de evaluación. En consecuencia, en lugar de evaluar la viabilidad del entorno se cree más idóneo evaluar la capacidad de respuesta de la empresa que desarrolla entornos de simulación ante posibles errores: ¿tiene el vendedor una certificación de calidad del *software*?; ¿cuán robusta es la tecnología empleada en desarrollar el entorno de simulación?; y ¿cuál es el mecanismo de corrección de errores y cómo son las mejoras proporcionadas al usuario final?

Portabilidad

La portabilidad evalúa si el entorno de simulación puede ejecutarse en diferentes sistemas operativos. Algunos aspectos que afectan o están ligados a la portabilidad son:

- Si precisa de placas gráficas o *drivers* especiales.
- La capacidad o no de trabajar en red.
- Si requiere de compiladores o entornos adicionales. Por ejemplo Excel, Access, Visual Basic, C++,…

4.5.2 Entornos de simulación de eventos discretos

La tabla siguiente muestra un directorio de herramientas de simulación con precios por debajo de los 20,000 euros. Esta lista no incluye todas las herramientas de simulación disponibles en el mercado en este rango de precios. Los nombres de los productos y las empresas pueden cambiar debido a alianzas entre empresas, adquisiciones u otras razones. Se recomienda consultar en Internet si se desea efectuar una exploración más detallada.

© Los autores, 2002; © Edicions UPC, 2002.

Producto	*Empresa*	*Dirección WWW*
Arena	Rockwell Software	www.rockwellsoftware.com
Automod	AutoSimulations, Inc.	www.autosim.com
GPSS/H	Wolverine Software Corporation	www.wolverinesoftware.com
LeanSIM	Fib-UPC	www.fib.upc.es/leanSim
Quest	Delmia Corp.	www.delmia.com
MicroSaint	Micro Analysis Design, Inc.	www.madboulder.com
ProModel	Promodel Corporation	www.promodel.com
Simul8	Simul8 Corp.	www.simul8.com
Taylor II/ED	F & H Simulatiions	www.taylor-ed.com
Witness	Lanner Group	www.lanner.com

Tabla 4.14 Principales herramientas de simulación de precio inferior a los 20,000 euros

En http://www.lionhrtpub.com/orms/surveys/Simulation/Simulation.html se puede encontrar un análisis comparativo de las principales herramientas de simulación del mercado.

Una alternativa a las herramientas anteriores es emplear directamente un simulador de redes de Petri. En http://www.daimi.au.dk/PetriNets/ (World of Petri Nets) se encontrará una lista completa de herramientas para la simulación de redes de Petri. No obstante, la casi totalidad de estas herramientas tienen interés estrictamente académico dado que su potencial práctico está muy alejado del de las herramientas comerciales de simulación.

4.6 Ejercicios

1. Repite 2 veces la simulación manual del apartado 1.2 con valores aleatorios diferentes y estima para cada simulación el retardo promedio de la cola, el tamaño promedio de la cola y el grado de utilización de las máquinas.

2. En el ejemplo 2.11 del capítulo 2 se representa la RdPC de un proceso hopitalario

   a) Identifica los recursos y las entidades temporales empleadas.
   b) Identifica las actividades del proceso.
   c) Idenifica todos los eventos y clasifícalos en condicionados, no condicionados, internos y externos.
   d) Identifica los lugares que se pueden asociar a colas.

3. Se ha simulado un proceso con una cola y un solo servidor y se han obtenido los tiempos de entrada al modelo y los tiempos de salida del servidor para las 15 primeras piezas procesadas:

Número depieza	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Entrada al modelo (mn)	0	0.5	2.2	2.4	2.6	2.7	4.1	4.9	5.1	6.1	7	8.8	9.1	9.6	12.2
Salida del modelo (mn)	0.4	1	2.7	4	5.4	5.6	5.7	6.4	6.8	6.8	7.9	11.7	13	14.1	16.1

Tabla 4.15 Evolución del proceso

© Los autores, 2002; © Edicions UPC, 2002.

Determina el tamaño promedio de la cola contabilizando desde el tiempo inicial (minuto 0) hasta la llegada al sistema de la pieza número 10 (minuto 6.1).

4. En la siguiente tabla se muestran los tiempos de llegada y de servicio de 13 clientes que llegan a un sistema con una sola cola y un servidor.

Número de pieza	1	2	3	4	5	6	7	8	9	10	11	12	13
Entrada al modelo (mn)	12	31	63	95	99	154	198	221	304	346	411	455	537
Tiempo de servicio (mn)	40	32	55	48	18	50	47	18	28	54	40	72	12

Tabla 4.16 Evolución del proceso

   a) Determina el tiempo de salida de los 13 clientes, el tamaño promedio de la cola, el tiempo de espera promedio en la cola y el grado de utilización del servidor.
   b) Repite el caso anterior pero suponiendo que hay dos servidores en el sistema.
   c) Repite el caso a) pero supón que, cuando el servidor finaliza el servicio de un cliente, el próximo cliente que entra es el que se ha esperado menos (el último en llegar).

5. El problema de los filósofos se ha presentado en el apartado de ejercicios de los capítulos 2 y 3. Se supone que el tiempo de meditación sigue una distribución uniforme entre 20 y 210 minutos y el de comer una distribución uniforme entre 10 y 30 minutos y que, en el instante inicial, todos los filósofos empiezan a meditar a la vez. Simula manualmente la evolución del proceso durante 10 horas cuando primero cogen el tenedor derecho y después el izquierdo. Si lo crees oportuno, detalla la secuencia de transiciones de la red de Petri que han llevado al proceso desde el estado inicial M0 hasta el estado final después de 10 horas.

6. El puerto de Bilbao tiene un muelle dedicado exclusivamente a la descarga de barcos que transportan contenedores. Este muelle tiene dos grúas para efectuar las operaciones de descarga. A este muelle llegan dos tipos de barcos:

   - Barcos de longitud aproximada de 100 metros. Estos barcos son descargados por una sola grúa. Por tanto, si es necesario, se pueden descargar dos barcos de 100 metros a la vez dado que tenemos dos grúas. El tiempo necesario para la descarga es variable. A partir de las estadísticas efectuadas se ha obtenido que el tiempo en días necesario para la descarga, se puede representar mediante una función de densidad de probabilidad triangular (0.5,1,3) como la representada en la figura:

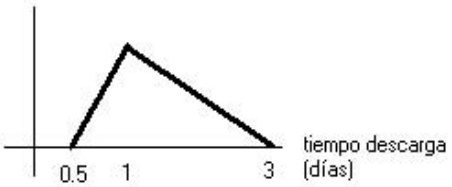


Figura 4.24 Distribución del tiempo de descarga de barcos de 100 m

© Los autores, 2002; © Edicions UPC, 2002.

- Barcos de longitud aproximada de 200 metros. Estos barcos son siempre descargados por las dos grúas disponibles. Estos barcos están muy automatizados, por lo que el tiempo de descarga es de 3 días empleando las dos grúas.

La frecuencia de llegadas de los dos tipos de barcos es:

- Los barcos de 100 metros llegan cada día con una probabilidad de 0.6; cada 2 días con una probabilidad de 0.35 y cada 3 días con una probabilidad de 0.05.
- Los barcos de 200 metros llegan cada 3 días con una probabilidad de 0.3; cada 4 días con una probabilidad de 0.4 y cada 7 días con una probabilidad de 0.3.

Para simplificar el problema se supone que se trabaja con días naturales y con exceso, en consecuencia:

- Si al barco de 100 metros le corresponde, por ejemplo, un tiempo de descarga de 1.7 días se supondrá que ocupa la grúa 2 días completos.
- Si el barco de 200 metros tiene que llegar después de 10.5 días, se supondrá que llegará después de 11 días.

El barco de 200 metros es prioritario, en consecuencia:

- Si llegan el mismo día un barco de 100 y uno de 200, el de 200 tiene prioridad.
- Si un barco de 200 llega al puerto y otro de 100 ya está esperando, el de 200 adelanta al de 100.
- Si llega uno de 200 pero una grúa ya esta ocupada por uno de 100, el barco de 200 se tiene que esperar a que salga el de 100 dado que el de 200 necesita las dos grúas.
- Si llegan uno de 200 y uno de 100 el mismo día pero una grúa ya está ocupada por otro de 100, el de 200 se tiene que esperar forzosamente. El de 100 puede entrar a ocupar la segunda grúa solo si este tiene un tiempo de descarga tal que no penalice más al barco de 200 que esta esperando. Es decir, el barco de 100 que desea atracar tiene que salir (desocupar la grúa) como más tarde el mismo día que el barco que ya está atracado.

Para simular emplea las siguiente secuencias de números aleatorios:

- Llegadas de barcos de 100 metros: 0.21, 0.24, 0.57, 0.17, 0.74, 0.72, 0.38, 0.16, 0.83.
- Tiempos de atraque (uso de grúa de barcos de 100): 0.86, 0.63, 0.52, 0.55, 0.80, 0.76, 0.40, 0.53, 0.37.
- Llegadas de barcos de 200: 0.47, 0.14, 0.38.

a) Calcula teóricamente si el número de grúas es suficiente. Para ello supón que no se conoce el número de grúas. Calcula teóricamente las necesarias y compáralas con las que tenemos.
b) Simula manualmente el proceso hasta que todos los barcos salgan del proceso (9 barcos de 100 y 3 de 200 metros) con las dos grúas supuestas inicialmente.
c) Determina el número de barcos promedio en espera para descargar, el tiempo medio de espera y el tiempo máximo.
d) Si cada hora de espera para el atraque de un barco de 200 metros representa una pérdida para el armador de 600 euros/hora y una perdidad de 300 euros/hora el barco de 100 metros,

© Los autores, 2002; © Edicions UPC, 2002.

calcula las pérdidas totales acumuladas en el periodo de 2 semanas (14 días) desde el inicio de la simulación.

7. La figura siguiente representa el proceso simple compuesto por una estación de trabajo con dos máquinas descrito en el capítulo 2 (ejemplo 2.1), en el capítulo 3 (ejemplo 3.1) y en este capítulo (ejemplo 4.2). Se supone que el tiempo entre llegada sigue una distribución exponencial de media de 4 minutos y el tiempo de proceso sigue una distribución exponencial de 8 minutos de media.

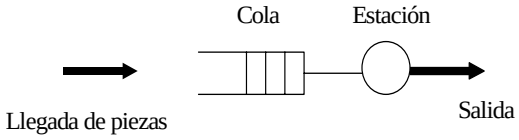


Figura 4.25 Estación de trabajo

Construye un programa con el lenguaje de programación que creas apropiado para poder simular este proceso durante 8 horas y obtén estadísticas sobre el tamaño promedio de la cola, el tiempo de espera promedio y el grado de utilización de las dos máquinas. Hay que incluir en el programa, como mínimo, un generador de números aleatorios, un ejecutivo que trabaje sobre la lista de eventos y ejecute el código del próximo evento, el código asociado a cada uno de los eventos y el código para obtener las estadísticas.

© Los autores, 2002; © Edicions UPC, 2002.

5 Desarrollo de experimentos de simulación

En los anteriores capítulos se han determinado y descrito las distintas fases básicas del proceso de construcción de un modelo de simulación: análisis y formulación del problema, determinación de objetivos, estudio de los datos de entrada disponibles, diseño del modelo conceptual, y su traducción a un modelo ejecutable por ordenador.

A continuación y en este capítulo, dedicado a lo que pretendemos identificar como el entorno genérico en el que se va a desarrollar un estudio de simulación se presentan, de forma esquemática, algunos aspectos interesantes de los procesos de verificación, validación y acreditación (VV&A) a lo largo de las diferentes fases de un proyecto genérico de simulación.

Esta primera parte pretende aproximarnos a un conjunto de metodologías y herramientas que tienen como objetivo último garantizar la máxima calidad del estudio sobre la base de los conocimientos, objetivos y recursos que hayamos querido o podido destinar a nuestro trabajo. Aunque se trata de un aspecto del universo de la simulación aún en vías de consolidación y sin un grado de utilización excesivamente grande en el sector, los autores consideran fundamental introducir los criterios de rigor y racionalidad que diversos autores como Balci, Sargent o Law y Kelton han conseguido sistematizar a lo largo de estos últimos años.

En los apartados siguientes se plantearán las líneas básicas y los instrumentos recomendados en el ámbito estratégico y táctico entre otros por Kleijnen. Consideramos muy interesante la distinción entre estos dos niveles en la resolución final de un problema atacado mediante las técnicas de simulación: estrategia en cuanto a definir con claridad los escenarios de experimentación que deseamos tener en consideración, desarrollando las aproximaciones necesarias mediante el diseño adecuado de los experimentos como fase final de un proyecto de simulación orientado a la ayuda en la toma de decisiones; táctica para plantear la ejecución real del estudio de la forma más precisa y eficiente, aprovechando el conocimiento que tengamos del sistema y del modelo del mismo, a fin de usar las técnicas que nos permitan la utilización racional de nuestros recursos.

En la segunda parte abarcaremos el diseño de los experimentos mediante aproximaciones factoriales y factoriales parciales o fraccionales. Finalmente, y dentro de un gran grupo conceptual que denominaremos *análisis de resultados,* abordaremos el estudio de los sistemas durante su fase transitoria o en el estado estacionario, la determinación de la longitud y número de repeticiones de las ejecuciones del modelo de simulación ayudándonos de las técnicas para la reducción de la variancia, la determinación de los intervalos de confianza de las medidas de rendimiento o de interés, y los procedimientos de búsqueda del óptimo, los metamodelos y el análisis de sensibilidad.

© Los autores, 2002; © Edicions UPC, 2002.

Nuestro objetivo será, simplemente, utilizar los modelos de simulación con la mayor calidad y eficiencia [KELT95].

Nos planteamos aprender del comportamiento del modelo basándonos en el análisis de los resultados que obtendremos al experimentar con el modelo ejecutable básico de nuestro sistema objetivo. Dicho ejecutable habrá sido convenientemente verificado y habremos sometido al conjunto del proceso a un procedimiento de validación.

Dicho sea de paso, esto último equivale a decir que nuestro trabajo ha sido, de manera explícita o implícita, aceptado por el cliente, quien habrá identificado, en el conjunto que se le ha presentado de objetos y relaciones convenidos, el instrumento de investigación que se corresponde con las expectativas generadas durante las fases de formulación del problema y de elaboración de preguntas a realizar al modelo.

En la práctica profesional de los estudios de simulación se trabaja normalmente con diversos enfoques alternativos en lo que se refiere al tipo de respuesta que pretendemos obtener de nuestro modelo.

En algunos casos deseamos obtener una buena estimación de determinada medida de rendimiento, la cual nos servirá como variable de decisión. Comparamos dicho valor en simulaciones efectuadas con distintas alternativas de configuración del sistema y seleccionamos como solución de nuestro problema aquella combinación de niveles de nuestros factores de experimentación o parámetros que proporcionen el mejor valor rendimiento.

Puede también ocurrir que debamos emplear un enfoque multicriterio en cuanto a las medidas de rendimiento que vayamos a emplear para tomar la decisión. Constituye éste un nivel de complejidad añadido, que nos conducirá al empleo de técnicas de optimización basadas en más de una variable resultado o análisis de resultados multivariante.

En otros problemas, nos interesará determinar si una alternativa puede considerarse mejor que otra o no, independientemente de la precisión absoluta de las medidas de rendimiento. Lo que deseamos es decidir, por ejemplo, si debemos descartar la propuesta de una nueva política de gestión de nuestro sistema porque no podemos demostrar que sea mejor que la actualmente en vigor.

Finalmente, pero sin pretender ser exhaustivos en esta enumeración de aproximaciones posibles a la estrategia de resolución de problemas, podemos pensar en sistemas en los que alcanzar una determinada magnitud o la ocurrencia de un determinado evento dan por finalizada la ejecución de la simulación. En estos casos nos encontraríamos con datos de salida que podrían no ser homogéneos ni comparables, lo que llevaría a una gran complicación en los procesos de evaluación o comparación de alternativas del sistema.

La elección de uno u otro enfoque tiene mucho que ver con las características específicas del sistema que se quiera simular. Un mismo sistema se puede prestar a distintas aproximaciones en función de la información disponible, de la precisión de los datos y parámetros de entrada, de las posibilidades de validación del modelo, etc.

© Los autores, 2002; © Edicions UPC, 2002.

EJEMPLO 5.1 Medidas absolutas: dimensionado del número y tipo de taquillas en un peaje de autopista.

Normalmente el espacio destinado a las taquillas de peaje de una autopista es una zona formada por una batería de estaciones individuales donde se realiza el servicio de cobro a los distintos vehículos que pasan por aquel punto. En general, este espacio es limitado y algunas de las estaciones se construyen de forma que el sentido de uso sea invertible en función de la densidad del tráfico en diferentes tramos horarios, convenientemente segmentados a lo largo del día. Se supone que éstas serán las estaciones que ocupen la zona central de la instalación a fin de facilitar al máximo su reconfiguración en cuanto a reversibilidad.

En cada una de ellas podremos disponer de forma no exclusiva (y desde nuestro punto de vista como simuladores, parametrizable) de: servicio manual con cobro en metálico (monedas) y con tarjeta, automático exclusivamente con tarjeta, automático con tarjeta y metálico, telepago, etc. Determinados tipos de vehículos estarán obligados a circular por el peaje manual (motocicletas, vehículos con más de dos ejes, etc.) mientras que otros podrán ser atendidos en cualquiera de las unidades siempre y cuando dispongan del medio de pago adecuado.

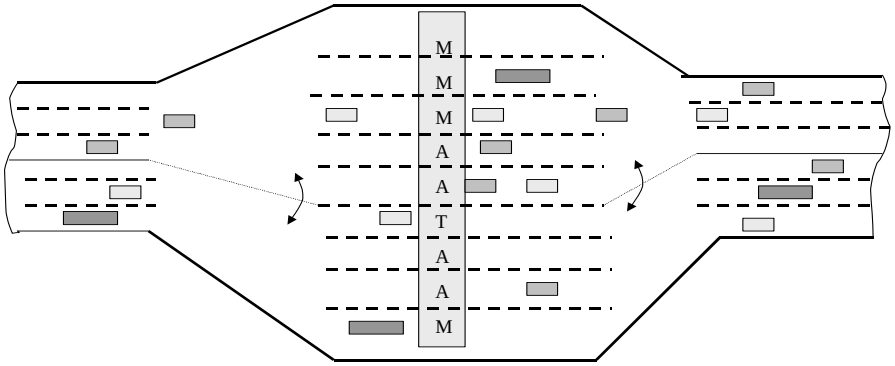


Figura 5.1 Estación de peaje con puestos reversibles

Se trata de determinar las políticas de abertura y asignación de sentido para cada una de las estaciones a lo largo de un período de tiempo, considerando diferentes tipos de día (laborable, festivo, especial,...) y otras características, en función de una carga de trabajo predefinida o del tiempo atmosférico, pongamos por caso. También deberemos tener en cuenta el número de operadores del peaje necesarios disponibles en función de los turnos de trabajo (diario, semanal, vacaciones) y el tiempo de servicio a partir del análisis de los tiempos empleados en sistemas equivalentes por los distintos tipos de usuario. Se da el caso, por ejemplo, que los peajes manuales se sobrecargan en los días punta de verano debido a una mayor afluencia de vehículos extranjeros que prefieren pagar en metálico o no se sienten seguros con los sistemas automáticos, por lo que se forman largas colas que bloquean la totalidad de los carriles y colapsan la autopista.

© Los autores, 2002; © Edicions UPC, 2002.

EJEMPLO 5.2 Evaluación y diseño de una infraestructura de transporte: estudio de la capacidad determinados tramos en una red viaria.

Deseamos evaluar la conveniencia de ampliación de un conjunto de algunos tramos de una red interurbana de vías rápidas en proceso de planificación por parte de las autoridades de tránsito. Estos viales disponen, en la actualidad, de dos carriles en cada sentido. Se conoce la tipología y distribución de la carga actual, modos de transporte, posibles incidencias, etc.. Se prevé asimismo un determinado crecimiento en los próximos años debido a la urbanización de varias zonas vecinas a uno de los nuevos accesos.

En función de los modelos conocidos de simulación de tránsito, comparamos la mejora tras la ampliación en un carril por sentido y sometemos a consideración del ingeniero de tránsito la conveniencia y justificación de la construcción o ampliación de dicha infraestructura.

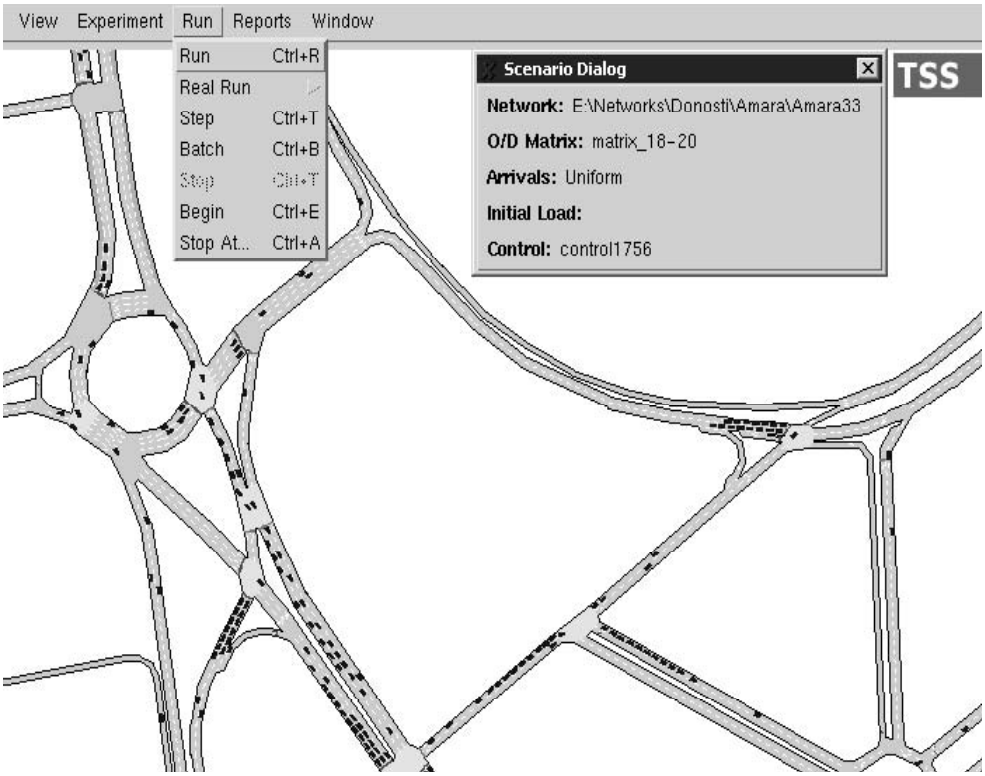


Figura 5.2 Sección de vías urbanas y periféricas modeladas con AIMSUN (reproducido con permiso de TSS)

© Los autores, 2002; © Edicions UPC, 2002.

EJEMPLO 5.3 Determinación del tiempo medio de espera hasta la realización de una transacción en un TPV en un sistema bancario.

Se trata de dimensionar el número y la velocidad de transmisión de unos equipos terminales en los puntos de venta de unos grandes almacenes, que permiten realizar los cobros mediante tarjetas de distintas entidades bancarias y, por tanto, con distintos protocolos de validación y autorización de pago y, en consecuencia, distintos tipos y tiempos de respuesta. El recibo correspondiente también varía entre entidades y el tiempo de impresión puede ser distinto.

Se considerarán errores de lectura y saturaciones en la red en función de la hora del día y del tipo de día (normal, sábado, especiales: días de rebajas o navidades, etc.).

Aunque el modelo puede llegar a ser muy sofisticado, se trabajará a un nivel suficientemente abstracto para obtener una primera aproximación de la cantidad de recursos necesarios.

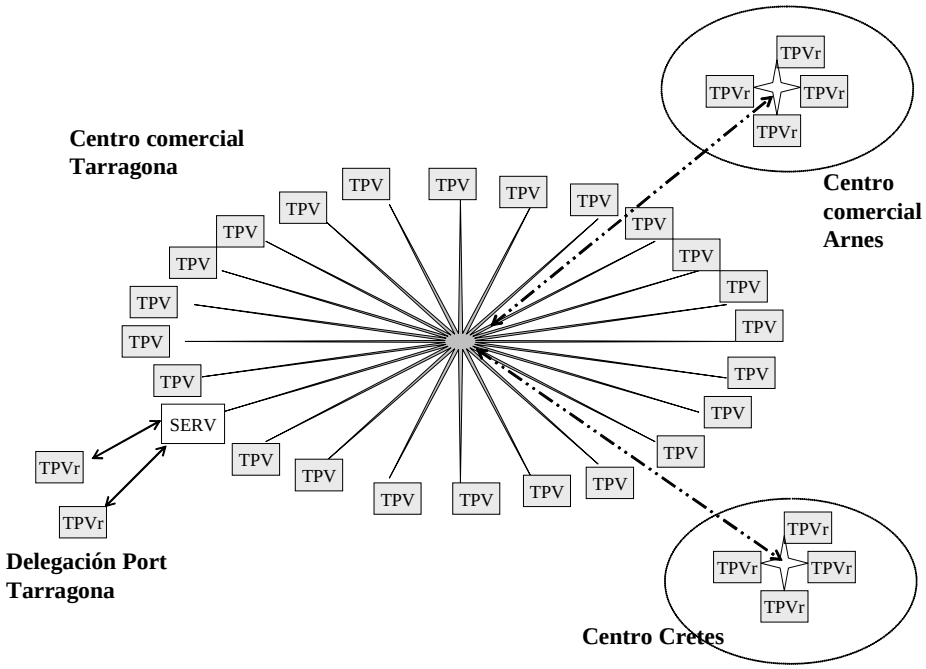


Figura 5.3 Configuración local y remota de TPV

EJEMPLO 5.4 Comparación

El director de almacén de una planta de fabricación de electrodomésticos de la gama blanca cree que una modificación en la asignación de tareas de su equipo de trabajo en la zona de expedición de pedidos al por mayor permitirá una mejora sustancial en el número de pedidos expedidos por día, especialmente los lunes a primera hora, después del descanso semanal. Compararemos las dos alternativas sugeridas por un estudio previo que valorarán escenarios con reasignación y sin reasignación, analizando las diferencias en los resultados de pares de ejecuciones que contemplen cada una de dichas programaciones de tareas en

© Los autores, 2002; © Edicions UPC, 2002.

idénticas, o similares, condiciones de carga, turnos de trabajo, elementos de transporte. Partiremos de una misma configuración, estacionalidad y cantidad de pedidos, teniendo en cuenta la gran variabilidad del mercado en el sector y la posibilidad de averías en los elementos de transporte de la planta.

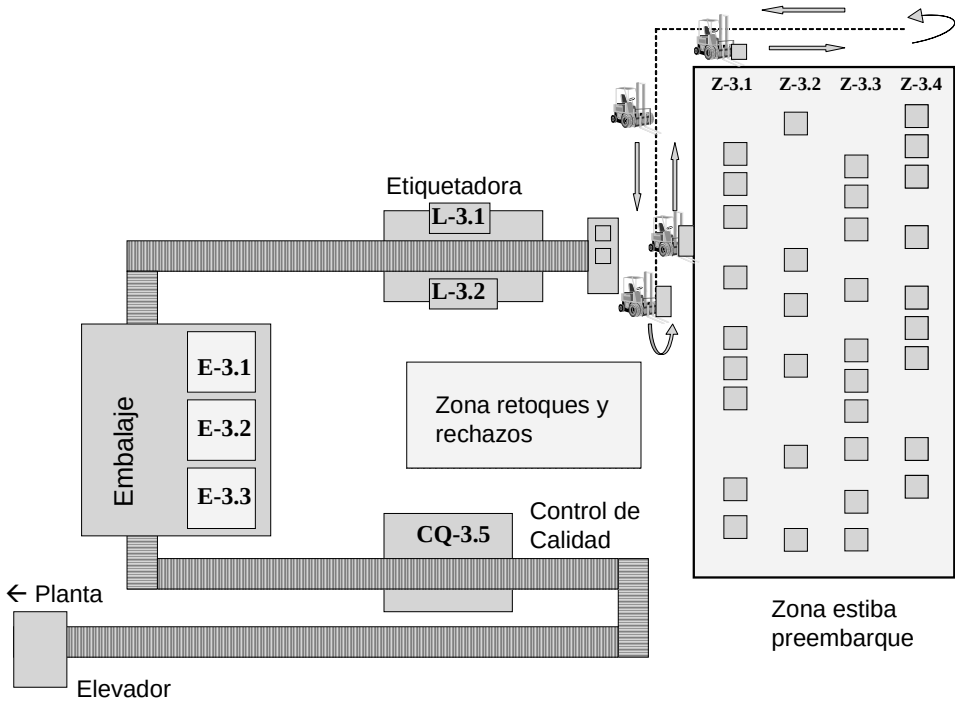


Figura 5.4 Zona expedición en fábrica de electrodomésticos

EJEMPLO 5.5 Estudio de saturación de una central automatizada de atención telefónica (*Call Center)*

Una compañía dedicada a los servicios de atención al automovilista desea dimensionar adecuadamente el servicio de respuesta y tramitación telefónica de incidentes o accidentes de sus asociados y en caso de avería mecánica (figura 5.5).

Esta misma empresa ofrece el mismo servicio convenientemente personalizado a compañías aseguradoras extranjeras, por lo que se ve obligada a disponer de operadores/as con conocimientos de otros idiomas. El personal determina automáticamente el origen de la llamada en función del número de entrada correspondiente que les aparece en un *display* y recogen la misma cuando la pueden atender, es decir, si se pueden expresar en la lengua del cliente de la llamada entrante.

Los operadores identificarán al usuario o asociado, lo localizarán con la ayuda de un sistema de información geográfica y le asignarán un servicio que podrá conducir a una reparación del vehículo o al remolcado del mismo mediante un servicio de grúa. Los medios disponibles podrán ser propiedad de la empresa o alquilados a terceros siguiendo una política de pedido telefónico.

© Los autores, 2002; © Edicions UPC, 2002.

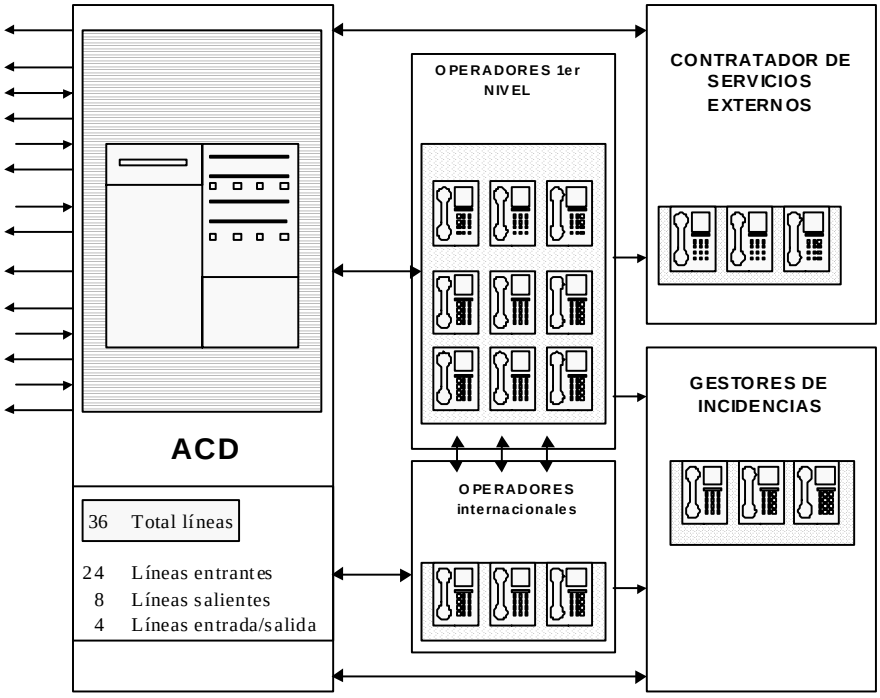


Figura 5.5 Servicio de asistencia basado en una central automatizada de atención telefónica

En definitiva, el ACD o central telefónica automatizada, registrará llamadas entrantes de clientes y salientes de los operadores: las líneas estarán configuradas como de salida, de entrada y de entrada/salida respectivamente, a fin de prever bloqueos en situaciones de saturación (figura 5.6). Se trata de dimensionar el ACD, el número de operadores y sus turnos, el número de recursos propios de la empresa y los que se van a subcontratar y el grado de automatización que se desea de cada una de las operaciones (por ejemplo, se puede diseñar el sistema de llamadas de forma que el cliente teclee directamente en su terminal personal (PDA) o teléfono (UMTS) su número de identificación mediante un diálogo pregrabado).

5.1 Verificación, validación y acreditación en simulación

Como se ha mencionado en la introducción, la complejidad y la responsabilidad asociada a los modelos actuales de simulación, junto con sus implicaciones económicas y sociales, requieren y obligan al establecimiento de procedimientos estructurados y sistemáticos que permitan asegurar unos niveles apropiados de calidad y fiabilidad a lo largo de todo el proyecto.

El problema de la validación de modelos ha sido ampliamente debatido por el mundo científico y está estrechamente conectado con profundas discusiones filosóficas vinculadas a problemas epistemológicos fundamentales.

© Los autores, 2002; © Edicions UPC, 2002.

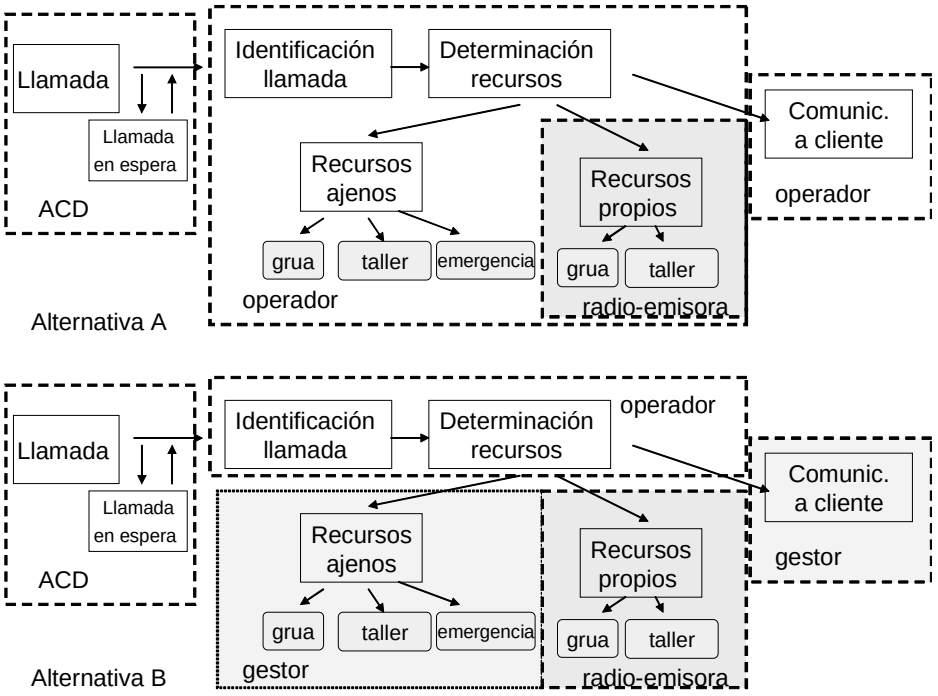


Figura 5.6 Alternativas de diseño de servicio para el problema de la central de atención telefónica

Antes de pasar a plantear una aproximación más aplicada de la verificación, validación y acreditación (VV&A), conviene citar el enfoque más formal de Kleindorfer, O'Neill y Ganeshan [KLEIN98] sobre las aproximaciones metodológicas y las inquietudes de los modelizadores respecto a las guías de la validación, quienes reflexionan sobre las distintas posiciones filosóficas en relación con la validación en simulación: discuten los problemas vinculados a los distintos tipos de argumentación que permiten defender los modelos de simulación, y convierten el problema de la validación en un problema ético en el que el modelizador debe sentirse totalmente responsable e implicado.

El debate sobre el compromiso del asesor respecto a las implicaciones de sus recomendaciones o conclusiones emerge con fuerza en el panorama actual de la ciencia y la tecnología. A las ya complicadas discusiones sobre el valor universal de determinado conocimiento y la infalibilidad de determinados principios o metodologías, debemos añadir en nuestro caso los problemas que emanan de las herramientas computacionales sobre las que se soporta todo proyecto de simulación.

El alejamiento de los usuarios finales respecto de los detalles reales de implementación de los modelos, desde el punto de vista del *software* que los sustenta, exige un esfuerzo adicional de sistematización de los procesos de colaboración entre los equipos de expertos implicados en un proyecto complejo de simulación.

En la literatura sobre V&V encontramos también muchas referencias en términos de verificación, validación y pruebas (VV&T), por ejemplo del propio Balci [BALC00b], aunque aquí hemos preferido darle una perspectiva más abierta incluyendo el concepto de acreditación como elemento de

© Los autores, 2002; © Edicions UPC, 2002.

cierre en el proceso de establecimiento de confianza entre los participantes de un proyecto de simulación. También debe destacarse, por su carácter decisorio en los desarrollos destinados a programas relacionados con la defensa, el establecimiento de la instrucción DoD 5000.61 referente a la adquisición de *software* y la VV&A en M&S ([CONW00], [DoD00, http://www.deskbook.osd.mil] y [DoN99, http://navsmo.hq.navy.mil]). Finalmente, el tema en su conjunto se incluye en la órbita de los problemas vinculados al desarrollo de proyectos de *software* complejos y el aseguramiento de su calidad. El progreso decidido y continuado en estos campos garantizará las mejoras de los futuros proyectos de simulación.

Vamos, en primer lugar, a introducir ciertas definiciones que nos ayudarán a centrar el tema. Nos valdremos del esquema de la figura 5.7, basada en Balci [BALC98] con la incorporación de algunas relaciones y procedimientos adicionales, en el que podemos visualizar de forma simplificada el proceso de construcción y explotación del modelo.

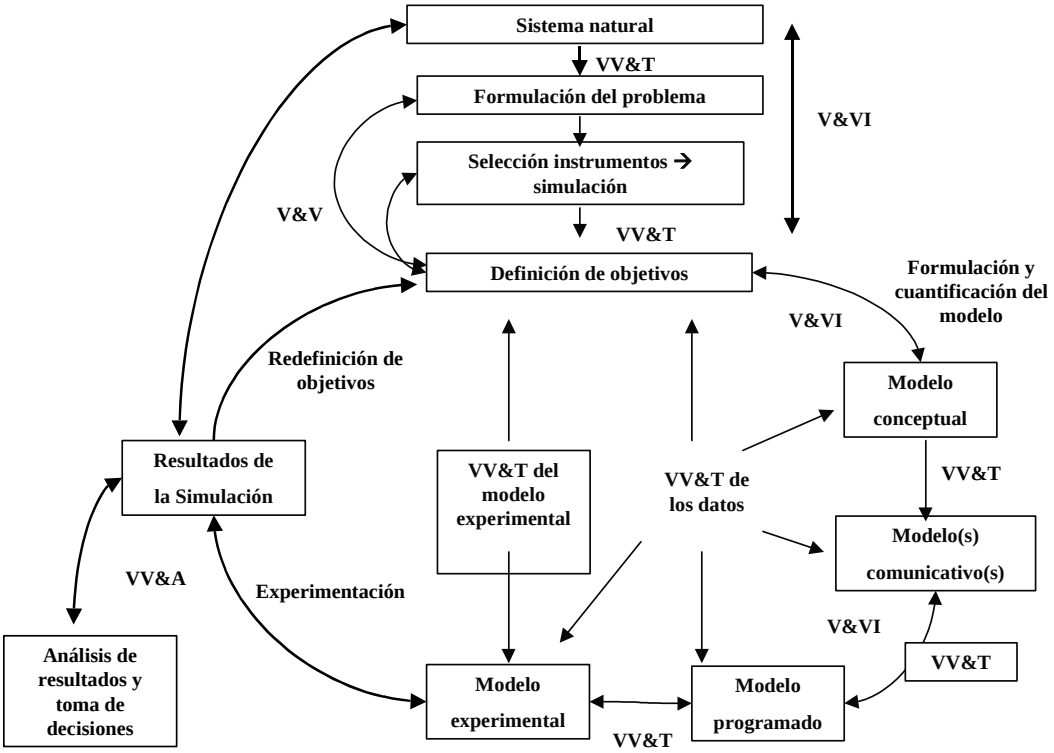


Fig. 5.7 Procesos de validación y verificación (adaptado a partir de [BALC98])

Entenderemos por *validación del modelo* el proceso de determinación de que dicho modelo de simulación es una representación correcta y exacta del sistema en estudio, en función de los objetivos específicos definidos para el problema que pretendemos resolver. O también, cómo determinamos que un modelo computarizado en el marco de su dominio de aplicabilidad posee un grado satisfactorio de precisión de forma consistente con la aplicación pretendida del modelo [SARG99][SCHL].

© Los autores, 2002; © Edicions UPC, 2002.

El segundo concepto que deberemos definir es el de *verificación del modelo*. En este caso nos referimos a la relación entre lo que pretendemos modelar, y que hemos en cierto modo validado según lo especificado previamente, y su implementación real como modelo ejecutable en un ordenador. Es decir, se trata de comprobar que construimos correctamente aquello que ya hemos establecido que refleja la realidad y los objetivos del estudio de una forma suficientemente fidedigna.

Es interesante distinguir entre ambos procesos: *validaremos* que el modelo conceptual se ajusta a nuestras necesidades y podrá responder a nuestras preguntas y *verificaremos* que el proceso de construcción de las herramientas con las que pretendemos experimentar, sigue unas determinadas reglas y condiciones que nos garantizan que nuestro modelo computacional hace exactamente aquello que se supone debe hacer en función de las especificaciones validadas del modelo conceptual.

A lo largo de los procesos de validación y verificación someteremos al modelo a una serie de *pruebas* sobre los datos, o mediante casos de prueba, a fin de evaluar la precisión en el comportamiento del modelo y en la transformación entre los distintos tipos de modelos implicados en el ciclo de vida de un proyecto de simulación.

A partir de aquí llegamos al tercer pilar pata que sustenta la fiabilidad de dicho proyecto: la acreditación. En este caso nos quedamos con la definición de Law y Kelton [LAW99], en la que se afirma que la acreditación de un modelo de simulación es una determinación oficial de que dicho modelo es (simplemente) aceptable para un propósito particular.

Este concepto debe ser necesariamente vinculado con el de certificación. Ésta constituiría el proceso mediante el cual garantizamos documentalmente que una determinada respuesta o, pongamos por caso, las funcionalidades y el comportamiento de determinados componentes (*software*) de un entorno de simulación realizan exactamente aquello para lo cual han sido diseñados. Una parte importante de lo que se entiende por certificación constituye un verdadero apoyo a la componente de relación con el cliente. Facilita lo que venimos a denominar *credibilidad de un proyecto de simulación.* Ésta se identifica con la confianza que el cliente final podría otorgar a un modelo, a la información proporcionada, o a los resultados y conclusiones que se derivarían de su utilización.

Asegurar la calidad total en la modelación y simulación (M&S) implica la medición y la apreciación de una variedad de características tales como la exactitud, la eficiencia en la ejecución, mantenibilidad, portabilidad, reusabilidad y usabilidad de los distintos componentes del proyecto, entendido en un sentido muy amplio.

5.1.1 Principios para la validación, verificación y acreditación en simulación

Balci y otros autores [BALC95] [BAN99] [BALC00a] [BALC00b] [SARG99] desarrollan todo un cuerpo de conocimiento sobre los principios y las técnicas que deben guiar la VV&A en simulación. A continuación presentamos, entre la gran cantidad de métodos y propuestas en la literatura sobre el tema, aquello que consideramos más relevante y útil para el tipo de problemas de aplicación práctica que enfoca el presente texto. El lector más interesado en esta importante temática y que desee conocer o incorporar más herramientas de validación en su modelo puede consultar la bibliografía reseñada a lo largo del apartado.

En su esfuerzo de sistematización, Balci propone el establecimiento de una serie de principios que puedan ayudar a los implicados en un proyecto de simulación a entender qué representa la VV&A

© Los autores, 2002; © Edicions UPC, 2002.

como motor de éxito del mismo. Relacionamos a continuación algunos de éstos y las aportaciones derivadas de la experiencia de diversos autores en problemas de simulación.

Principios de buena práctica desde el punto de vista de la VV&A:

- Conviene desarrollar las actividades de VV&A durante todo el ciclo de vida del estudio de simulación, lo cual nos permitirá detectar todo tipo de deficiencias y problemas antes de que sean irremediables o de muy costosa solución. Para ello Balci, y previamente Ören [ÖREN87], proponen crear, a lo largo de todo el estudio, un departamento específico e independiente (SQA, *Software Quality Assurance*), encargado de garantizar la calidad de la simulación. Arthur y Nance [ARTH00] abogan, además, por la introducción del concepto de validación y verificación independiente (V&VI) como elemento clave e imprescindible para lograr reducir la curva de coste de generación de *software* en general y citan, específicamente, los esfuerzos de grupos como los del Software Engineering Institute (SEI) de la Universidad Carnegie Mellon , creado por el DoD, o los vinculados al IEEE.

- Un modelo no debe ser considerado intrínsicamente como absolutamente bueno o absolutamente malo. Podemos emplear algún tipo de medida subjetiva que dé soporte al grado o ratio de credibilidad que asignamos a un determinado modelo. Aunque no esté muy extendido en nuestro entorno cultural, deberíamos poder formular una frase del tipo “puede afirmarse con un grado de credibilidad del 95% (no confundir con expresiones del mundo de la estadística sobre el grado de precisión o de confianza de alguna medida, pues no pretendemos llegar tan lejos en el grado de formalización) que el proceso de carga del sistema está modelizado de forma fiel y fiable”.

  No olvidemos que el modelo de simulación se construye según los objetivos deseados para el proyecto. Su credibilidad se juzga de forma coherente respecto a esos mismos objetivos. Depende de la importancia de cada uno de ellos que deberemos exigir un grado de precisión o fiabilidad mayor o menor cuando realicemos una evaluación del modelo en detalle. En definitiva, la credibilidad para la cual el modelo es verificado, validado y acreditado sólo deberá considerarse en las condiciones prescritas o en el dominio de aplicación de aquél.

- Un estudio que incorpore una VV&A efectiva requiere preservar la independencia de sus analistas a fin de evitar prejuicios causados por personas con opinión sesgada. Esto tiene mucho que ver con los problemas derivados del mantenimiento de la imagen de las personas o consultoras a las que se encarga el estudio, quienes pueden ver afectada su neutralidad por los intereses presentes y futuros de su organización.

- La VV&A es difícil, por lo que requiere creatividad y visión para diseñar las pruebas. Será fundamental la experiencia en otros proyectos afines y en aspectos metodológicos relacionados con el trabajo en desarrollo o con la cultura tecnológica de los actores implicados.

- No es posible probar por completo el modelo contra todos los *inputs* posibles, por cuestiones de tiempo, recursos y racionalidad intelectual. Los datos de entrada que nos permitirán caracterizar los distintos procesos y los datos destinados a la validación deben ser suficientes en función del tamaño y complejidad del sistema en modelación y, como se ha mencionado anteriormente, de los objetivos y la importancia de las respuestas que pedimos al modelo.

© Los autores, 2002; © Edicions UPC, 2002.

- La VV&A debe planificarse y ser documentada, tal como presentan Brade [BRAD00] y Balci, Ormsby, Carr y Saadi [BALC00b] (figura 5.8), mediante una formalización de la documentación, la descripción de las listas de control, una definición precisa y consensuada de los indicadores y el diseño de los niveles de presentación de los resultados. Un comentario específico merece la cuestión de los indicadores en cuanto a las ventajas que conlleva su utilización de cara a la evaluación progresiva y final del proyecto. Aunque no hay duda de la utilidad de su inclusión para la búsqueda de la calidad, de su uso ligero y sin criterio se deriva a menudo una adaptación de los procedimientos y buenas prácticas a la obtención de "buenos" indicadores, independientemente de la aplicabilidad real de los mismos o a la necesidad de cambio y mejora continua de los mecanismos de evaluación y prueba.

Credibilidad del modelo conceptual
Credibilidad del modelo
Credibilidad de la implementación
Credibilidad de la integración
Credibilidad de los datos
Calidad de la gestión de configuraciones
Calidad global del producto
Calidad de la documentación

Figura 5.8 Plan de VV&A para una aplicación de modelización y simulación (adaptado de Balci, Ormsby, Carr y Saadi [BALC00b])

- Deben prevenirse los errores en la formulación de hipótesis, por las repercusiones en coste o por la criticidad del sistema en estudio. Habitualmente hablamos de dos tipos de errores, aunque incluimos un tercer tipo próximo a la problemática de los proyectos de simulación.

1. Error de tipo I o riesgo del constructor del modelo, cuando los resultados de la simulación sean rechazados pese a ser, de hecho, suficientemente creíbles.
2. Error de tipo II o riesgo del usuario del modelo: resultados de la simulación que no pueden ser considerados válidos son aceptados como *suficientemente* válidos.
3. Error de tipo III: se resuelve el problema equivocado debido a que la formulación del problema, establecida en las fases previas o iniciales del proyecto, no contiene adecuadamente "el problema real" [BALC81].

   Estos y otros errores menores o de implementación deben detectarse lo antes posible durante el ciclo de vida de la modelización y de la simulación, y el esfuerzo debe especialmente centrarse en la fase de especificación de los modelos, en los aspectos funcionales, de datos o de relación con el usuario.

- El problema de la respuesta múltiple -que se analiza al final del capítulo- debe identificarse rápida y claramente y ser resuelto adecuadamente. Nos basaremos en procedimientos de análisis estadístico multivariante.

© Los autores, 2002; © Edicions UPC, 2002.

- Probar con éxito cada uno de los submodelos componentes no significa que el modelo completo adquiera automáticamente credibilidad. El modelo en su conjunto debe ser de todos modos analizado.
- Un problema interesante surge durante la aplicación de la técnica por la que se comparan los resultados de la simulación con datos observados directamente del propio sistema real, cuando ambos utilizan los mismos datos de entrada. El problema de la validación doble trata de la validación de estos datos de entrada teniendo en perspectiva la posibilidad de que hayan sido caracterizados de forma defectuosa en el modelo. Ello podría conducir a aceptar el modelo como bueno al haber interpretado como evaluación positiva o como argumento de validación la similitud (por pura coincidencia) entre los resultados originales el sistema real y los obtenidos tras la ejecución del modelo.
- La validez del modelo de simulación es sólo una condición suficiente [BALC95] y no garantiza la credibilidad y aceptabilidad de los resultados de la simulación. Dicha aceptación depende de otros muchos factores, como el que se hayan formulado correctamente los objetivos del estudio y caracterizado oportunamente los detalles relevantes del problema.

Finalmente, debe quedar claro que de nada nos sirve haber superado satisfactoriamente una serie de pruebas durante la VV&A de nuestro estudio si éste no es aceptado, por falta de credibilidad, por quien nos haya encargado el proyecto de simulación, pues esta aceptación y la utilización de nuestra simulación y/o de nuestras propuestas es también nuestra responsabilidad.

5.1.2 Técnicas de VV&T

Consideramos aquí cuatro familias básicas en cuanto a las técnicas disponibles para la VV&T ([BALC95] [BANK98] para una taxonomía exhaustiva y su explicación detallada). Describiremos a continuación de forma breve cada una de estas categorías: informales, estáticas, dinámicas y formales, deteniéndonos solamente en algunos de los instrumentos cuya aplicación parece más extendida. Dejaremos explícitamente fuera del alcance de esta sección las técnicas directamente vinculadas a aspectos de la ingeniería del *software*.

Técnicas informales

Huyendo de excesivos formalismos matemáticos, éstas son las técnicas más utilizadas en la práctica de la simulación. Se basan en el aspecto general, en la opinión e intuición de expertos independientes, en los procesos clásicos de auditoria del *software*, en la realización de inspecciones, revisiones, comprobación de documentación, recorridos de control de fallos, etc.

Muchas de estas técnicas se fundamentan en instrumentos de seguimiento y evaluación de otros tipos de proyectos y en prácticas universalmente aceptadas para la auditoria y control de los mismos. Se trata de utilizar el rigor y la independencia en el análisis, junto con el sentido común y la experiencia. Es importante la constitución de equipos multidisciplinares de evaluación y la definición precisa de las funciones y responsabilidades de cada uno de sus miembros. Se logrará así una mejora perceptible de la calidad del sistema, aseguraremos el cumplimiento de los objetivos de los proyectos a un coste razonable y controlado, e introduciremos una nueva perspectiva entre los miembros encargados del desarrollo del entorno de simulación.

© Los autores, 2002; © Edicions UPC, 2002.

Cabe incluir específicamente mecanismos de control de la gestión del proyecto así como de los aspectos financieros del mismo. En este mismo ámbito, deberá preservarse al máximo la independencia financiera del grupo evaluador para, así, garantizar su objetividad. También debemos aceptar el hecho obvio de que la inclusión de estas técnicas y sus procedimientos asociados implican un incremento de coste directo en el proyecto, que se estima entre 5 y el 20% del monto total. No obstante, se supone que la VV&A incorpora método y rigor al proyecto e incidirá favorablemente en el respeto de los tiempos de desarrollo (hoy se estima que más del 75% de los proyectos de *software* exceden ampliamente los tiempos de entrega previstos).

Técnicas estáticas

Se basan en la evaluación de las características de diseño del modelo estático y del código fuente. Algunos entornos de desarrollo de proyectos de simulación ya constituyen en sí mismos, a través de sus procedimientos de traducción automática de modelo a código ejecutable, mecanismos de verificación adecuados. Estas técnicas permiten obtener información sobre la estructura del modelo, las técnicas de modelización utilizadas, los datos y los flujos de control, etc. Entre las más interesantes destacaremos la definición de gráficos y listas causa-efecto, y el análisis de concurrencias, de los flujos de control y de las transiciones entre estados. En especial debe procederse al análisis de la dependencia entre los datos, al análisis estático de las interfaces del modelo y de usuario y a la evaluación de los objetivos de la simulación en relación con la presencia de objetos o entidades que hagan viable su cumplimiento.

Técnicas dinámicas

Nos referimos aquí a la evaluación del modelo en tiempo de ejecución, introduciendo mecanismos de recolección dinámica de información que va a ser relevante para los procesos de VV&T. Se trata de explotar nuestro control sobre la codificación del modelo para articular medios de seguimiento o monitorización, tanto de aspectos relacionados con la plataforma *hardware* y *software* sobre la que se ejecutará la simulación, como son las evaluaciones de versiones *alpha* y *beta*, pruebas *top-down* y *bottom-up* que afectan a la estructura modelo/submodelos, el análisis de acceso y seguridad, seguimiento de trazas, la aplicación de técnicas de caja-negra, el chequeo de interfaces de datos, de edición del modelo y de usuario, el análisis de sensibilidad frente a variaciones en los datos de entrada y en valores o condiciones extremas, las pruebas de estrés, etc. Un gran número de técnicas estadísticas son de aplicación en este epígrafe, en función del tipo de sistema y de análisis de resultados proyectado: regiones e intervalos de confianza, análisis de factores, análisis de la variancia simple y multivariante, tests no paramétricos, análisis de regresión y series temporales, entre otras.

Dedicaremos, más adelante, un comentario específico a los aspectos relacionados con la representación dinámica de la evolución del sistema, en especial aquello que afecta a las necesidades de visualización, animación y realidad virtual e interacción de usuario en general.

Técnicas formales

El objetivo aquí es la formalización al máximo nivel para alcanzar la garantía de corrección del modelo. Es esta aún una cuestión pendiente en el ámbito de la simulación de problemas complejos, objeto del presente libro. Se trata de aplicar conceptos relacionados con la inferencia, la inducción, el cálculo y la transformación de predicados.

© Los autores, 2002; © Edicions UPC, 2002.

5.1.3 Validación

Un modelo es válido si responde (correctamente) a las preguntas que nos queramos formular sobre el sistema real, es decir, es válido si nos es útil. El proceso de validación debe dar repuesta a la pregunta: ¿es el modelo de simulación una representación adecuada del sistema correspondiente en el mundo real?

Aplicaremos algunos de los principios y técnicas antes mencionados, enmarcándolos dentro del ciclo de vida de un proyecto de simulación. No nos detendremos en todos los pasos, perfectamente detallados en los trabajos de Balci y otros autores, aunque destacaremos algunos de ellos ayudándonos del diagrama de la figura 5.9.

Entre otras acciones, trataremos de comparar los resultados del modelo con datos que consideramos correctos (preferentemente de un sistema real). Si el sistema aún no existe, trataremos de validar una parte del mismo mediante el análisis de algunos submodelos que nos sean conocidos.

La acreditación requiere la evaluación y medición de elementos cualitativos y cuantitativos del modelo, mediante la utilización del conocimiento de los expertos en muy diversos campos, tanto de la organización, del proceso de fabricación, de la construcción de modelos de simulación, informáticos o estadísticos. Debe preservarse la coherencia y honestidad del análisis mediante la utilización de procedimientos de evaluación independiente de escenarios [SARG98] [BALC98][ARTH00].

La búsqueda y mantenimiento de la credibilidad a lo largo de todo el ciclo de vida del proyecto es una tarea que consume tiempo y dinero. Lamentablemente, se suele dedicar a esta fase muchos menos recursos de los necesarios; en los proyectos, se sacrifica la documentación y la V&V.

Dado que todo el proceso requiere de un fuerte soporte computacional, cuántas pruebas deben hacerse depende de los objetivos del problema. Las pruebas pueden continuar hasta tener confianza suficiente, lo cual vendrá determinado por los objetivos fijados en el proyecto.

Como ya hemos mencionado, es muy importante establecer un programa de *aseguramiento de la calidad de la simulación* (SQA). La SQA va más allá de la validación y la verificación, y analiza y se responsabiliza de características como la mantenibilidad, reusabilidad y usabilidad.

Finalmente, la subjetividad es y formará razonablemente parte de la evaluación de la credibilidad en los estudios de simulación complejos.

Algunos comentarios sobre validación

Así como podemos considerar que la validación mejora y se facilita si se dispone de medidas de pruebas reales, *outputs* de otro modelo validado y los datos observados, la validez no mejora necesariamente si nos limitamos a una revisión somera del modelo, o a realizar únicamente comprobaciones sobre la matemática y la lógica del mismo, o validando exclusivamente los submodelos, o bien simplemente contrastando las predicciones del modelo con los datos utilizados para la construcción del propio modelo.

Prerrequisitos para la validación

Desde un punto de vista más operativo, debemos tratar de identificar aquellas situaciones realmente observables y medibles que nos permitirán caracterizar los datos y el modelo de simulación.

© Los autores, 2002; © Edicions UPC, 2002.

Deberemos controlar la constancia de la estructura y la constancia frente a variaciones en condiciones no especificadas en el modelo.

Naturalmente intentaremos recolectar gran cantidad de datos, tantos como nos sea posible. En este sentido, más adelante relacionamos algunos de los requisitos que exigiremos tanto al proceso de recolección de los datos como a estos mismos, pues su calidad nos será imprescindible tanto si trabajamos con un enfoque dirigido por la traza (*trace-driven*) como con el basado en la programación de eventos, explicado en el capítulo 4.

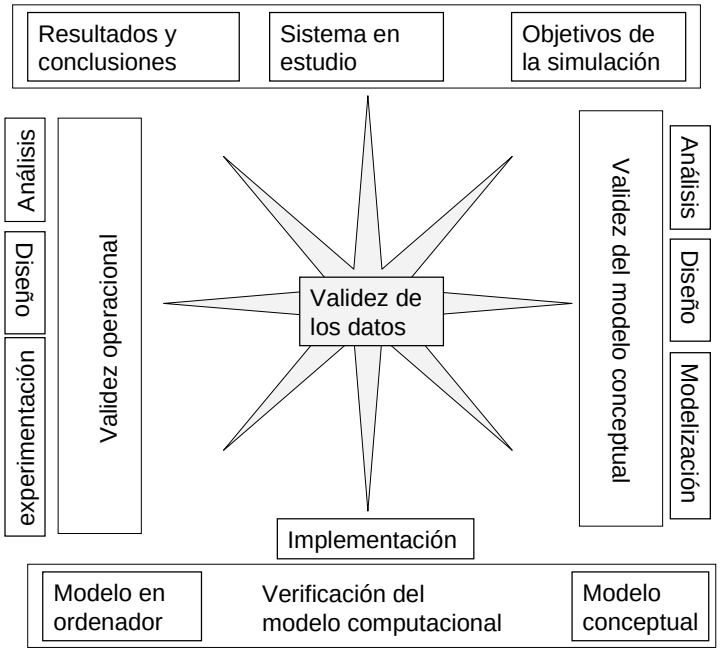


Figura 5.9 Ciclo de vida de la simulación y procesos de V&V

Sólo son validables los modelos con situaciones validables [LAW00]. La validación del modelo para situaciones validables sólo es necesaria si el modelo se pretende usar para fines predictivos. En consecuencia, en determinadas circunstancias, podemos utilizar modelos no validables si ello fuera necesario.

Proceso de validación

Un modelo se desarrolla para un propósito específico y su validez se mide respecto a ese propósito. La arquitectura básica del proceso de validación se estructura alrededor de tres niveles de actuación:

1. El equipo de desarrollo del modelo realiza una apreciación subjetiva (es el método más común) basándose en tests y evaluaciones como parte del mismo proceso de desarrollo.

© Los autores, 2002; © Edicions UPC, 2002.

2. Debe articularse un segundo nivel en el que se garantice la verificación y validación "independiente" (V&VI). Una vez desarrollado el modelo, éste es evaluado independientemente de los diseñadores y de los usuarios. Los resultados indicarán la validez del modelo.

3. El tercer método asigna ponderaciones o pesos a ciertos aspectos del modelo. Se evalúa cada aspecto y se obtiene una calificación global. El modelo se acepta si dicha calificación es mejor que otra preestablecida como aceptable. Aunque esta aproximación es la menos usada, no por ello es menos interesante y constituye un paso intermedio hacia los procesos, en general heurísticos, que pretenden llegar a la solución "óptima".

Validez de los datos

Los datos son necesarios en la simulación para tres propósitos:

- Construcción del modelo conceptual.
- Validación del modelo.
- Desarrollo de experimentos con el modelo validado.

Se precisan datos suficientes y con garantías para construir las teorías que conformarán el modelo, para desarrollar las relaciones matemáticas y lógicas del mismo y para probar los supuestos del modelo. Si no se dispone de datos suficientes, el nivel de confianza del modelo puede llegar a ser muy bajo.

En [BALC00c] se puede encontrar una valoración de los requerimientos para la credibilidad de los datos, basada en tres conceptos: la calidad de los datos propiamente dichos (figura 5.10), del proceso de recolección de los datos y de la calidad de la documentación sobre los mismos.

Accesibilidad/disponibilidad
Precisión
Claridad
Completitud
Consistencia
Actualidad/vigencia
Precisión
Relevancia
Resolución
Reputación
Trazabilidad

Figura 5.10 Requerimientos de calidad en los datos

Vamos a aplicar brevemente estos requerimientos al ejemplo 5.3 sobre el terminal de punto de venta TPV y su relación con el subsistema bancario de transacciones.

© Los autores, 2002; © Edicions UPC, 2002.

EJEMPLO 5.6

Dada la criticidad del sistema al que nos referimos, la calidad de los datos es fundamental. Cualquier pequeño error o suposición aproximada, puede degradar considerablemente la validez del modelo. La información sobre determinados fenómenos, especialmente aquellos que repercuten sobre la duración de las operaciones, los tiempos de respuesta, o la tipología de las transacciones, y contemplar variantes como errores, correcciones o fallos en la comunicación entre nodos tendrán una gran importancia en los resultados obtenidos a lo largo del estudio.

¿Qué requerimos en referencia a los datos? En primer lugar, *accesibilidad* a los datos, en cuanto éstos estén disponibles y nos los puedan suministrar con las debidas condiciones de seguridad y respeto a la privacidad. También deberemos fijar el tiempo de entrega de los datos, estableciendo los correspondientes límites y penalizaciones o determinando las condiciones de reprogramación del proyecto en caso de alteraciones en las cláusulas del contrato.

Precisión y *claridad* de cara a la correcta interpretación de los datos sobre los protocolos de comunicación, las características de los TPV, la capacidad de las líneas o ancho de banda, etc. Identificaremos la información incompleta o parcial, detectaremos los *retrasos* en su entrega y exigiremos la corrección de los ficheros entregados. Requeriremos datos de las transacciones de compra generadas a lo largo de la semana, diferenciándolas por días y tipo de día (laborable, pre-fiestas como Navidad o Reyes) y el nivel de actualidad o *vigencia* de la información en función del momento de su recolección.

Descartaremos datos no relevantes o de dudoso *origen*, identificaremos fehacientemente los datos según su origen y la forma como han sido recolectados (comercio, tipo de comercio, entidad gestora de la transacción, entidad financiera del comercio, entidad financiera del cliente, etc., y, si la *fuente* es una encuesta, los datos monitorizados de la red, las especificaciones de los protocolos de comunicación y de seguridad y fiabilidad de las transmisiones, etc.).

Validación del modelo conceptual

A través del proceso de validación del modelo conceptual pretendemos determinar dos cuestiones:

1. si las teorías y suposiciones del modelo son correctas (linealidad, independencia, estacionalidad, llegadas que siguen una distribución de Poisson, etc.), con ayuda de métodos estadísticos (ajuste, distribución de datos, estimación de parámetros, etc.);
2. si la estructura, la lógica, representaciones matemáticas, y las relaciones causales del modelo son congruentes con la representación del modelo del problema entidad (técnicas principales: *face validation*, consulta a expertos y seguimiento de trazas).

© Los autores, 2002; © Edicions UPC, 2002.

Verificación del modelo computarizado

La verificación del modelo en el ordenador asegura la adecuación y corrección del programa computacional y de su implementación. Se aplican aquí algunas de las técnicas de la VV&T más cercanas a la tecnología y a la ingeniería del *software*, entre ellos el diseño orientado a objetos, el diseño *top-down*, la programación estructurada, y el desarrollo y mantenimiento de módulos y componentes.

Una vez se ha desarrollado el programa y se han eliminado todos los pequeños errores, debe probarse la precisión y la exactitud del mismo, mediante dos familias de técnicas básicas:

- *Pruebas estáticas*: el programa se revisa con técnicas como las pruebas de exactitud, el *walkthrough* y se realizan exámenes de las propiedades estructurales del programa.
- *Pruebas dinámicas*: el programa es ejecutado en diferentes condiciones y los resultados son usados para determinar si el programa y su implementación son correctos. Pruebas: bottom-up, top-down, mixtos, y obtención y análisis de trazas.

Validez operacional

Se busca saber si el comportamiento de las salidas del modelo tienen la exactitud requerida de acuerdo con el problema y con su aplicabilidad. Aquí es donde se aplican muchas de las pruebas conocidas de validación y verificación. La selección de las pruebas suele depender del equipo de desarrollo del modelo y de la observabilidad del mismo.

Para obtener un nivel de confianza alto en el modelo y en sus resultados se requieren por lo menos dos comparaciones entre el modelo y las entradas y salidas de sistema real. Hay tres formas de hacer esto:

1. Comparación gráfica de los datos del modelo con los datos del sistema real (histogramas, diagrama de Box, etc.).
2. Intervalos de confianza para las medias, varianzas o distribuciones para diferentes salidas del modelo y del sistema real.
3. Pruebas de hipótesis, para medias, varianzas, distribuciones y series de tiempo, para las salidas del modelo y del sistema real.

Animación como instrumento de soporte a la validación

La *animación* o *visualización interactiva* es útil especialmente si la consideramos como una herramienta a un primer nivel para la depuración de errores y como un mecanismo básico que nos facilita el proceso de validación y, por lo tanto, de aceptación.

También, evidentemente, se muestra como un fuerte argumento de venta o convencimiento a la hora de presentar una solución o de seducción al contratar un servicio de consultoría en el que se utilice la simulación. El futuro usuario o cliente puede verse más identificado con un modelo que le resulta "familiar" pues reproduce, en cierto modo, la realidad que él conoce. En la actualidad, algunos entornos de desarrollo de proyectos de simulación poseen o pueden ser conectados a módulos de visualización en realidad virtual (RV) (es el caso de Arena, Witness, LeanSIM, etc.)[CASA99]. En

© Los autores, 2002; © Edicions UPC, 2002.

otros casos nos ayudamos de instrumental especializado para la RV, como puede ser el entorno CAVE desarrollado por la Universidad de Michigan, las mesas de visualización en RV como la diseñada en el centro de realidad virtual CRV/Gedas-UPC, cascos, gafas y otros entornos inmersivos, lo que añade un nivel importante de credibilidad al experimento, así como facilita la utilización de la simulación en procesos de formación.

No debemos, sin embargo, confiar exclusivamente en nuestro poder de observación, tanto por nuestras propias limitaciones en cuanto a tiempo disponible para ello, como por la dificultad y poca capacidad de seguimiento de diversos sucesos o procesos simultáneos. Por ejemplo, no seremos capaces de tener una visión de la evolución de variables que nos describan, en la pantalla de simulación, posiciones de planta situadas en lugares muy alejados, pues probablemente estarán en ventanas de representación distintas.

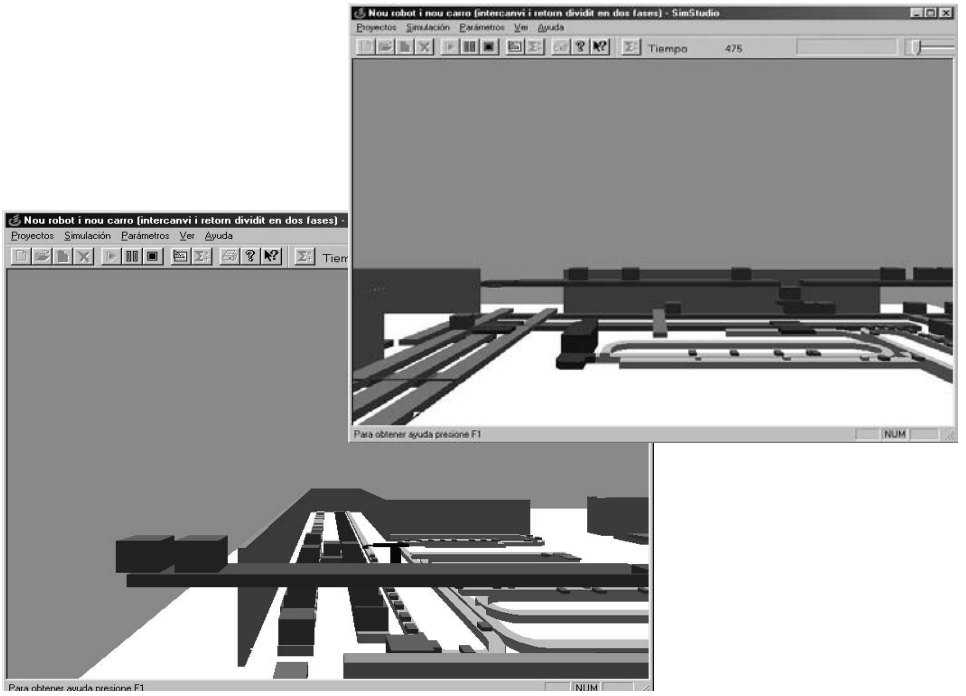


Figura 5.11 Simulación en 3D de una planta de expedición de productos farmacéuticos

Naturalmente, estos problemas son relativamente fáciles de solventar y podemos tratar de agrupar la representación dinámica de los valores instantáneos de dichas variables en zonas o pantallas comunes. Este proceso puede ser complicado y laborioso. Acumular un exceso de información en la misma pantalla suele llevar a confusión. En cualquier caso, tampoco es probable que nos quedemos observando el monitor durante horas, a la espera de que acontezca algún fenómeno de interés.

Son pues necesarios, adicionalmente, otros procedimientos de análisis más prácticos, cómodos, fiables, seguros, y, en definitiva, más viables. Entre estos, se están desarrollando entornos de análisis basados en sistemas inteligentes, que asistirán a los expertos a lo largo de todo el proyecto de

© Los autores, 2002; © Edicions UPC, 2002.

simulación y que facilitarán asimismo la selección e interpretación de herramientas estadísticas sofisticadas y de sus resultados.

Documentación

La documentación relacionada con los procesos de validación y la verificación es importante para el convencimiento de los distintos niveles de usuarios de que el modelo está bien y por ello debe incluirse en la documentación final del modelo de simulación. Deben incluirse las especificaciones de los tests, las evaluaciones realizadas, datos, resultados, etc.

El resumen de documentación debe contener una tabla de evaluación para la validez de los datos, la validación del modelo conceptual, la verificación del modelo computarizado, la validez operacional, y el resumen general.

Figura 5.12 Pantalla del visualizador Getram-3D de AIMSUN (imagen cedida por TSS)

EJEMPLO 5.7 Aplicación del procedimiento recomendado

Retomamos aquí uno de nuestros ejemplos iniciales, en concreto, el ejemplo 5.5 para el dimensionado de una central automatizada de atención telefónica (*call center*). Un simulador, desde la óptica de la credibilidad del modelo, puede ser utilizado como ayuda cuantitativa o cualitativa. En primer lugar, asesorará a quien toma la decisión (a nuestro interlocutor en la negociación) y, a continuación, si hace "predicciones satisfactorias", podrá ser usado realmente para la toma de decisiones.

En nuestro caso seguiremos como mínimo los siguientes pasos:

- Especificaremos los *niveles mínimos de aceptación* para la verificación y la validación, antes de desarrollar el modelo.
- Especificaremos el *nivel de exactitud* requerido en las variables de salida del modelo.

© Los autores, 2002; © Edicions UPC, 2002.

- Probaremos los supuestos y teorías del modelo.
- En cada iteración, realizaremos al menos la prueba sobre *validez de aspecto (face validity)* del modelo conceptual.
- En cada iteración, exploraremos al menos el comportamiento del modelo, usando el modelo computarizado.
- Como mínimo en la última iteración del modelo, haremos dos comparaciones entre el comportamiento de los datos del modelo y los del sistema real.
- Elaboraremos la documentación de la validación para incluirla en la documentación del modelo de simulación. Gestionaremos adecuadamente el control de versiones.
- Si el modelo ha de ser usado durante un largo período de tiempo, revisaremos regularmente la validez del mismo, para lo cual estableceremos un procedimiento rigurosamente temporizado y con mecanismos que certifiquen su realización.

La verificación y validación del modelo son críticas en el desarrollo de un modelo de simulación. Desgraciadamente aún no existen pruebas sencillas y específicas para determinar si los modelos son correctos o no. No se ha establecido, además, un procedimiento sistemático y universal para seleccionar las técnicas a usar. Cada proyecto de simulación nuevo, representa un nuevo reto.

Costes asociados con la verificación, validación y acreditación en simulación

Vamos finalmente a considerar el problema de la VV&A desde la óptica de los costes que implican tanto su implantación en nuestro proyecto, como los inconvenientes por no realizarla o no hacerlo adecuadamente [PACE95].

Elementos de coste asociados a la realización de la VV&A

Ya hemos mencionado el peso creciente de los recursos dedicados a VV&A en los proyectos de simulación. Relacionamos, a continuación, algunas de dichas actividades que se supone generan, en un proyecto, un coste adicional significativo:

- desarrollo y modificaciones de planes de VV&A
- validación de requerimientos/ validación del modelo conceptual
- verificación de que el diseño y la implementación de la simulación satisfacen los requerimientos de la simulación
- recolección, generación y organización de los datos acerca de los sujetos a los que procede validar los resultados
- validación / acreditación / certificación de los datos de entrada

© Los autores, 2002; © Edicions UPC, 2002.

- determinación de la adecuación del equipo de trabajo, *hardware* y otro equipamiento que pueden implicarse en la simulación
- validación de la respuesta, rendimiento y de los resultados de la simulación
- revisión de la acreditación
- otros difícilmente diferenciables de los costes normalmente asociados al desarrollo de un proyecto de simulación

Coste de simulaciones incorrectas o sin credibilidad

Vamos a describir a continuación algunas situaciones creadas a partir de un deficiente seguimiento de un proyecto de simulación desde el punto de vista de la VV&A. En la vida profesional nos encontramos muchas empresas que se muestran reticentes a la utilización de la simulación. Ello no es debido tanto a su desconocimiento de la tecnología en cuestión, como a malas experiencias, que actúan como precedentes disuasorios y que impiden, a menudo, que el proyecto se lleve finalmente a cabo.

La simulación debe *venderse* con la máxima responsabilidad. Debe huirse de la presión por situar una licencia de un software especializado en una compañía que no disponga de los requisitos esenciales para llevar a término por sí misma un desarrollo completo respetando los criterios que hemos expuesto hasta el momento. En este caso, debe convencerse al cliente de que precisa del asesoramiento de una empresa experimentada en el desarrollo de proyectos de simulación y en otros ámbitos de la investigación operativa y de la estadística. En general, los proyectos de simulación son realizados por grupos externos, lo que puede garantizar experiencia e independencia, con la participación activa, eso sí, de las personas implicadas por parte del cliente. Se comentan a continuación tres casos.

Caso 1: el proyecto no ha satisfecho o convencido al cliente. La simulación no puede ser usada y, en consecuencia, se han malgastado los costes de desarrollo de la simulación. Es desgraciadamente frecuente que el estudio no encaje bien en la estructura de la empresa o no coincida con los deseos del cliente. A menudo el proyecto es encargado por personas distintas a las directamente vinculadas con el proceso de mejora. Los afectados ven las soluciones como una imposición o incluso una frivolidad, pues consideran que los mismos recursos “podrían haberse aplicado mucho mejor a otro tipo de estudio” o a la mejora de la planta o a la adquisición de una nueva máquina, pongamos por caso. La simulación o el simulador se “quedan en un cajón” y nunca son empleadas.

Caso 2: el resultado del proyecto de simulación ha sido aceptado. Se han tomado decisiones erróneas a partir del estudio, lo que ha provocado perjuicios o incluso accidentes a los usuarios de los sistemas mal diseñados. Estamos aquí en un caso mucho más grave, pues al perjuicio económico se le pueden sumar desgracias personales que impliquen, además, una responsabilidad civil o penal. Las cláusulas de determinados contratos contienen prevenciones ante estas desgracias y deben ser tenidas muy en cuenta. En los proyectos vinculados a determinados sectores, como defensa o seguridad nuclear, los requerimientos de fiabilidad suelen ser muy altos y vienen determinados por especificaciones y normativas de acreditación muy estrictas.

© Los autores, 2002; © Edicions UPC, 2002.

Caso 3: se ha generado una controversia innecesaria debido a la incorrección de los resultados de la simulación, retrasando programas o proyectos o impidiendo su terminación. A menudo en la práctica de los proyectos de simulación se ven implicados diferentes actores de una o varias empresas, con culturas empresariales y tecnológicas distintas. Los problemas de comunicación y de interpretación tanto de los datos como del resto de la información, pueden llegar a generar tensiones importantes a lo largo de todo el estudio [JAIN91].

5.2 Diseño de experimentos

5.2.1 Introducción

Existen numerosos textos que tratan extensamente los conceptos que abordaremos hasta el fin del presente capítulo [PRAT97]. El grado de detalle con el que los autores se enfrentan al tema es muy variado y responde a menudo, no tanto al nivel de comprensión que se quiere conferir al conjunto de la información suministrada, como al perfil o áreas de interés de aquellos autores en un aspecto específico de todo el complejo proceso que aquí englobamos en el epígrafe *Diseño de experimentos* (DdE) y *Análisis de resultados en simulación* (AdR). Las referencias [KLEI92], [BRAT87], [LAW99], [BANK99] no dejan de ser, de todos modos, los textos sobre simulación discreta por excelencia sobre estos temas y de obligada consulta, tanto por su calidad como por la experiencia de sus autores.

A menudo el tema viene desmenuzado en distintos y numerosos capítulos [BANK99], con poca articulación entre sí, lo que dificulta la labor de comprensión global del lector, especialmente de quien desee alcanzar un primer nivel de conocimientos que le proporcione una buena base conceptual, pero desprovista de excesivos formalismos matemáticos que le desorienten.

La complejidad de algunos de los textos hace que muchos de los conceptos que allí se describen no atraigan el interés de las personas implicadas en los proyectos de simulación o de toma de decisiones en general, pues se desaniman al no encontrar soluciones a determinados aspectos de los proyectos en los que ya detectan ciertas carencias: desearían introducir instrumentos de mejora en cuanto al rendimiento del estudio y respecto a la confianza que puedan depositar en las conclusiones y resultados obtenidos.

Esta problemática la podemos visualizar a través de tres situaciones muy frecuentes. La primera viene provocada por la limitación de recursos disponibles en cualquier proyecto de ingeniería. A pesar de los esfuerzos de planificación, los proyectos se extienden más allá de lo programado. En el caso de un estudio de simulación, se dedica mucho tiempo y dinero a las fases de discusión previa de los objetivos, de obtención y análisis de los datos del sistema, de diseño del modelo conceptual y especialmente a la de desarrollo e implantación del modelo ejecutable. Además, otros factores absolutamente incontrolados y fuera del ámbito del propio estudio generan más retrasos: discusiones por los costes y la forma de pago, determinación de quién debe firmar el contrato, cláusulas de salvaguarda y de litigio, de confidencialidad, de explotación futura del producto o servicio, tiempo para la selección, adquisición e instalación del *hardware* y del *software* necesarios, contratación de personal especializado, y un largo etcétera.

© Los autores, 2002; © Edicions UPC, 2002.

Una vez llegados a este punto, en un numeroso grupo de proyectos del ámbito de la fabricación, de la logística y de los servicios, ocurre que los objetivos iniciales se han ido perfilando o delimitando cada vez más a lo largo del propio desarrollo del estudio. Las personas implicadas conocen bien lo que tienen entre manos y sólo tienen *in mente* un subconjunto reducido de las alternativas de configuración del sistema, que son las que consideran potencialmente o razonablemente "mejores". Esto viene frecuentemente guiado por la experiencia y, desde luego, no debe ser despreciado sino todo lo contrario. Este mero hecho ya condiciona uno de los temas del presente capítulo: el de la determinación de los distintos escenarios que deben evaluarse de una forma sistemática y completa o diseño de los experimentos. El cliente pretende comparar únicamente dos o tres alternativas que considera "interesantes" o útiles. ¿Para qué, pues, complicarse más la vida?.

El segundo caso afectaría a la confianza que se debe depositar en los resultados obtenidos de la simulación. A nivel elemental ya hemos tratado el tema en la sección previa de este mismo capítulo, en cuanto a los aspectos de validación de los datos, calidad del *software*, acreditación, etc.

Finalmente, una tercera cuestión, no menos importante. Para algunas empresas el desarrollar un proyecto de simulación complejo implica la dedicación de personas muy cualificadas y con alta responsabilidad en los organigramas de las mismas. Al principio del proyecto suelen estar presentes muchos de los expertos clave, pero el largo desarrollo de este tipo de estudios hace que su participación se diluya ante otras responsabilidades más críticas o del día a día. Para un buen fin del proyecto pueden tener la necesidad de contratar a personal especializado, lo que no siempre es posible, pues se precisa de un período de adaptación y formación o representa un coste no previsto o no asumible por el cliente.

El análisis de los resultados es un cuerpo de materia compuesto por varios componentes operativos tácticos que interactúan entre sí, tal como se describe en la figura 5.13. En ella podemos distinguir varios bloques principales:

- La configuración del diseño experimental.
- La consideración del sistema, desde el punto de vista de la simulación, como terminal o no terminal: análisis de la estacionariedad y determinación del punto de fin de transitorio.
- La comparación de alternativas. Intervalos de confianza y pruebas de hipótesis. Análisis de sensibilidad.
- La determinación del tamaño de la muestra: longitud de la simulación, métodos para la recolección de las variables de respuesta. Técnicas para la reducción de la variancia.
- La consideración de escenarios con múltiples respuestas: análisis de resultados multivariantes.
- La incorporación del concepto de optimización en simulación y la construcción de metamodelos.

© Los autores, 2002; © Edicions UPC, 2002.

Algunos conceptos y definiciones previas

En nuestro caso, el diseño de experimentos (DdE) será la aproximación científica cuyo objetivo es decidir cómo se obtiene la información como producto de la evaluación de distintos escenarios de simulación. Desde otro punto de vista el objetivo de un diseño experimental apropiado sería obtener el máximo de información con el número mínimo de experimentos [JAIN91].

Las simulaciones de gran tamaño contienen normalmente un gran número de factores de entrada, por lo que con frecuencia deberemos realizar algún proceso previo de reducción en su cantidad si queremos trabajar con un entorno de experimentación manejable y adecuado a nuestros propósitos. Deberemos seleccionar cuáles de estos factores se mantienen como tales porque los consideramos trascendentales para nuestro trabajo y cuáles pueden simplemente ser fijados a un determinado valor razonable, y trabajar con ellos asignándoles este nivel en el transcurso de todo nuestro análisis. El DdE es la selección de las combinaciones de niveles de factores que serán realmente simulados en un experimento con un modelo de simulación. El programa de simulación se ejecuta para dichas combinaciones y, como conclusión, se analizan los datos de entrada/salida del experimento a fin de derivar de ellos conclusiones sobre la importancia de los factores.

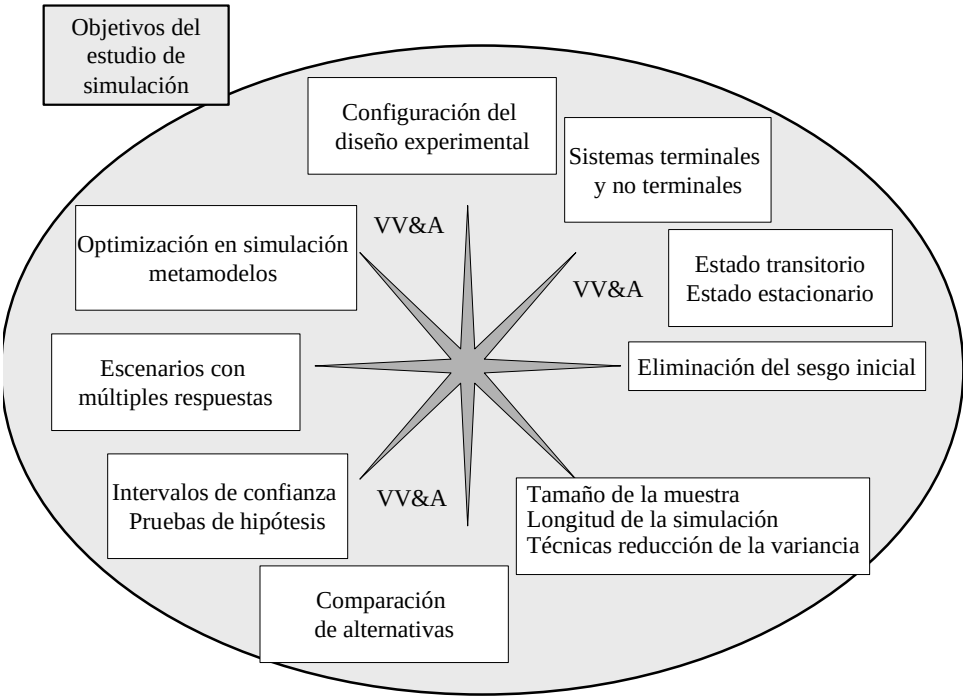


Figura 5.13 Componentes principales del DdE y del AdR

© Los autores, 2002; © Edicions UPC, 2002.

Proporcionamos a continuación un conjunto de definiciones aplicables al universo de la simulación que nos pueden ser útiles para fijar los objetivos del capítulo [KLEI92][JAIN91][PRAT97].[6]

- *Variables de estado*: Aquellas que nos caracterizan el estado de un sistema en un momento dado de la simulación. El número y características de las mismas dependen de los objetivos y del grado de detalle del modelo en cuestión. En todo caso, una variación en el valor de una variable de estado determina un cambio en el estado del sistema y, por lo tanto, el seguimiento de su evolución constituye una secuencia de información durante el proceso de simulación.

  Ejemplos: el número de clientes en espera de ser atendidos en un servicio de urgencias, el estado (ocioso, operativo, averiado, esperando reparación, bloqueado,...) de una máquina, etc.

- *Variable respuesta*: es el resultado de un experimento, con el objetivo, por ejemplo, de constituir una medida de rendimiento del sistema. Una variable de estado puede generar o constituir, por sí misma, el objeto de nuestra medición y ser, por lo tanto, analizada como variable de respuesta.

  Ejemplos: tiempo medio de espera para ser atendidos en el departamento de radiología en un servicio de urgencias de un hospital, porcentaje de utilización de una canal de comunicación, porcentaje del tiempo en el que un dispositivo no está disponible por estar averiado o en reparación, etc.

- *Eficiencia*: relación entre el rendimiento máximo real alcanzable por un sistema y su capacidad nominal.

  Ejemplo: en un sistema de comunicaciones, entenderíamos por eficiencia de un determinado tramo de una red el cociente entre la velocidad de transmisión máxima alcanzada expresada en megabits por segundo y el ancho de banda nominal de aquel enlace.

- *Factores*: denominamos factor a aquella variable que afecta de algún modo a la variable respuesta y que puede presentar diversos valores alternativos. Algunos de ellos pueden considerarse principales y los cuantificaremos. De otros factores, a los que denominaremos secundarios, aunque afecten al rendimiento del sistema, no estaremos interesados en cuantificar su aportación. Identificaremos diversas familias de factores como pueden ser los parámetros, las variables de entrada, los submódulos de simulación y otros.

  Ejemplos de tipos de factores:

  - *Parámetros:* el tiempo medio de servicio en una empaquetadora, el tiempo medio entre llegadas de clientes; el tamaño medio de un lote de piezas; la duración media y la distribución del tiempo de averías y de reparación de una cinta transportadora, etc.
  - *Variables de entrada*: número de servidores disponibles durante cada una de las franjas horarias que constituyen el patrón de un turno de trabajo determinado, número y función de las taquillas de peaje en una autopista.
  - *Submódulos de simulación*: submodelo que caracteriza una determinada política o que gestiona las prioridades (FIFO, LIFO, SPT, etc.) constituyendo por sí mismo una alternativa de diseño programada de forma externa o como componente.
  - *Otros*: factores que caracterizan, como "caja negra" la carga del modelo, por ejemplo, la

[6] Consultar referencias adicionales de diversos autores en [BANK98].

© Los autores, 2002; © Edicions UPC, 2002.

producción de botellas en una planta envasadora o los *palets* vacíos de retorno del área de *picking* de un almacén de distribución. En este caso, la información proviene de un sistema dinámico exógeno o, más propiamente, de su sistema de información y de control de la producción. Nos interesa solamente una aproximación a los factores de carga.

- *Niveles:* Valores que puede alcanzar cada uno de los factores, es decir, alternativas que caracterizarían dichos factores. El nivel puede ser cualitativo o cuantitativo. Por definición, al menos un factor es modificado a lo largo de la experimentación, configurando un número mínimo de alternativas para comparar. Una formulación frecuente es, por ejemplo, que cada factor pueda tener exactamente dos niveles, lo que dará paso a los diseños de tipo 2k, en los que k representa el número de factores que se consideran.

  Ejemplos de niveles: para el tiempo medio de servicio en una empaquetadora: 20 segundos, 30 segundos; para el factor *prioridad* establecida, en el orden de extracción de piezas en un almacén intermedio, se pueden evaluar tres posibles niveles: FIFO, LIFO, aleatorio; para el tráfico de llamadas a un servicio de atención al automovilista: día laboral, día festivo; para el establecimiento de la duración de los bloques de turnos en el procesado de pedidos en una factoría farmacéutica: 4 horas, 8 horas; para el número de taquillas en modalidad telepago en una autopista: 2,3,4; etc.

- *Interacción*: dos factores interaccionan si el efecto de uno de ellos depende del nivel del otro. El análisis de las interacciones es uno de los objetivos principales de un buen diseño experimental.

  Ejemplo: podemos analizar la interacción en un sistema multiprocesador 8x8 entre el tamaño de las memorias del sistema multimemoria y la velocidad de transferencia del bus.

- *Escenarios*: son cada una de las alternativas del modelo que pretendemos evaluar o comparar.

  Ejemplo: en una simulación de tránsito (figura 5.14) un escenario puede quedar definido por la red de viales, modelados en detalle como una colección de nudos y arcos (cruces y carriles) convenientemente prefijados, el plan de control semafórico a lo largo del día, y la carga de tránsito o volumen que circulará por la red por periodos de tiempo (*time-slices*), distribuida por distintas modalidades de transporte (tipos de vehículo y comportamiento de los mismos, por ejemplo).[7]

- *Repeticiones*: réplicas o ejecuciones del modelo computacional que se realizan de cada uno de los escenarios que configuran la base de los experimentos. En cada repetición se trabaja con el mismo escenario base o alternativa de experimentación pero se utilizan distintas series de valores que caracterizan los fenómenos modelizados, basadas normalmente en la generación de valores aleatorios que siguen una determinada distribución teórica o empírica. En este caso, se modifica la semilla inicial del generador de números aleatorios que caracteriza cada una de las variables y se repite la ejecución siguiendo la misma configuración base del escenario.

[7] Para más referencias sobre este tipo de simulación ver los estudios de Barceló sobre tránsito y el entorno especializado de simulación de tránsito Aimsun [BARC00]

© Los autores, 2002; © Edicions UPC, 2002.

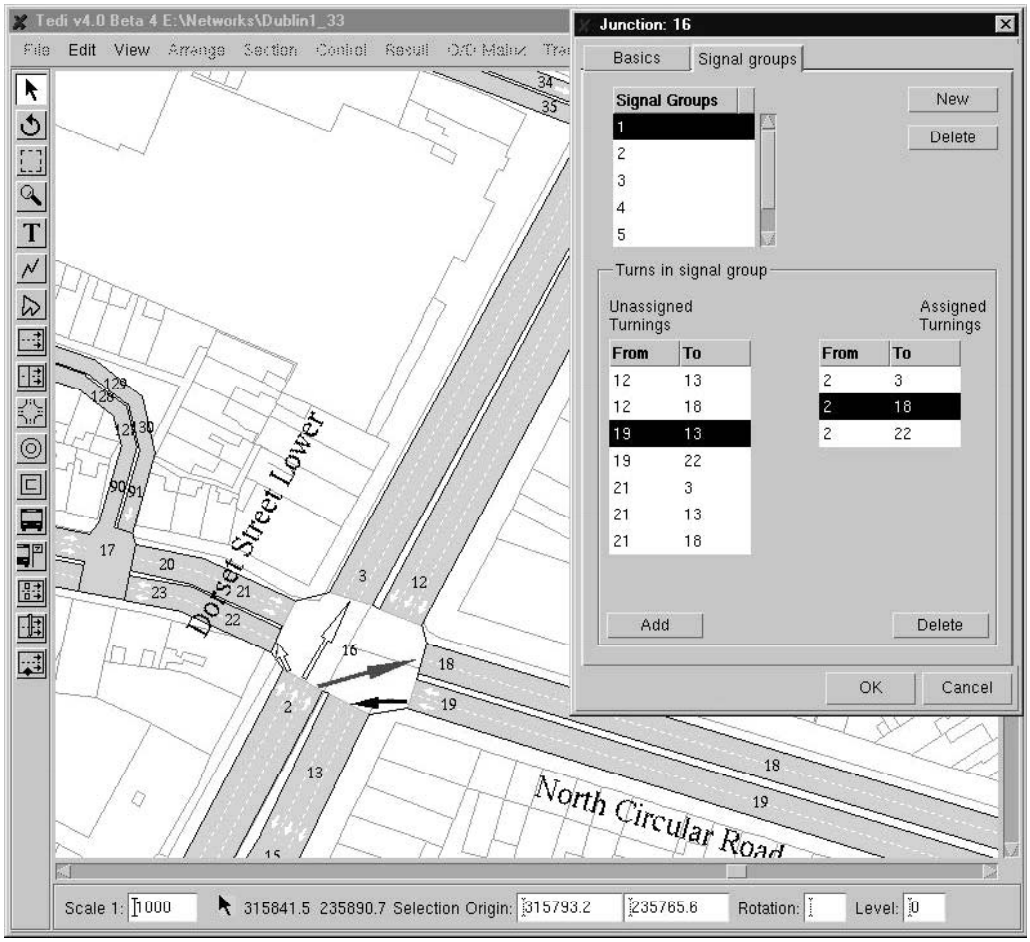


Figura 5.14 Escenario de simulación de tránsito modelado en AIMSUN (TSS)

- *Diseño experimental*: un diseño experimental consiste en la especificación, de forma estructurada y con el objetivo de planificar y proporcionar la información necesaria para un trabajo de investigación, un estudio, o las conclusiones de una consultoría, de:

  - el número de experimentos,
  - la combinación de niveles de factores,
  - el número de repeticiones de cada experimento.

Planificación y diseño del proceso de experimentación

La experimentación en simulación es un proceso que puede llegar a ser muy costoso y consumir grandes cantidades de recursos, tanto en tiempo como en talento de los especialistas necesarios para llevar a cabo el análisis de los resultados de nuestro trabajo. En términos prácticos, además, se produce como culminación de todo un complejo proceso que venimos describiendo a lo largo de este

© Los autores, 2002; © Edicions UPC, 2002.

libro y que, a menudo, consume una gran parte, si no la totalidad, del tiempo y dinero previstos para la globalidad del proyecto. Es, pues, muy importante afinar mucho en la especificación de los procedimientos experimentales durante la fase de diseño del proyecto y, de forma coherente, traducirla adecuadamente a la planificación temporal e incorporar sus costes reales al presupuesto.

Parece conveniente desarrollar metodologías y sistemas de soporte para analistas con poca experiencia en DdE, que les orienten en la planificación y diseño de los experimentos de simulación y les proporcionen herramientas y procedimientos -inteligentes- para realizar correctamente su trabajo. Algunos paquetes estadísticos comerciales ya disponen de utilidades para generar algunos de los diseños de uso más general, por ejemplo SAS, SPSS o SYSP.

En función de los objetivos que persiga el analista o el usuario de un simulador deberán seguirse cierto número de pasos. Coleman y Montgomery [COLE93] definen unas hojas guía para sistematizar la planificación de experimentos y ayudar al equipo de analistas en la definición clara de los objetivos y el alcance de los mismos. Özdemirel [ÖZDE96], desarrolla un entorno de ayuda para problemas de fabricación discreta (*DOES*), basado en un sistema experto y aplicado a un problema de manufactura de teléfonos móviles, que pretende orientar al usuario hacia uno de los diseños predeterminados que almacena en su base de conocimiento en lugar de construir uno partiendo de cero. En la figura 5.15 se relacionan las fases de un proceso de planificación y diseño experimental, las cuales constituyen, por sí mismas, un proceso iterativo en el que la información o las conclusiones obtenidas en una de las fases puede implicar reconsiderar las obtenidas en alguna de las anteriores.

Las herramientas de ayuda pueden estar orientadas a construir diseños clásicos, diseños *óptimos*, o a la optimización experimental secuencial en superficies de respuesta. La elección del enfoque del análisis y diseño del sistema debe ser realizado por todo el equipo de experimentación y de forma cooperativa. Los objetivos han de ser representativos, específicos, medibles y conducentes a resultados prácticos [COLE93]. Se trata de analizar alternativas, decidir políticas de gestión, de ordenación o de producción, identificar problemas y puntos de conflicto, bloqueos, cuellos de botella, zonas compartidas, identificar recursos, etc.

Para finalizar este bloque destinado al planteamiento previo del diseño experimental en simulación, haremos algunos comentarios sobre las exigencias básicas para una buena práctica. En primer lugar, es recomendable realizar un número equilibrado de experimentos. Es perjudicial trabajar en proyectos masivos en los que se consideran muchas posibles combinaciones de factores a distintos niveles. Ello constituye un malbaratamiento de recursos, al requerir demasiados experimentos para obtener información que podría y debería obtenerse de forma más eficiente. Es mejor dividir un proyecto en distintas fases más simples y controlables, con diseños más pequeños, con menos factores y niveles. Todo ello redundará en un mayor control sobre el trabajo, una mayor capacidad de identificación y depuración de errores, de eliminación de información o de componentes superfluas. Adicionalmente, esta estrategia puede proporcionarnos un medio de poner de relieve la necesidad de un cambio en el tipo o cualidades de los datos que nos sirven para fijar los niveles de los factores.

Deben aislarse convenientemente los efectos de los distintos factores, y no trabajar simultáneamente con información de la que no se pueda distinguir o determinar apropiadamente su origen. Todo ello nos puede conducir a asumir como buenas conclusiones erróneas. Por otro lado, tampoco es recomendable la utilización de diseños excesivamente simples como, por ejemplo, variar un factor

© Los autores, 2002; © Edicions UPC, 2002.

cada vez. Un diseño experimental adecuado debe proporcionar intervalos de confianza mejores (más estrechos) con el mismo número de experimentos.

	Elección del enfoque para el análisis y el diseño del proceso
	Identificación y elección de las variables de respuesta
	Selección de los objetivos del diseño experimental
	Información sobre restricciones temporales del proyecto
	Información sobre el número de factores
	Clasificación de los factores
	Definición de los niveles de los factores
	Selección de las opciones de estimación de interacciones
	Determinación de métricas para la evaluación
	Selección de un diseño experimental

Figura 5.15 Fases del proceso de planificación de un diseño experimental, adaptado y simplificado de Özdemirel [ÖZDE96]

En la selección de las variables de respuesta, trabajaremos con diversos tipos de medidas. En el apartado siguiente discutiremos algunas de las más frecuentes. Para cada uno de los objetivos fijados en la fase anterior, deberemos identificar una o más variables de respuesta a observar.

Evaluación del rendimiento de un sistema

El objetivo de un estudio de simulación es, en definitiva, adquirir mayor conocimiento sobre el comportamiento de un sistema gracias a la adecuada exploración de las alternativas de diseño generadas por la combinación de un conjunto restringido de factores. En particular, nos interesará determinar el rendimiento global del sistema o de algunos de sus componentes, a fin de establecer los criterios que nos conducirán a tomar una determinada decisión.

Nos referiremos a diversos tipos de problemas, entre los que destacamos: la identificación de cuellos de botella y su influencia en el rendimiento, la determinación del valor recomendable para un parámetro, la comparación entre dos o más sistemas, la determinación del número y de las características de elementos del sistema, o la predicción del rendimiento del sistema frente a cargas de trabajo crecientes.

EJEMPLO 5.8

En el caso del ejemplo 5.5 y 5.7 (centro automatizado de atención a clientes), nos interesará observar la variación en el tiempo de respuesta ante peticiones de servicio de atención en carretera cuando incorporamos nuevos grupos de clientes potenciales (figura 5.16).

Partimos de un sistema ya operativo en el que estamos atendiendo, según un determinado estándar de calidad, a nuestro conjunto de asociados. Se nos presenta la posibilidad de ofrecer nuestro *know-how* como servicio de valor añadido a clubes de automovilistas de otros países que tienen como destino turístico el mismo ámbito geográfico que la empresa actualmente ya cubre. En este entorno disponemos de una red de asistencia propia importante y tenemos identificados y contratados los

© Los autores, 2002; © Edicions UPC, 2002.

servicios de talleres y grúas en las poblaciones en las que no ofrecemos directamente servicio. Nuestros operadores y gestores de información han alcanzado un grado de eficiencia alto, a partir del esquema de trabajo y dotación de personal que nos había sugerido un estudio precedente de simulación. Se trata ahora de comprobar si nuestros recursos pueden absorber la nueva demanda (fuertemente estacionalizada y con una punta máxima situada en el mes de agosto) cumpliendo con los requerimientos de calidad que nos han permitido alcanzar nuestra posición en el mercado.

Caracterizaremos distintos niveles de carga en función del tipo de día (verano, no verano), del paquete de clientes (nuevo grupo del Automóvil Club Moldavo incorporado, no incorporado) y de las configuraciones de turnos de trabajo confeccionables con nuestro personal actual. Deberemos averiguar si la configuración actual de la centralita telefónica automatizada (ACD) soporta la nueva carga, tanto en número total de líneas disponibles como en la distribución de las mismas por derecho de acceso (entrada, salida, entrada/salida).

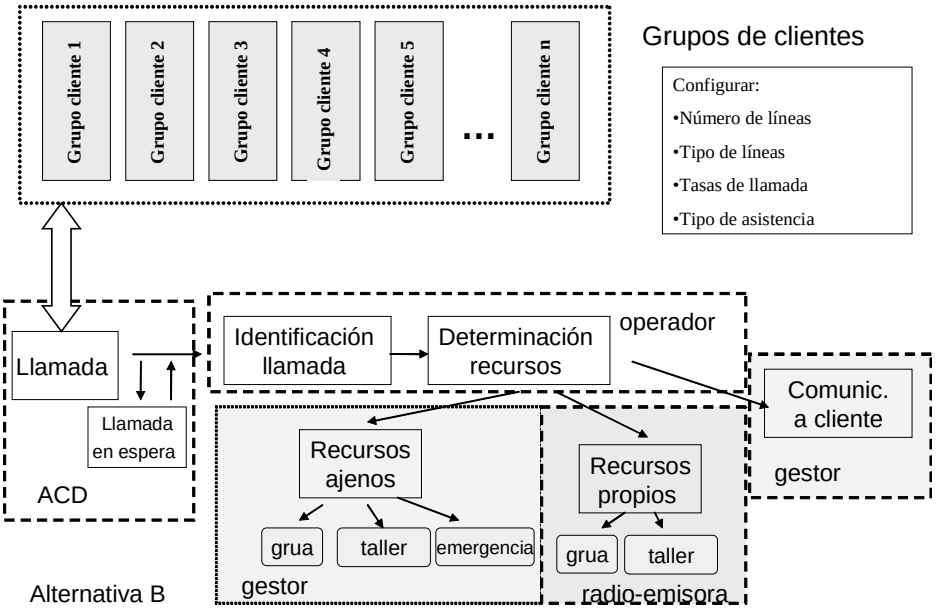


Figura 5.16 Alternativas de grupos de clientes a evaluar

En todo caso, nos interesará determinar las métricas o criterios que deberemos usar para comparar el rendimiento de las correspondientes alternativas de diseño. La utilización de los instrumentos apropiados de análisis estadístico, nos permitirá obtener la máxima información y el mejor resultado con el mínimo de recursos posibles.

Cabe destacar que un mismo sistema puede ser analizado desde diferentes perspectivas, en cuanto que las medidas de rendimiento y su interpretación dependen de la experiencia o de los intereses, de los analistas que realicen el estudio o redacten las conclusiones.

© Los autores, 2002; © Edicions UPC, 2002.

EJEMPLO 5.9

Retomando el ejemplo 5.4, supongamos las necesidades de expedición de la fábrica de electrodomésticos, en la que debemos comparar nuestros dos planes de asignación de tareas al personal del almacén, especialmente en relación con distintas cargas de trabajo. Podemos pensar en condiciones estacionales: en verano se venden muchos frigoríficos, mientras que en invierno se comercializan más microondas. Esto configura volúmenes y pesos distintos. Se trata de comparar si la nueva organización es mejor.

Para ello nos ayudamos de la siguiente tabla, en la que se reflejan el número de *palets*, independientemente del tipo de producto empaquetado, servidos por día.

plan	verano	invierno	promedio
actual	280	135	205,5
nuevo	158	267	212,5

El rendimiento promedio de los dos planes prácticamente coincide. En principio, no tendríamos suficientes elementos de peso como para realizar el cambio estudiado. Supongamos ahora que fijamos el nivel actual de expedición como base, es decir, le asignamos un valor índice 100 y, recíprocamente hacemos lo mismo con la nueva opción. En la tabla siguiente podemos apreciar que en el primer caso si ponemos como base la configuración actual, la nueva organización de tareas es considerablemente preferible (un 27,21%). Todo lo contrario si fijamos al nivel base 100 esta última opción.

	opción base nivel actual			opción base nivel nuevo		
plan	verano	invierno	promedio	verano	invierno	promedio
actual	100	100	100	177,2	50,6	113,9
nuevo	56,42	198	127,21	100	100	100

De este ejemplo se deduce la posibilidad de emplear alternativas de presentación de resultados que, aparentemente, favorecen una de las opciones planteadas, según sea el enfoque, malicioso, del informe final.[8]

Selección de las medidas de rendimiento

En este sentido, al proceder al diseño de un estudio de rendimiento basado en simulación, deberemos seguir los siguientes pasos: *identificar los servicios del sistema*, las *métricas asociadas de rendimiento*, los *parámetros* del sistema y los que caracterizan la carga a que vamos a someter al mismo, los *factores* y sus rangos (o mejor, los *niveles* que se considerarán para cada uno de ellos), y las *alternativas de carga de trabajo* que deseamos comparar. A continuación detallamos algunos de estos aspectos.

Entre otros muchos de los estadísticos que deberemos recolectar, destaca en general el *tiempo de respuesta* o intervalo transcurrido entre una petición de un servicio y el fin de la respuesta del correspondiente elemento o recurso del sistema al que se ha solicitado. En algunos casos conviene

[8] Para una descripción de juegos de presentación de resultados, ver [JAIN91].

© Los autores, 2002; © Edicions UPC, 2002.

distinguir entre el instante de tiempo en el que se inicia o finaliza la petición respecto del tiempo de finalización del servicio, así como puede interesar diferenciar claramente el momento en el que el servidor atiende la petición o tiempo de reacción del sistema. En general, nuestro objetivo será alcanzar tiempos de respuesta lo más bajos posible.

EJEMPLO 5.10

Esto es especialmente cierto en el caso de la simulación de la interacción entre el usuario y un sistema de computación. En este tipo de modelos, el tiempo de reacción del usuario (del orden de segundos) es realmente trascendente en relación con el tiempo de reacción del programa o del sistema operativo (que puede medirse en unidades millones de veces menores, aunque, naturalmente, el retraso puede ser del mismo orden de magnitud si se está compitiendo por un recurso escaso, por ejemplo CPU).

Otro caso similar sería la simulación del funcionamiento de un proceso basado en internet: algunos recursos colapsados, como el ancho de banda disponible, pueden desdibujar el peso de otros, como el tiempo de procesado del encaminamiento de paquetes por la red.

En muchos problemas, nos interesará determinar:

- Los *tiempos* medios, máximos y mínimos de espera frente a cada uno de los servicios, y con más detalle las características de su distribución.
- Las *longitudes* medias y máximas de las colas, y los momentos de la simulación en los que se han producido estos valores máximos. Valores inaceptables de estas medidas de respuesta pueden mostrarnos bien un escenario de configuración inviable para nuestros objetivos de calidad de servicio, bien algún error de implementación de nuestro modelo, entre otras posibles conclusiones.
- En algunos estudios sobre seguridad, por ejemplo, puede interesarnos conocer el *número total* medio de clientes en el sistema, así como identificar los instantes de máxima afluencia o concurrencia de público.
- Paralelamente al tiempo de respuesta, podemos analizar la tasa a la cual las peticiones de servicio son atendidas por el sistema. Aquí nos interesa alcanzar los máximos valores posibles. Ahora bien, cabe destacar que este tipo de rendimiento puede alcanzar un techo, cercano a su capacidad nominal, en el que el tiempo de respuesta crece hasta alcanzar niveles inaceptables. Conviene entonces identificar el punto en el que el sistema ofrece el máximo de su capacidad utilizable manteniendo el nivel de respuesta adecuado, lo que vendría a definirnos el nivel de máxima *eficiencia* del mismo. Este aspecto lo veremos a continuación al tratar de los recursos. En este tipo de situaciones interesa a menudo determinar la interacción entre situaciones de *saturación*: dos recursos no completamente saturados pueden coincidir en el tiempo y generar una situación de congestión no deseada.

En cuanto a las estadísticas vinculadas con los recursos, en general nos interesa la información relacionada con el grado de utilización de un servicio o recurso. Esta relación vendrá expresada en porcentaje del tiempo total de disponibilidad del recurso o variantes del mismo que describimos a continuación:

© Los autores, 2002; © Edicions UPC, 2002.

- Grado o *tasa de utilización* de una máquina, diferenciando el motivo por el que el dispositivo no ha sido utilizado, es decir, caracterizando el *tiempo ocioso*. Así, no es lo mismo que una máquina no esté operando porque no tiene trabajos pendientes, que si esto ocurre porque no puede desprenderse de una pieza ya elaborada porque el robot que debe liberar la máquina de dicha pieza no está disponible. Diremos entonces que nuestro servidor está bloqueado. Un caso particular de esta información se refiere a las averías y lo describimos a continuación.
- *Porcentaje* de tiempo averiado, número de averías y su distribución temporal, tiempo de máquina averiada, tiempo en espera de ser reparada una vez se ha comunicado la avería, tiempo de reparación, número y especialización de los operarios necesarios para la reparación, tiempo de puesta a punto, número de piezas descartadas por averías, etc.
- En muchos casos nos interesará determinar la *fiabilidad* del dispositivo medida en términos de probabilidad de fallo o de tiempo medio entre fallos (TMEF).
- No debe olvidarse la referencia a la *relación entre coste y rendimiento* para comparar convenientemente dos sistemas. El rendimiento puede venir condicionado por la exigencia de un tiempo de respuesta mínimo garantizado [JAIN91].

En todo caso, está claro que, aunque parecería deseable obtener un porcentaje de utilización de un recurso igual o cercano al 100%, la casi saturación de un servicio crea problemas en el rendimiento global de los sistemas. Así, por ejemplo, un sistema computacional en el que la memoria RAM esté prácticamente ocupada por estar infradimensionada, padecerá grandes retrasos en el tiempo de respuesta deberá de paginar constantemente sobre disco a fin de liberar el espacio que precisan los diferentes procesos que compiten por la memoria y el procesador.

Finalmente, comentaremos algunas causas frecuentes de errores en la evaluación del rendimiento, que pueden conducir a conclusiones incorrectas. A menudo se trabaja sin unos objetivos claramente definidos, tal vez por una falta de entendimiento del sistema. Debemos identificar claramente el problema que debemos resolver. Las métricas, las cargas de trabajo y las metodologías dependen de los objetivos. A veces, éstos vienen sesgados por el interés en defender una determinada opción y buscamos las métricas y cargas de trabajo que destacan la bondad de nuestro sistema.

Los analistas con experiencia saben que *no existen modelos de propósito general*, y que *el modelo es el medio y no el fin*. Debemos realizar una aproximación sistemática a la hora de seleccionar parámetros, factores, métricas, caracterizar la carga, etc. Métricas de rendimiento incorrectas, cargas de trabajo no representativas, el desprecio u olvido de factores o de parámetros importantes, no identificar los que realmente tienen impacto en el rendimiento o la selección de una técnica de evaluación equivocada, pueden conducir, simplemente, al desastre.

Un aspecto sobre el que reflexionar a la hora de determinar el modelo experimental final y que arranca de las fases iniciales del ciclo de vida de un proyecto de simulación, es el relacionado con el *nivel de detalle adecuado a nuestros objetivos y recursos*. Alternativas con poco nivel de detalle podrán ayudar mucho a definir estudios que serán tratados más adelante con mayor precisión o minuciosidad. Debemos *partir de datos fiables*. Ignorar errores en datos de entrada automatizados, como pueden ser los provenientes de sensores remotos averiados o fuera de servicio o el tratamiento inadecuado de

© Los autores, 2002; © Edicions UPC, 2002.

datos incompletos, inconsistentes o *outliers*, nos conduce a manipular y utilizar información defectuosa.

A la hora de seleccionar los factores debe darse preferencia a aquellos que podrán ser modificados por el usuario final: no debemos hacer propuestas imposibles, el cliente viene normalmente condicionado por fuertes restricciones en cuanto a inversión, a funcionamiento o a disponibilidad de personal cualificado para implantar las distintas alternativas. No debemos ignorar los aspectos sociales, ser conscientes de cuestiones como la sostenibilidad o el medio ambiente. No podemos omitir asunciones y limitaciones y no trataremos de forzar una solución... ¡en busca de un problema!

Nos interesará realizar un análisis de sensibilidad y trabajar cuidadosamente desde el punto de vista estadístico, documentando adecuadamente nuestro trabajo. Es fundamental la presentación apropiada de resultados y la utilización de elementos gráficos que apoyen nuestro estudio y conclusiones.

Es importante elegir el diseño experimental apropiado: diseños factoriales completos y fraccionarios pueden ser los más adecuados. En la sección siguiente los desarrollamos con mayor detalle.

5.2.2 Diseños clásicos: simples, factoriales completos y factoriales fraccionales

En la mayoría de los estudios de simulación tratamos de comparar alternativas de configuración o escenarios a fin de elegir aquella que más nos convenga en función de los objetivos del proyecto y de los recursos disponibles. A partir de éstos y de la información proveniente de los elementos de entrada al modelo, y de la identificación de los factores relevantes que afectan al rendimiento del sistema, determinamos los niveles a los cuales vamos a fijar dichos factores y procedemos a establecer las combinaciones de los mismos con las que vamos a experimentar. Con esta finalidad, analizamos ahora los tipos de diseños más frecuentes: simples, factoriales completos y factoriales fraccionales [PRAT97].

Diseños simples

En este tipo de diseños trabajamos con una configuración o escenario básico de partida y, a continuación, procedemos a variar un factor cada vez y vemos de qué manera este factor afecta al conjunto del rendimiento del sistema simulado. Aunque sugerente por su simplicidad, está claro que este proceso malgasta recursos, no es estadísticamente eficiente y tampoco identifica correctamente las interrelaciones entre los distintos factores, con lo que necesariamente conducirá a conclusiones erróneas.

Para tres factores (k=3) y dos niveles, denotaremos las combinaciones mediante 1 o –1 para identificar un valor “alto” o “bajo” del nivel en el caso de factores cuantitativos o la *activación* o *desactivación* de un factor en otro tipo de problemas. Obtenemos la siguiente tabla:

© Los autores, 2002; © Edicions UPC, 2002.

Diseño simple			
Combinación	X_1	X_2	X_3
1	-1	-1	-1
2	1	-1	-1
3	-1	1	-1
4	-1	-1	1

Tabla 5.1

No obstante, si el criterio empleado para construir esta tabla se hubiera basado en que cada factor cambiase, uno tras otro, sin retornar el factor anterior a su valor primitivo, obtendríamos la tabla:

Diseño simple			
Combinación	X_1	X_2	X_3
1	-1	-1	-1
2	1	-1	-1
3	1	1	-1
4	1	1	1

Tabla 5.2

Se puede calcular fácilmente el número total de configuraciones que se evaluarán en esta aproximación. El número n de combinaciones que consideraremos, con k como el número de factores, cada uno de ellos con m_i niveles, vendrá dado por la expresión:

$$n = 1 + \sum_{i=1}^{k} (m_i - 1)$$

En definitiva, esta aproximación no es en la gran mayoría de los casos reales muy recomendable, debido al poco rigor de las conclusiones: por ejemplo, no proporciona estimadores no sesgados ni detecta interacciones que se puedan alcanzar a partir de los resultados obtenidos con un diseño que debemos considerar excesivamente simplista.

Diseños factoriales completos

Es esta una aproximación masiva al análisis de las distintas alternativas de configuración del sistema en estudio. Utilizaremos todas las combinaciones posibles de factores a sus distintos niveles. Esto equivale al producto del número de niveles de cada uno de los factores respectivamente.

Como ventajas, tenemos que evaluamos *todas* las posibilidades y, en consecuencia, lo vamos a aprender todo sobre el modelo con el que vamos a trabajar: efectos e interacciones.

Está claro que este planteamiento puede resultar muy costoso desde el punto de vista del volumen del problema y del tiempo necesario para la ejecución de las repeticiones. El número total n de

© Los autores, 2002; © Edicions UPC, 2002.

experimentos a realizar con k factores, en los que cada i-ésimo factor dispusiera de m_i niveles, respectivamente, vendría dado por la expresión:

$$n = \prod_{i=1}^{k} m_i$$

Para reducir este inconveniente, podemos simplificar el diseño disminuyendo el número de factores y/o el número de niveles considerados para cada factor. Tradicionalmente se ha resuelto el tema concentrándose, al menos a un nivel inicial del estudio de simulación, sólo en dos niveles para cada uno de los factores. En este caso hablamos de diseños 2^k, donde 2^k es el número de experimentos que realizaremos con las combinaciones de k factores con dos posibles niveles:

Diseño factorial completo 2^k			
Combinación	X_1	X_2	X_3
1	-1	-1	-1
2	1	-1	-1
3	-1	1	-1
4	1	1	-1
5	-1	-1	1
6	1	-1	1
7	-1	1	1
8	1	1	1

Tabla 5.3

Otra alternativa es sólo trabajar con aquellos factores que nos sean realmente relevantes y lograr la reducción del número total de los mismos. Se puede dar el caso de que podamos entonces aumentar el número de niveles que consideraremos para esta familia reducida de factores y enriquecer, en cierto modo, la granularidad de nuestro experimento.

Diseño factorial 2^k incl. interacciones						
Combinación	X_1	X_2	X_3	$X_1 X_2$	$X_1 X_3$	$X_2 X_3$
1	-1	-1	-1	1	1	1
2	1	-1	-1	-1	-1	1
3	-1	1	-1	-1	1	-1
4	1	1	-1	1	-1	-1
5	-1	-1	1	1	-1	-1
6	1	-1	1	-1	1	-1
7	-1	1	1	-1	-1	1
8	1	1	1	1	1	1

Tabla 5.4

© Los autores, 2002; © Edicions UPC, 2002.

En la tabla 5.4 se han incorporado a las combinaciones de la tabla anterior las interacciones entre dos factores. A continuación, mediante un ejemplo, se introduce el procedimiento de análisis de los efectos entre factores. Los signos de los componentes R_i se configuran a partir de las tablas utilizando la combinación de signos que de ellas se desprenden.

EJEMPLO 5.11

Definimos como *efecto principal* del factor *j* la media del cambio en la respuesta estudiada del sistema debido al cambio del factor *j* de su valor "-1" a su valor "+1" mientras se mantienen invariantes los otros factores del diseño. Para el diseño 2^3 factorial descrito en las tablas 5.3 y 5.4, la respuesta R_i del diseño i el efecto principal del factor 1 es:

$$e_1 = \frac{(R_2 - R_1) + (R_4 - R_3) + (R_6 - R_5) + (R_8 - R_7)}{4}$$

Para el factor 2:

$$e_2 = \frac{(R_3 - R_1) + (R_4 - R_2) + (R_7 - R_5) + (R_8 - R_6)}{4}$$

Para el factor 3:

$$e_3 = \frac{(R_5 - R_1) + (R_6 - R_2) + (R_7 - R_3) + (R_8 - R_4)}{4}$$

Se puede dar el caso que la respuesta de un factor dependa de otro, o sea que los factores interaccionen. Podemos medir el efecto de esta interacción mediante el análisis del *efecto de interacción entre dos factores.* Definimos esta medida como la mitad de la diferencia entre el efecto medio del factor *i* cuando el factor *j* está a su nivel "+1" y los otros permanecen invariantes y el efecto medio del factor *i* cuando el factor *j* está a su nivel "-1". Para el mismo ejemplo obtenemos:

$$e_{12} = \frac{1}{2}\left[\frac{(R_4 - R_3) + (R_8 - R_7)}{2} - \frac{(R_2 - R_1) + (R_6 - R_5)}{2}\right]$$

$$e_{13} = \frac{1}{2}\left[\frac{(R_6 - R_5) + (R_8 - R_7)}{2} - \frac{(R_2 - R_1) + (R_4 - R_3)}{2}\right]$$

$$e_{23} = \frac{1}{2}\left[\frac{(R_7 - R_5) + (R_8 - R_6)}{2} - \frac{(R_3 - R_1) + (R_4 - R_2)}{2}\right]$$

En determinados casos puede interesar la interacción entre más de dos factores. En [PRAT97] puede encontrarse una presentación exhaustiva y clara de la metodología a emplear.

© Los autores, 2002; © Edicions UPC, 2002.

Diseños factoriales fraccionales

A partir del problema detectado con el uso de la alternativa anterior en cuanto al volumen necesario de recursos en un planteamiento "completo", nos podemos ver obligados a seguir una estrategia que, aunque no resultará tan exhaustiva como la anterior, pretende trabajar con sólo una fracción del diseño completo.

Ello puede implicar pérdida de información sobre interacción entre factores. De todos modos suele ocurrir que algunas de dichas interacciones sean de poca relevancia en referencia al estudio que deseamos desarrollar. En consecuencia, un buen plan experimental basado en la consideración de una parte o fracción del diseño factorial completo puede representar un importante ahorro en cuanto a recursos: tiempo y coste del estudio.

También nos puede ocurrir que no haya interacciones entre los factores. En este caso pasaría que utilizaríamos 2^k observaciones para estimar sólo k+1 efectos. Podemos trabajar entonces con una fracción de aquéllas, pongamos por caso 2^{k-p} [KLEI92].

En la tabla 5.5 podemos ver un diseño factorial fraccional 2^{k-p} para k=3 y p=1. Nos queda un diseño con sólo 4 combinaciones:

Diseño factorial fraccional $2^{k-p} = 2^{3-1}$			
combinación	X_1	X_2	X_3
1	-1	-1	1
2	1	-1	-1
3	-1	1	-1
4	1	1	1

Tabla 5.5

Los diseños factoriales fraccionales interesan cuando hay una gran cantidad de factores y tenemos limitaciones en nuestros recursos. Aunque sean parciales, pueden sernos suficientes considerando que una gran parte de los sistemas que nos ocupan suelen estar influenciados primariamente por unos pocos factores y algunas interacciones de orden inferior [MONT91][PRAT97].

La combinación de dos o más diseños factoriales parciales nos pueden generar tanta información sobre la interacción y la influencia de los factores como diseños de mayor dimensión [MONT91].

5.2.3 Criba de factores

La visualización de factores (*Factor Screening*) aparece como una solución para los casos en los que disponemos de un gran número de factores. En estadística no se ha dedicado mucho esfuerzo a esta línea al no ser posible trabajar con tantos factores como los que habitualmente son considerados en un modelo de simulación: los analistas no están, pues, familiarizados, en general, con dichos métodos [KLEI92].

© Los autores, 2002; © Edicions UPC, 2002.

Bifurcación secuencial y group screening

El *screening* por grupos se basa en el principio de agregación. No todos los factores tienen la misma importancia en relación con sus efectos en las respuestas de los sistemas simulados [FRIE96]. Vamos a aislar los factores importantes de aquellos de los que podemos prescindir. Agrupamos los factores individuales en *clusters* o "factores de grupo". Un grupo tiene el valor 1 si todos sus componentes tienen el valor 1. Igualmente valdrá –1 en el caso recíproco. Para cada grupo comprobamos la significancia del efecto estimado del mismo. Si dicho efecto no es significativo, eliminamos todos los factores del grupo para la siguiente fase ya que se entiende que si el grupo no influye tampoco lo harán sus componentes individuales, con lo que quedamos libres de reducir considerablemente el volumen y coste de nuestra experimentación en una fase inicial del proyecto.

Una variante introducida por los autores citados [BETT97] es conocida como bifurcación secuencial (BS). El analista debe conocer a priori si un factor determinado tiene un efecto positivo o negativo en la respuesta. Si para algunos de estos factores no conocemos dichos signos, podemos estudiarlos fuera del diseño de BS. La secuencialización se produce al generar las combinaciones de factores a medida que efectuamos ejecuciones de la simulación que nos permiten acumular conocimiento sobre el efecto de los factores y así eliminar grupos de factores (*clusters* que no contienen factores importantes).

Los grupos disminuyen de tamaño. Cada grupo que parezca que incluya uno o más factores relevantes es dividido en dos subgrupos del mismo tamaño (lo que denominamos bifurcación). Finalmente llegamos a investigar factores individuales.

Otros autores han combinado la metodología de Taguchi con una aproximación *group-screening* a fin de obtener diseños eficientes que trabajen con un menor número de factores [MADU92].

5.3 Análisis de resultados

En este apartado desarrollaremos los conceptos que nos ayudarán a diseñar las ejecuciones de los modelos de simulación y a interpretar convenientemente los resultados obtenidos. Bajo este epígrafe general, se agrupan temas muy variados, como son el estudio de los sistemas en función de si están en el estacionario, las técnicas para la reducción de la variancia, el análisis de sensibilidad, los métodos de superficie de respuesta, los metamodelos y la optimización en simulación.

Distinguimos dos tipos de simulación con respecto al tipo de sistema que vamos a estudiar, en función de determinadas características intrínsecas a su comportamiento dinámico [CENT98]. En concreto diferenciamos entre *sistemas terminales* o de horizonte finito y *sistemas no terminales* (figura 5.17), entre los que son de especial interés aquellos que se caracterizan por estar en el *estado estacionario*.[9]

[9] Una visión muy bien estructurada de Alexopoulos y de Seila puede obtenerse en [BANK98] y en [ALEX00].

© Los autores, 2002; © Edicions UPC, 2002.

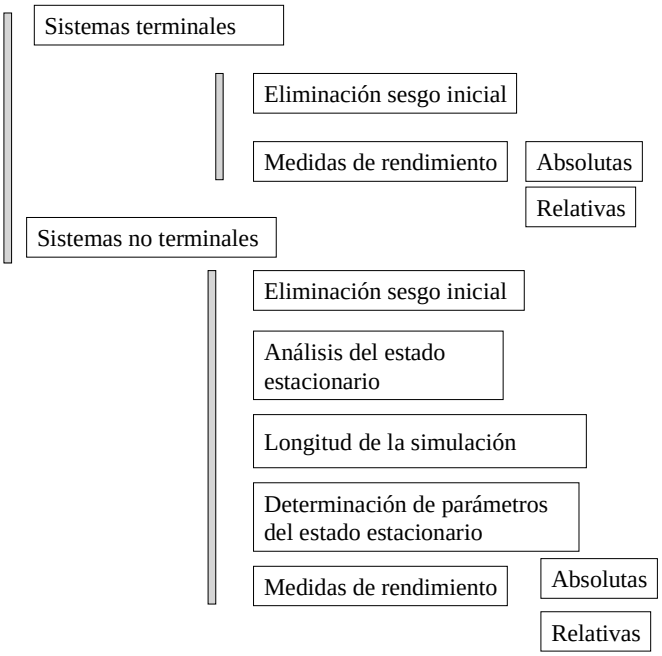


Figura 5.17 Tipos de sistemas desde el punto de vista del AdR en simulación

5.3.1 Simulaciones terminales

Simulación de un sistema con horizonte final o *terminal* (o también en el transitorio, o en el período de carga): es aquella simulación que se inicia en un determinado estado previsto por el diseñador y que va a ser ejecutada hasta que ocurra un determinado evento -previamente identificado- mediante el cual vamos a detener la simulación y, por consiguiente, la observación del sistema o experimentación.

Las *condiciones iniciales* pueden llegar a ser determinantes. Pueden venir caracterizadas por el sistema inicialmente vacío: por ejemplo, un banco por la mañana antes de la entrada del primer cliente, o un servicio de cambio rápido de tubos de escape de automóviles y motocicletas el lunes por la mañana, o por un estado predeterminado: por ejemplo, cuando todos los mecánicos están ocupados o ha pasado una hora de simulación.

El *evento de terminación* puede ser la hora de cierre del establecimiento (momento en que los clientes son invitados a abandonar el recinto), el final de una acontecimiento cultural o deportivo, un determinado horizonte de simulación, por ejemplo, 5 semanas, o el vaciado de los servicios pendientes de atención en una entidad bancaria.

El interés por este tipo de sistemas es múltiple. Muchos problemas corresponden a procesos de puesta en marcha de una instalación o tienen sentido en la fase transitoria de la evolución del sistema. Por otro lado, en muchos casos el sistema simplemente no alcanza nunca un estado que podamos identificar con la estacionariedad. Se trata de situaciones en las que se cumplen condiciones de finalización naturales. Es el caso, por ejemplo, de que se haya finalizado una determinada tarea:

© Los autores, 2002; © Edicions UPC, 2002.

finalización de la construcción de un navío o de una locomotora, cierre diario de un cine o teatro, relajación de la congestión de tráfico al final de la jornada, proceso de revisión y de carga de combustible en una central nuclear (para lo que se requiere de una gran cantidad de personal temporal especializado), etc.

Para el análisis de los resultados en un sistema terminal nos centraremos en los aspectos siguientes:

- Dado que la longitud de la ejecución de la simulación viene determinada por el evento de finalización, no deberemos preocuparnos sobre este aspecto (en oposición de lo que ocurre en las simulaciones de procesos en estado estacionario en las que éste es un problema realmente importante).
- Podemos tratar cada una de las repeticiones del escenario a simular de forma independiente entre sí, considerando las respuestas obtenidas como observaciones independientes, y aplicar métodos estadísticos estándar.
- Debemos tener en consideración el efecto de las condiciones iniciales de la simulación. Algunos métodos proponen la eliminación de los datos iniciales pues suelen introducir sesgo en los resultados del estudio. Surge aquí la dificultad de identificación del punto a partir del cual procedemos a recolectar los datos para elaborar los estadísticos de la ejecución.

En general, en una simulación estocástica estudiamos determinadas variables de respuesta o de rendimiento y presentamos nuestros resultados en términos de su intervalo de confianza (IC). Afirmaremos la bondad de nuestra medición en términos de la cobertura y la dimensión relativa del IC (precisión relativa).

El número de repeticiones necesarias para alcanzar cierta “garantía” de calidad a la hora de extraer conclusiones puede venir determinado de dos maneras [BRAT87]:

- Especificamos una determinada *precisión* deseada (en general, en términos relativos) de nuestro IC y realizamos tantas repeticiones como sean necesarias hasta alcanzarla. Esta aproximación tiene el problema de que, *a priori*, no sabremos cuantas réplicas vamos a necesitar.
- Especificamos un número de *repeticiones* y comprobamos que las observaciones obtenidas nos permiten aceptar la bondad de las conclusiones emitidas. Naturalmente, esto puede obligarnos a prolongar el experimento hasta lograr la precisión deseada.

Los análisis se expresan en forma de *intervalos de confianza* (o regiones de confianza en el caso de análisis multivariante), preferentemente a la utilización de los tests de hipótesis, que parecen más incómodos. Los métodos gráficos (histogramas, *dot plots*, etc.) resultan muy útiles para mostrar de forma directa algunos patrones observables solamente de forma visual.

En conclusión, debemos recolectar las observaciones suficientes como para poder aplicar el teorema del límite central (asumiremos la normalidad de los datos), así como trataremos de preservar unas condiciones mínimas de independencia entre las muestras y trataremos, como ya hemos manifestado, de eliminar el efecto de las condiciones iniciales.

© Los autores, 2002; © Edicions UPC, 2002.

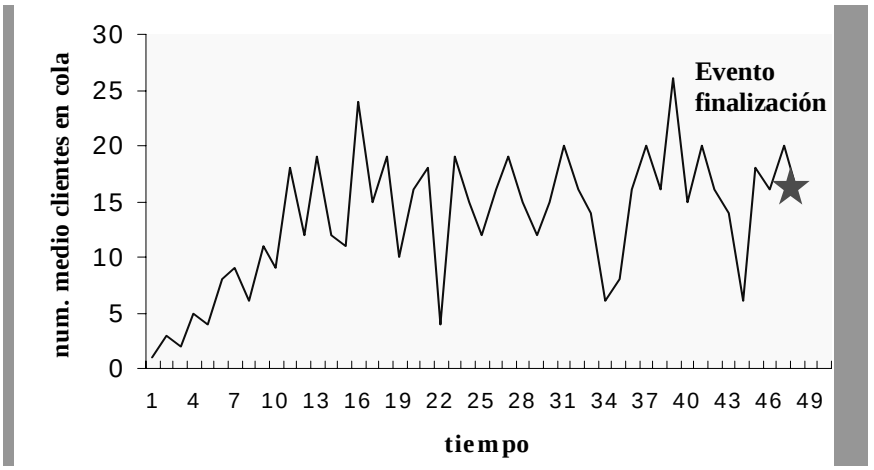


Figura 5.18 Simulación de un sistema con evento de finalización (sistema terminal)

5.3.2 Simulación en el estado estacionario

En principio asociaremos este tipo de sistemas con los estudios sobre el comportamiento a medio-largo plazo de un sistema. Trabajaremos con determinadas medidas de rendimiento como los parámetros en el estado estacionario (que distinguirán precisamente a este tipo de sistema del que correspondería a una simulación terminal). Por ejemplo, podemos estar determinando el tamaño medio de un *stock* intermedio en una zona de almacenaje de una fábrica de componentes para el sector de la automoción, que trabaja continuamente las 24 horas del día durante todo el año. Otro ejemplo sería la determinación del tamaño medio de los mensajes transmitidos entre los nudos de una red de área local o bien el tamaño medio de un paquete de comunicación en un sistema de banda ancha.

En cualquiera de los estados en el que se encuentre nuestro sistema en estudio, debemos determinar el número de repeticiones k, la longitud l de la simulación (lo que nos proporcionará un número determinado de observaciones) y, finalmente, el tamaño de la muestra obtenida n, en el caso de que ésta sea distinta de k (lo que ocurre, por ejemplo, cuando realizamos una *única* ejecución muy larga y de las observaciones recolectadas a lo largo de la misma obtenemos los n valores de la muestra).

5.3.3 Longitud de las ejecuciones de la simulación

En función del tipo de sistema (terminal o de estado estacionario) elegiremos los métodos más apropiados para generar las muestras que precisamos en nuestro experimento. Describimos a continuación tres de ellos: método de las repeticiones, método regenerativo y método de las medias de lotes (*batch means*). Debemos decidir si queremos medidas en ejecuciones "largas" (apropiadas para el estado estacionario) o realizar repeticiones independientes (frecuentemente utilizadas para sistemas terminales).

En cualquier caso se trata de hacer varias repeticiones (reales o dividiendo una ejecución "suficientemente larga" en lotes o ciclos de regeneración) de nuestro experimento de forma que se genere una muestra independiente a la que se puedan aplicar con suficientes garantías los métodos de

© Los autores, 2002; © Edicions UPC, 2002.

la estadística clásica. En referencia a las simulaciones en el estado estacionario, el análisis es mucho más complejo y las ejecuciones de la simulación acostumbran a ser relativamente largas.

Método de las repeticiones

Es éste el método más simple e intuitivo: se basa en nuestra capacidad de utilización de diferentes series de números aleatorios a partir de distintas semillas (según se explicó en el capítulo 3 en referencia a los generadores de números aleatorios). Estas series permitirán generar los correspondientes valores de las variables aleatorias empleadas, usando, por ejemplo, el método de la transformada inversa. Consistirá en un número específico de ejecuciones del modelo con un inicio y un final predeterminados. Comentamos a continuación su aplicación en los casos de sistemas que no alcanzan el estacionario o, también, en proceso de carga y en el caso en que llegan a tener un comportamiento estacionario.

Simulación terminal: vamos a realizar diferentes repeticiones (o réplicas) que puedan ser consideradas como puntos de una muestra sobre la que se aplican los métodos estadísticos clásicos, ya que consideramos que las réplicas son independientes entre sí y están idénticamente distribuidas. En este caso, van a tener mucha importancia los valores iniciales de las variables del sistema a fin de evitar el sesgo no deseado que introducen dichas condiciones de partida.

Sean $\{X_1,\ldots,X_n\}$ *n* datos de salida recolectados con el objetivo de estimar $\mu=E\left(\overline{X}_n\right)$ con

$$\overline{X}_n=\frac{1}{n}\sum_{i=1}^{n}X_i$$

que es un estimador no sesgado de μ. En general,

$$S^2(n)/n=\sum_{i=1}^{n}\left(X_i-\overline{X}_n\right)^2/\left[n(n-1)\right]$$

es un estimador sesgado de $Var\left(\overline{X}_n\right)$

Para solventar este problema, se suelen ejecutar *k* réplicas independientes del sistema de simulación. Si para el conjunto de n observaciones en el conjunto de ejecuciones para la i-ésima observación obtenemos los datos de salida X_{i1} X_{i2} ,..., X_{in}, y podemos calcular las variables aleatorias IID

$$Y_i=\frac{1}{n}\sum_{j=1}^{n}X_{ij}$$

Tenemos que

$$\overline{Y}_k=\frac{1}{k}\sum_{i=1}^{k}Y_i$$

es también un estimador no sesgado de μ, y $\hat{V}_R=\frac{1}{k-1}\sum_{j=1}^{k}\left(Y_i-\overline{Y}_k\right)^2$

© Los autores, 2002; © Edicions UPC, 2002.

es un estimador no sesgado de la varianza. Si además n y k son suficientemente grandes, una aproximación del intervalo de confianza para μ es

$$\overline{Y}_k \pm t_{k-1,1-\alpha/2}\sqrt{\hat{V}_R / k}$$

donde $t_{k-1,1-\alpha/2}$ representa el cuantil 1-α/2 de la distribución t con k-1grados de libertad.

Construiremos un intervalo de confianza 1-α para μ con el error absoluto $\left|\hat{Y}_k - \mu\right| \leq \beta$

o podemos obtener un estimador con un error relativo $\gamma = \left|\hat{Y}_k - \mu\right| / \left|\mu\right|$

A menudo, se considera suficiente que el número de repeticiones sea $k_0 \geq 10$ y que $\gamma \leq 0.15$.

En [ALEX00] y [LAW99] se examinan procedimientos secuenciales para determinar el número de réplicas necesarias para estimar μ con una determinada precisión absoluta o relativa. En [CENT98] se describe un método secuencial en el que, a partir de una primera serie de n réplicas de las que obtengo un semi-intervalo de confianza g y una precisión relativa deseada γ, el número total de repeticiones a realizar viene expresado por

$$n^* = \left\lfloor n \times \left(\frac{g/2}{\gamma} \right)^2 \right\rfloor$$

Simulación del estado estacionario: En primer lugar debe constatarse que podemos utilizar un método similar al de las simulaciones terminales: se considera que ya se ha alcanzado el estado estacionario, se trabaja con las observaciones independientes. Atención también aquí a las condiciones iniciales, por ejemplo, cuando muchos de los valores son inicialmente puestos a cero, ya que ello puede afectar al resultado de la simulación durante un periodo apreciable de tiempo. Podemos amortiguar éste y otros efectos del transitorio, no teniendo en consideración las observaciones iniciales que correspondan al periodo "de carga" (*truncación de datos*): el trazado de las observaciones representadas en un gráfico puede ayudar a determinar estos puntos, que marcan un cierto tipo de frontera entre los dos tipos de estado allí dónde nos parece que el sistema "se estabiliza". También podemos intentar obtener condiciones de partida que nos ubiquen de forma rápida en una situación más representativa del estado "normal" del sistema.

En general, se considera poco recomendable utilizar el método de repeticiones independientes en el caso de sistemas en el estado estacionario, principalmente debido al efecto del sesgo inicial de cada una de las réplicas. Es preferible un método como el de *batch means* en el que dicho sesgo "se pierde" a lo largo de una extensa ejecución de la simulación. El factor a tener en cuenta en este último caso es el de la dependencia entre las observaciones subyacente en este tipo de planteamientos: el dimensionamiento y distribución de los lotes será determinante para obviar este problema.

© Los autores, 2002; © Edicions UPC, 2002.

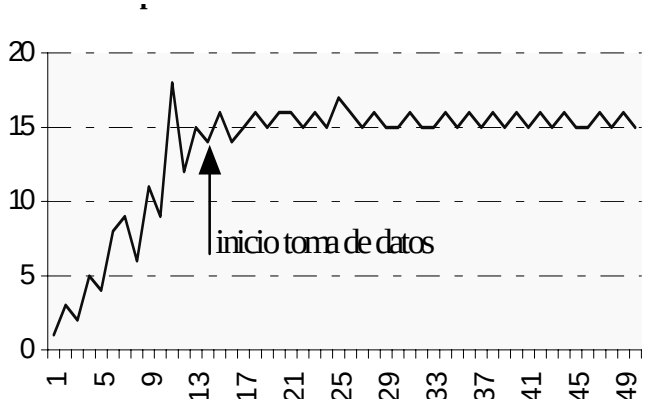


Figura 5.19 Determinación del período de calentamiento (warm-up)

Robinson describe una técnica heurística para determinar la longitud de una ejecución para simulaciones no terminales en estado estacionario. Primero determina el periodo de *warm-up* detectado por el método de Welch. Con esta consideración realiza tres ejecuciones largas en las que recolecta la medida de respuesta de interés, por ejemplo el tiempo de espera. A partir de la observación de las series temporales así generadas y considerando que los resultados de las repeticiones convergen en la medida que aumenta la longitud de la ejecución, determina la longitud de la simulación al fijar el punto en el que las tres series convergen suficientemente como para proporcionar al modelo un nivel de precisión suficiente [ROBI95].

Inicialización: características y posible sesgo. Deberemos proceder a la eliminación por truncación de los datos u observaciones recogidos hasta un determinado punto, en el que consideramos que el proceso llega a un régimen estacionario. Desarrollamos entonces métodos de detección del punto de corte, como el procedimiento gráfico de Welch (1983), analizando la ventana temporal y las medias móviles.

El procedimiento gráfico de Welch es simple y útil. A partir de las observaciones X_{i1} X_{i2} ,..., X_{in} obtenidas para cada una de k repeticiones independientes computa los promedios

$$\overline{X}_j = \frac{1}{k}\sum_{i=1}^{k} X_{ij}$$

Entonces para una ventana de tiempo w, el procedimiento traza las medias móviles

$$\overline{X}_j(w) = \begin{cases} \dfrac{1}{2w+1}\displaystyle\sum_{m=-w}^{w} \overline{X}_{j+m} & w+1 \le j \le n-w \\ \dfrac{1}{2j-1}\displaystyle\sum_{m=-j+1}^{j-1} \overline{X}_{j+m} & 1 \le j \le w \end{cases}$$

Una vez identificado el punto inicial l por este método, si pensáramos usar repeticiones múltiples, se descartarían n las primeras l observaciones de cada ejecución. Calculamos

© Los autores, 2002; © Edicions UPC, 2002.

$$Y_i(l,n) = \frac{1}{n-l} \sum_{j=l+1}^{n} X_{ij}$$

que son IID, por lo que obtendremos ahora el punto estimado

$$\hat{Y}_k(l,n) = \frac{1}{k} \sum_{i=1}^{k} Y_i(l,n)$$

Método regenerativo

Este método se basa en la explotación de algunas propiedades de simulaciones en las que, en cierto modo, determinadas variables retornan a un estado de tal manera que podemos interpretar que en estos puntos consideramos que se produce un fenómeno de reiniciación (y, en cierto modo, de reinicialización). Obtenemos así algo parecido a ciclos independientes (*ciclos de regeneración*) para ser tratados estadísticamente [CRAN74][CRAN75][ALEX00]. No siempre es posible aplicar este método pues, en muchos casos, se detectan muy pocos puntos de regeneración a lo largo de nuestra simulación y, por lo tanto, la muestra obtenida será excesivamente pequeña. Puede incluso ocurrir que no encontremos ningún instante en el que se cumplan nuestras condiciones de regeneración. En este caso, evidentemente este método no es de aplicación.

EJEMPLO 5.12

Supongamos que el sistema que queremos simular es un supermercado. Podemos considerar que delimitamos un ciclo regenerativo cada vez que el local se queda sin clientes o que el número de éstos queda por debajo de un determinado umbral o reproducimos las condiciones de inicio de la simulación.

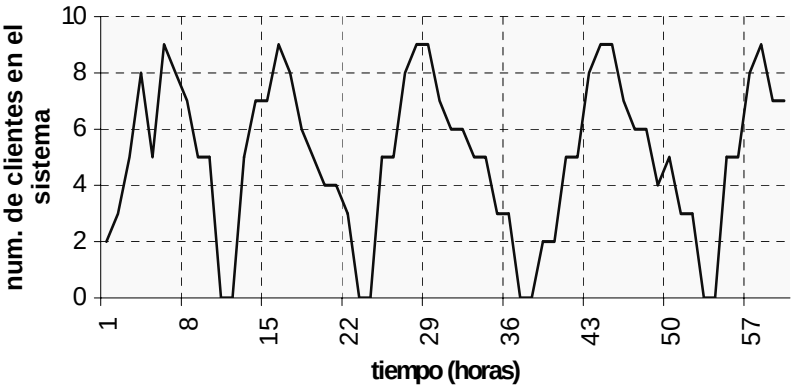


Figura 5.20 Ciclos de regeneración

© Los autores, 2002; © Edicions UPC, 2002.

*Método de las medias de lotes (*Batch means*)*

Para evitar algunos de los problemas antes citados, procederemos a realizar una ejecución de la simulación suficientemente larga y, tras dividir ésta en *lotes o bloques de observaciones*, trabajaremos con los valores promedio de dichos lotes, considerándolos como observaciones independientes y no sesgadas de lo que ocurre en el estacionario.

El método permite en gran manera menguar el efecto de las condiciones iniciales. Es evidente, no obstante, que las observaciones mantienen un cierto grado de dependencia entre sí. Debemos construir intervalos o lotes de tamaño "suficiente". Existen múltiples métodos para determinar dicho tamaño:

- Métodos de las medias de los lotes con estimación consistente (aproximación débil y fuerte (AWA y ASA)).
- Reglas para lotes.
- Método de las medias de lotes solapados (OBM) (como casos particulares de una estimación espectral de frecuencia 0,...).
- Método de las series temporales estandarizadas.

Modelos de series temporales

Podemos considerar las series correlacionadas y no-estacionarias resultado de la simulación como si fueran series temporales, aplicar el correspondiente modelo (AR, ARMA) para ajustar los datos y aplicar dicho modelo ajustado para la realización de inferencia. En el caso del método de las *series temporales estandarizadas*, se aplica una versión proceso del teorema del límite central para estandarizar las series resultado y aplicar métodos de análisis estadístico para su correspondiente procesado.

5.3.4 Determinación de los intervalos de confianza

Nuestro objetivo básico será aquí dar soporte a la evaluación de alternativas, discutiendo cómo utilizar los datos resultado para comparar dos o más sistemas.

La idea básica es que no se puede afirmar taxativamente nada sobre las características de todos los sistemas, pero sí podemos dar una valoración probabilística sobre el rango en el que se moverían las características de la mayoría de sistemas [JAIN91]. Debemos introducir el concepto de intervalo de confianza (IC) como instrumento básico en la evaluación de sistemas.

Partiremos de las siguientes definiciones y notación:

- *población*: es el universo objeto de nuestro análisis
- *muestra*: $\{X_1,\ldots,X_n\}$, n observaciones extraídas de una población.
- μ: media de la población
- σ: desviación estándar o típica de la población
- $\overline{X}$: media de la muestra $\{X_1,\ldots,X_n\}$
- s : desviación típica de la muestra

© Los autores, 2002; © Edicions UPC, 2002.

Así, tendremos como características de la población, los parámetros (ejemplo μ, σ), y como estimadores de la muestra los estadísticos (ejemplo $\overline{X}$, s). Los parámetros son fijos mientras que los estadísticos son variables aleatorias.

Podemos así determinar o dibujar la distribución de un estadístico, por ejemplo la distribución de las medias de distintas muestras extraídas de una misma población. (Ésta tendrá una media μ, única aunque en general desconocida, y una desviación σ, mientras que de cada muestra obtendremos una media propia. En principio, las medias así obtenidas serán distintas entre sí.)

IC para la media

Cada media de una muestra $\{X_1,\ldots,X_n\}$ es una estimación de la media de la población μ. A partir de k estimaciones 'distintas' provenientes de k muestras, queremos obtener una estimación única de μ. Al no poder alcanzar la certeza absoluta o estimación perfecta a partir de un número finito de muestras finitas, intentaremos obtener cotas probabilísticas de la misma. En consecuencia, nos interesará determinar una cota inferior c_i y una cota superior c_s tal que, con una probabilidad "alta" $1-\alpha$, la media de la población esté en el intervalo entre estos dos límites (c_i , c_s), es decir:

$$\text{Prob}\{\ c_i \le \mu \le c_s\ \} = 1-\alpha$$

donde:

(c_i , c_s) = intervalo de confianza de la media de la población

α = nivel de significación

$(1-\alpha)$ = coeficiente de confianza

$100\ (1-\alpha)$ = nivel de confianza. (expresado normalmente como porcentaje)

Por el teorema del límite central, si las observaciones de una muestra $\{X_1,\ldots,X_n\}$ son independientes y provienen de una misma población de media μ y desviación σ, la media de la muestra para tamaños de muestra grandes se distribuye aproximadamente según una normal $N(\mu,\sigma/\sqrt{n})$.

Denominamos *error estándar* a la desviación estándar de la media de la muestra. Sea σ la desviación estándar de la población, tenemos que el error estándar será $\sigma/\sqrt{n}$, luego éste decrecerá siempre que aumente el tamaño n de la muestra. Podemos construir un IC para la media de la población a un nivel de confianza $100(1-\alpha)$

$$\left(\overline{X} - z_{(1-\alpha/2)}\frac{s}{\sqrt{n}}, \overline{X} + z_{(1-\alpha/2)}\frac{s}{\sqrt{n}}\right)$$

donde $z_{1-\alpha/2}$ es el cuantil-$(1-\alpha/2)$ para una normal N(0,1), valor consultable en las tablas.

Para tamaños de muestra menores de 30, sólo podemos construir IC si las observaciones provienen de una población normalmente distribuida. En este caso, el IC para la media de la población a un nivel de confianza $100(1-\alpha)$ será:

© Los autores, 2002; © Edicions UPC, 2002.

$$\left(\bar{X} - t_{(1-\alpha/2,n-1)} \frac{s}{\sqrt{n}}, \bar{X} + t_{(1-\alpha/2,n-1)} \frac{s}{\sqrt{n}} \right)$$

donde $t_{[1-\alpha/2;n-1]}$ es el cuantil-$(1-\alpha/2)$ de una distribución t con (n-1) grados de libertad.

Esto se basa en la siguiente constatación: para muestras extraídas de una población normal $N(\mu,\sigma^2)$, el estadístico

$$\frac{\bar{X} - \mu}{\sigma/\sqrt{n}}$$

sigue una distribución normal N(0,1).

Tenemos que $(n-1)s^2/\sigma^2$ sigue una distribución χ^2 con (n-1) grados de libertad. En consecuencia,

$$\frac{\bar{X} - \mu}{\sqrt{s^2/n}}$$

sigue una distribución t con (n-1) g.d.l.

Prueba de media nula

Puede interesarnos comprobar si un determinado valor es significativamente distinto de cero. Hablaremos en términos probabilísticos y para ello nos ayudaremos de los IC en el sentido siguiente: con un determinado nivel de confianza mínimo del $100(1-\alpha)\%$, comprobaremos si nuestro valor pasa el test de diferencia:

- el IC incluye el cero → la media es cero
- el IC no incluye el cero → la media es distinta de cero

En otros términos, si el IC incluye el cero, no podemos decir, con un grado de confianza determinado que la diferencia media es significativamente diferente de cero.

Este procedimiento también puede aplicarse para comprobar si la media es igual a un determinado valor m, o sea, si un IC construido de forma similar a la descrita precedentemente, incluye el valor m y, en consecuencia, no podemos rechazar que la media es m a un nivel de confianza especificado.

Sólo en los dos casos en los que el intervalo no incluye el cero podremos afirmar que la diferencia entre medias, con un determinado nivel de confianza, pongamos por caso 95%, es significativamente distinta de cero. En caso contrario no podremos afirmar que una configuración del sistema sea mejor que otra.

© Los autores, 2002; © Edicions UPC, 2002.

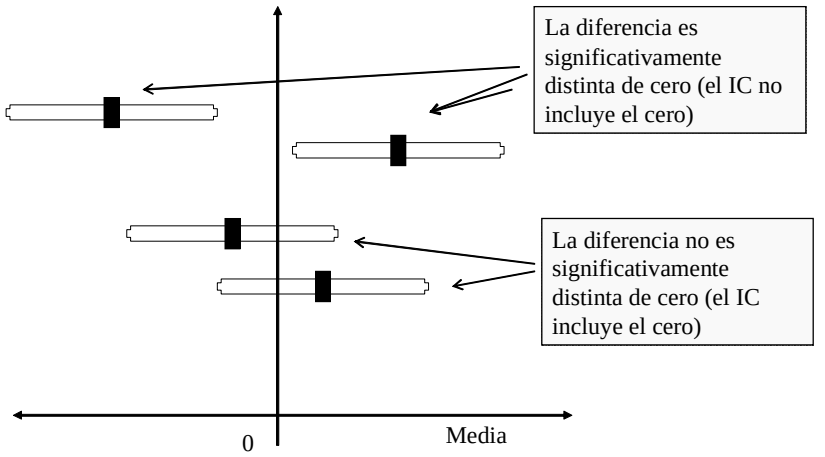


Figura 5.21 Comparación criterio inclusión de cero en intervalos de confianza

Comparación de dos alternativas

Tenemos interés en comparar dos o más escenarios o alternativas de configuración de sistemas. A un primer nivel, suponemos dos sistemas frente a cargas de trabajo similares.

Observaciones pareadas

En este caso, tenemos el mismo número de observaciones y se da una correspondencia uno a uno entre las medidas en cada uno de los sistemas a comparar. Es ésta una situación muy ventajosa de cara al análisis aunque, por desgracia, difícil de alcanzar debido a los problemas de sincronización entre escenarios.

Se trabaja, simplemente, con la diferencia entre cada par de los valores recolectados en los mismos puntos de muestreo y construimos el intervalo de confianza de dicha diferencia. Tal como hemos descrito en el apartado anterior, si el IC incluye el cero, no podremos afirmar que uno de los dos sistemas es significativamente mejor que el otro.

EJEMPLO 5.13

Se comparan dos alternativas de diseño para las cabinas de cobro de peaje del ejemplo 5.1. Se analizan los tiempos medios de espera en una hora punta del día en las taquillas de cobro manual en dirección Norte. Los valores recolectados son, respectivamente:

Alternativa 1: (8.3, 11.2, 14.6, 11.7, 9.9, 10.8)

Alternativa 2: (9.9, 13.1, 10.2, 12.1, 10.6, 9.3)

Las diferencias de los pares de valores son: (-1.6, -1.9, 4.4, -0.4, -0.7, 1.5)

© Los autores, 2002; © Edicions UPC, 2002.

Con lo que la media de la muestra de las diferencias será 0.22, la variancia 5.63, y la desviación estándar 2.63, por lo que el IC de la media al 90% de confianza y grados de libertad es:

$$\text{IC:} \quad [0.22 \pm 2.015 * \sqrt{(5.63/6)}] \quad \text{es decir } [0.22 \pm 0.97] = [-0.75, 1.19]$$

El IC incluye al cero, por lo que concluimos que ambas alternativas no pueden considerarse diferentes en cuanto a esta medida de rendimiento y podemos escoger la de coste inferior.

Observaciones no-pareadas.

Si por el contrario no están pareadas, tendremos n_1 y n_2 observaciones que corresponden, respectivamente, a las muestras obtenidas en la evaluación de las configuraciones alternativas del sistema, A_1 y A_2.

Bajo este escenario procederemos a evaluar las respectivas medias $\overline{X}_1$ y $\overline{X}_2$, y variancias $S_1^2 \, y \, S_1^2$. La variancia de la diferencia media vendrá dada por

$$S = \sqrt{\frac{S_1^2}{n_1} + \frac{S_2^2}{n_2}}$$

y los grados de libertad por

$$\nu = \frac{\left(\frac{S_1^2}{n_1} + \frac{S_2^2}{n_2} \right)^2}{\frac{1}{n_1 + 1}\left(\frac{S_1^2}{n_1} \right)^2 + \frac{1}{n_2 + 1}\left(\frac{S_2^2}{n_2} \right)^2} - 2$$

con lo que el IC vendrá expresado

$$[\left(\overline{X_1} - \overline{X_2}\right) \pm t_{[1-\alpha/2,\nu]} \cdot S \,]$$

EJEMPLO 5.14

Se comparan dos nuevas alternativas de diseño en las que las observaciones no están pareadas. Los valores recolectados son, respectivamente:

Alternativa 1: (7.8, 11.2, 14.6, 11.7, 9.3, 10.8)

Alternativa 2: (11.1, 13.1, 12.0, 12.1, 10.6, 9.3)

Las medias de las muestras son 10.9 y 11.4 y su diferencia –0.46, y la desviación estándar 1.087, por lo que el IC de la media al 90% de confianza y 12 grados de libertad será:

$$\text{IC:} \quad [-0.46 \pm 1.86] = [-2.61, 1.40]$$

El IC incluye al cero, por lo que concluimos que ambas alternativas no pueden considerarse diferentes en cuanto a esta medida de rendimiento y podemos escoger la de coste inferior.

© Los autores, 2002; © Edicions UPC, 2002.

Comprobaciones visuales

Podemos dibujar los IC de ambos sistemas y observar la posición relativa de medias e intervalos. Tenemos tres posibilidades:

a) Los intervalos, además de solaparse, recubren los valores promedio de la otra alternativa, respectivamente. Podemos afirmar que un sistema no es mejor que el otro.

b) Los IC se solapan pero no recubren la media de la otra configuración. En este caso no podemos concluir nada de forma visual y utilizaremos el método descrito anteriormente, basado en la distribución t.

c) Los intervalos no se solapan: una de las alternativas del sistema es la mejor.

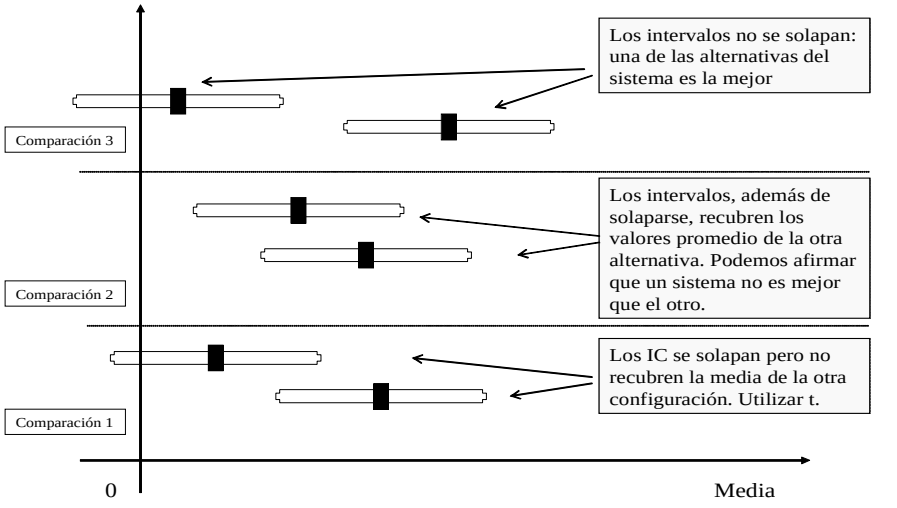


Figura 5.22 Comparación visual de alternativas mediante IC

Intervalos de confianza para proporciones

En muchos estudios de simulación nos interesa trabajar con otro tipo de estimadores al margen de la media, la cual, en algunos casos, no nos proporciona suficiente información. Trabajaremos, pues, con las probabilidades asociadas con valores categóricos estimando las proporciones de una forma similar que la descrita para los intervalos de confianza para la media.

Dada una determinada proporción de la muestra, p, obtenida como cociente entre las observaciones de un determinado tipo y el total de observaciones, obtenemos el intervalo de confianza:

$$\left(p \pm z_{(1-\alpha/2)} \sqrt{p(1-p)/n} \right)$$

dónde z es el cuantil $(1-\alpha/2)$ de la variable normal unitaria (válido para $np>10$).

© Los autores, 2002; © Edicions UPC, 2002.

5.3.5 Técnicas para la reducción de la variancia

Las técnicas para la reducción de la variancia (TRV) son métodos que explotan nuestra capacidad de control de los generadores de números aleatorios (GNA) que conducen nuestra simulación. Normalmente reutilizamos dichos números aleatorios para inducir correlaciones favorables que reduzcan el ruido en el *output*.

De hecho, no deben considerarse el único medio de ahorro de tiempo de ejecución. Con la misma finalidad y alternativamente, podemos utilizar GNA mejores y más eficientes como los que se han presentado en el capítulo 3, y desarrollar procedimientos de muestreo más rápidos y con las máximas garantías de ajuste a las distribuciones de referencia. También pueden alcanzarse importantes ahorros de recursos trabajando con entornos de desarrollo convenientemente probados y en los que se han seguido procedimientos de aseguramiento de la calidad, como los descritos en la sección 5.1, vinculados a los procesos de VV&T. En definitiva, seremos más eficientes si trabajamos con el mejor *software* disponible.

Entre la multitud de TRV existentes, vamos a presentar a continuación aquellas más extendidas y utilizadas en proyectos reales de simulación. Jain, por ejemplo, no considera estas técnicas especialmente eficientes en los proyectos de ingeniería en general [JAIN91], sino que son de mayor utilidad en problemas de tipo teórico o de aplicación de la simulación a la estadística. Cabe destacar que, en algunos casos, la utilización de TRV no garantiza una mejora asociada, sino todo lo contrario: introduce aún mayor variancia en la muestra.

No procederemos, por lo tanto, a una descripción exhaustiva de dichos métodos, los cuales dejamos como futura lectura en las referencias bibliográficas detalladas al final de este libro. Entre ellas destacamos [KLEI92][LAW99].

Situemos, en primer lugar, la finalidad de dichas técnicas. Partimos de la base de que vamos a obtener un estimador de los valores medios de alguna variable de respuesta. La fiabilidad del estimador puede venir expresada mediante su intervalo de confianza:

$$\left(\bar{x} - \mathrm{k}\ {s}/{\sqrt{n}},\ \bar{x} + \mathrm{k}\ {s}/{\sqrt{n}}\right)$$

en el que k es alguna constante, s es la desviación típica y $s/\sqrt{n}$ es el error estándar.

Luego esta valoración dependerá de dicho error estándar, el cual, en principio, disminuye cuando el tamaño de la muestra n aumenta. Las TRV intentan mejorar la fiabilidad del estimador sin la utilización de muestras mayores de lo que sería necesario, si aplicamos criterios que tiendan a hacer disminuir la variancia.

Entre las TRV más conocidas, analizaremos con más detalle las basadas en números aleatorios comunes (*Common Random Numbers (CRN)*), variables antitéticas y variables de control.

© Los autores, 2002; © Edicions UPC, 2002.

Números aleatorios comunes (Common Random Numbers)

Todos los autores citan este método como el más utilizado y el más simple. Incluso hay quien afirma que se llega a usar el mismo de forma intuitiva, automática o involuntaria cuando se utilizan las mismas series de números aleatorios en experimentos sucesivos de comparación de alternativas.

Se trata de obtener una reducción de la variancia de la diferencia a partir de la sincronización de las mismas series de números para distintas configuraciones. A partir de una misma semilla original los sistemas serán comparados en "las mismas circunstancias".

A pesar de su sencillez, su buena utilización exige la sincronización de los diferentes *streams* o series de números aleatorios (capítulo 3), lo que puede no ser evidente ni fácil. Así, deberemos caracterizar cada uno de los generadores de las variables aleatorias que vamos a utilizar en la simulación mediante la identificación de una serie específica para cada tipo de fenómeno, a fin de obtener una máxima sincronización entre la carga de trabajo generada en los escenarios alternativos.

Se trata de introducir una correlación positiva entre las ejecuciones. Esto se explica porque se trabaja con las diferencias en las respuestas de las dos alternativas que estamos comparando. Así, si Z=X–Y es la variable que observaremos como diferencia de las correspondientes variables respuesta, tendremos que:

$$var(Z) = var(X) + var(Y) – 2cov(X-Y)$$

Vemos que para cov(X-Y) alta, por haber introducido correlación positiva entre ambas series, obtendremos una var(Z) menor, que es nuestro objetivo.

Variables antitéticas

En esta técnica, lo que intentamos introducir precisamente es el fenómeno contrario del precedentemente descrito (CRN). Algunos autores advierten del peligro de su uso indiscriminado ya que, en determinados casos, los efectos obtenidos pueden ser opuestos a los deseados [KWON94].

El procedimiento es muy sencillo:

- Realizamos una primera ejecución con la serie de números aleatorios u_i (i=1,2,...n).
- En la segunda ejecución, empleamos precisamente los valores antitéticos ($1-u_i$) (i=1,2,...n).

Esto nos hace pensar que este muestreo complementario nos va a inducir una correlación negativa entre las respuestas de ambas simulaciones. En principio, ambas "se compensan" y se conjetura que las respuestas y(1) y y(2) estarán correlacionadas negativamente, por lo que la variancia de sus promedios decrece. Tenemos n/2 promedios de parejas (y(i), y(i+1)), lo que nos conduce a la expresión de la variancia de Y = Y(i)+Y(i+1):

$$Var(Y)=1/4[var(Y(i) + var(Y(i+1)+ 2\ cov(Y(i),Y(i+1)]$$

Aquí el problema de la sincronización entre eventos en ejecuciones sucesivas sigue siendo un aspecto de difícil control y, por lo tanto, una determinada descoordinación podría incluso introducir correlación positiva (¡exactamente lo contrario de lo que pretendemos!). En algunos estudios, se ha combinado esta técnica con la de CRN de forma favorable y, en algunos casos, ha conllevado una reducción de la variancia estimada de hasta el 80%.

© Los autores, 2002; © Edicions UPC, 2002.

Variables de control

Una de las ventajas de la simulación frente a otras herramientas de experimentación, es la capacidad de intervención en el modelo a través de la conveniente manipulación de su código al introducir mecanismos de monitorización y de compensación de efectos o de ciertos parámetros de forma dinámica a lo largo de la simulación [NELS93].

El método de las variables de control trata de corregir las desviaciones en alguna de las variables de la simulación en razón a su relación con otras de las variables sobre las que tenemos mayor información. En definitiva, los valores observados de las variables de entrada generadas por la simulación son comparadas con los valores teóricos o esperados. Controlamos así las desviaciones lo que nos permite reducir la variancia del estimador a lo largo de la ejecución de la simulación.

Otras técnicas

Existen otros métodos como el de las series temporales estandarizadas, el muestreo estratificado o el muestreo de importancia, que es una modalidad de muestreo estratificado en el que se sobremuestrea en aquellos estratos de datos que consideramos más críticos. En la simulación de sistemas complejos no existe un procedimiento general que nos garantice buenos resultados empleando la estratificación, a diferencia de los problemas en que podemos aplicar métodos de Monte Carlo dónde ésta suele funcionar mucho mejor.

5.3.6 Análisis de sensibilidad

Entendemos por análisis de sensibilidad (AdS) la investigación sistemática de la reacción de las respuestas de la simulación frente a valores extremos del *input* del modelo o a cambios drásticos en la estructura del mismo [KLEI92][KLEI95b][LAW99]. El enfoque no se refiere a pequeños cambios (o perturbaciones) en los valores de entrada, sino que constituye una investigación sistemática sobre la reacción de las respuestas de la simulación a:

- valores extremos en el *input* del modelo (por ejemplo, en la tasa de llegadas)
- cambios drásticos en la *estructura* del modelo (por ejemplo, pasar las prioridades de SPT (*Shortest Processing Time*) a BCF (*Best Customer First*))

Deberíamos considerar, además, un análisis de los cambios marginales**,** y trabajar con perturbaciones en los valores de entrada.

El modelo de simulación es tratado como una caja negra. Aquí, el diseño experimental no aprovecha las peculiaridades de un modelo específico de simulación. Trabajaremos con aproximaciones tipo las del análisis de perturbaciones, o las de muestreo de importancia.

También podemos definirlo como el proceso de medición del efecto sobre el resultado o del rendimiento de un sistema, de un cambio en uno o varios de los *inputs*. Este es un concepto asimilable al de derivada parcial del *output* respecto a uno de los *inputs*.

Para realizar el análisis se precisa el uso de técnicas estadísticas específicas y procederemos en dos fases:

- una primera fase piloto de *screening* o búsqueda de factores relevantes/importantes,

© Los autores, 2002; © Edicions UPC, 2002.

- una segunda, en la que realizaremos un análisis de regresión.

5.3.7 Optimización en simulación

Obtención de configuraciones óptimas

Nuestro objetivo último será encontrar la combinación de niveles de los factores *input* que nos permitan optimizar alguna medida de rendimiento (que nos conduzca al rendimiento óptimo). Como técnicas básicas se emplean la de estimación del gradiente, los metamodelos y la comparación de alternativas.

Trataremos, pues, de optimizar funciones no lineales condicionadas por la aleatoriedad que hemos introducido en el modelo. Una idea viable es estimar las derivadas parciales en un punto (el gradiente), y movernos en esa dirección (*steepest ascent* (máx.) o *descent* (min.)). Podemos ajustar un modelo de metaregresión y utilizar cálculo simple para optimizar, en lugar de utilizar el propio modelo de simulación. En definitiva, se trata de una adaptación de los métodos de programación estocástica.[ANDR98]

Métodos de superficie de respuesta (MSR) y metamodelos

Los métodos de superficie de respuesta (MSR), metodología inicialmente desarrollada por Box, constituyen una colección de herramientas que permiten explorar la relación entre el comportamiento del modelo de simulación en cuanto a la relación entrada/salida. Se basan en la aproximación mediante un modelo polinomial que es ajustado utilizando técnicas de regresión. Actualmente los MSR son considerados como parte del análisis basado en metamodelos y constituyen un primer nivel de aplicación de los mismos [FRIE96].

Si pensamos en la lógica y la acción de la simulación como una transformación de *inputs* en *outputs*, podemos imaginar la simulación como una función que podemos aproximar con una fórmula. Esta aproximación puede ser interesante si implicamos un gran número de *inputs* y el tiempo de ejecución de una repetición de la simulación es considerablemente grande. Se procede a ajustar un modelo de regresión al modelo de simulación, con las variables dependientes como salida y las variables independientes como entrada de la simulación. A continuación, se aplican técnicas como la selección de subconjuntos importantes de variables independientes, la modelización de no linealidades y la consideración de interacciones.

Los metamodelos de regresión generalizan los resultados de un experimento de simulación mediante un número pequeño de factores: un metamodelo de regresión estima la transformación entrada/salida (E/S) especificada en el modelo subyacente de simulación [KLEI98].

Un metamodelo es el *modelo del modelo* subyacente de simulación y es una aproximación de la *transformación E/S* del programa de simulación. Es, como ya hemos mencionado anteriormente, una variante del concepto conocido como método de superficie de respuesta.

© Los autores, 2002; © Edicions UPC, 2002.

Clases de metamodelos de regresión

- Polinomio de primer orden, en el que sólo analizamos los efectos principales. Es un método que podemos considerar asociado a la existencia de una "gran" media.
- Polinomio de primer orden modificado, en el que incluimos más interacciones entre parejas de factores.
- Polinomio de segundo orden, que incluye efectos cuadráticos puros.

Las técnicas de reducción de la variancia pueden ser estrategias útiles para la mejora de las estimaciones de los coeficientes de los metamodelos de simulación. Nos basamos, a partir de aquí, en Donohue [DONO95].

Un metamodelo de simulación es una ecuación matemática que relaciona las variables input y output de un modelo de simulación por computadora.

Entre los tipos de metamodelos más usados en estudios de simulación tenemos:

Los modelos de regresión polinómicos de primer orden, representados por la formulación

$$y_u = \beta_0 + \beta_1 \ x_{1u} + \beta_2 \ x_{2u} + ... + \beta_k \ x_{ku} + \ e_u$$

y los modelos de regresión polinómicos de segundo orden:

$$y_u = \beta_0 + \Sigma\beta_i \ x_{iu} + \Sigma\beta_{ii} \ x_{iu} + ... + \Sigma\Sigma\beta_{ij} \ x_{iu} x_{ju} + \ e_u$$

donde u es el identificador de la ejecución de la simulación; y_u es la variable respuesta, en general la media de las observaciones recolectadas. x_{iu} es el nivel del factor i-ésimo en la u-ésima ejecución; las β son los coeficientes a estimar del modelo mediante análisis de regresión; e_u es el error no explicado del modelo de regresión.

Los modelos de regresión polinómicos son utilizados por muchos investigadores, y actualmente están soportados por diversos paquetes estadísticos (para estimación y análisis) y en ellos se aplican las técnicas de reducción de la variancia.[10]

Existen métodos alternativos, como las aproximaciones a los dominios de frecuencia, los *kernel smoothing models*, las funciones de base radial, los modelos de correlación espacial y los modelos *spline*. Como ventajas podemos destacar la capacidad de ajustar una gran variedad de relaciones curvilíneas, normalmente en forma *piecewise*. Como inconvenientes requieren conocimientos estadísticos sofisticados y además, para dichos modelos alternativos, no se han desarrollado técnicas apropiadas de reducción de la variancia.

El experimentador ha desarrollado un modelo de simulación válido, ha identificado la variable *output* de interés y ha seleccionado un conjunto de k variables *input* controlables x_i. Tiene como objetivo desarrollar un modelo de regresión polinómica para optimizar, predecir o realizar un análisis de sensibilidad.

Como ejemplos tenemos los trabajos de Gordon, Ausink y Berdine [FRIE96], que utilizan modelos de regresión polinómica de segundo orden para evaluar el coste del control de satélites, de Kuei y Madu

[10] Para un mayor desarrollo de este tema ver Friedman [FRIE96].

© Los autores, 2002; © Edicions UPC, 2002.

[KUEI94] sobre costes de mantenimiento para máquinas en sistemas de colas, y de Friedman y Pressman [FRIE88], que relacionan en un modelo genérico M/M/s el tiempo en el sistema con tres variables de entrada (tasa de llegadas, tiempo de servicio y número de servidores). Lo aplican a un sistema de computación de tiempo compartido, que relaciona el tiempo de respuesta con la tasa de llegadas, el tiempo de servicio y la proporción de *overhead*. También analizan un sistema de control de inventarios, que relaciona los costes anuales de inventario con la demanda de piezas, el periodo de revisión y el nivel objetivo de *stock*, basándose en un metamodelo de primer orden en logaritmos de las variables:

$$\ln y_u = \ln \beta_0 + \beta_1 \ln x_{1u} + \beta_2 \ln x_{2u} + \beta_3 \ln x_{3u} + \ln e_u$$

Planes de diseño experimental con metamodelos

Nuestro objetivo consiste en estimar los coeficientes β de los modelos de regresión polinómicos. Deben recolectarse datos experimentales sobre la variable de respuesta y bajo una variedad de condiciones x_i. El diseño de experimentos será la aproximación científica para decidir cómo se recoge dicha información. En el contexto de la simulación, un diseño experimental debe especificar los valores de los k factores *input* controlables x_i para cada ejecución de la simulación, a fin de estimar correctamente los coeficientes β.

Para los modelos de primer orden utilizamos planes factoriales (completos o fraccionales) de 2 niveles. Estos diseños minimizan las variancias de los coeficientes β. Para los modelos de segundo orden utilizamos planes factoriales (completos o fraccionales) de 3 niveles. Estos diseños requieren mucho cálculo (3^k) combinaciones por cada replicación del diseño experimental.

Otros diseños experimentales utilizados son los de Box-Behnken, el *central composite*, el equiradial, el de *Notz*, y el *small composite* [FRIE96].

5.3.8 Análisis de resultados multivariantes

En muchos estudios estaremos interesados en la observación de un conjunto de medidas probablemente relacionadas de algún modo entre sí. Esto configurará un vector de medidas resultado, al que deberemos someter a análisis basados en métodos estadísticos multivariantes.

Esto permitirá estimar todos los parámetros *output* simultáneamente y entender cuáles son los mecanismos subyacentes entre distintas respuestas.

Se trata de resolver un tipo de problemas en los que debemos realizar inferencias estadísticas simultáneas sobre varias medidas resultado de interés para nuestro modelo de simulación [CHAR95].

En la mayoría de los modelos de simulación vamos a tomar decisiones basándonos en más de una medida resultado. Esto es perfectamente lógico dada la complejidad de los sistemas con los que tratamos, y por la facilidad con la que podemos recolectar adecuadamente los valores de muchas variables de nuestro modelo. En oposición a otros procesos de experimentación, realizados como estudios de campo, en los que la limitación de recursos nos obliga a monitorizar pocas variables y en los que los errores de medida suelen ser definitivos y catastróficos, nuestro control sobre el propio proceso de construcción y ejecución del modelo mediante un entorno de desarrollo de proyectos de simulación nos permite una mayor flexibilidad y una mayor abundancia de datos.

© Los autores, 2002; © Edicions UPC, 2002.

Estos datos suelen estar relacionados entre ellos y, a menudo, autocorrelacionados. Ello nos obliga a desarrollar técnicas que detecten estas relaciones entre los distintos resultados generados por nuestro modelo de simulación.

EJEMPLO 5.15

Un caso típico en el mundo de la fabricación es el análisis de los *stocks* de distintas piezas frente a grupos de máquinas, cuando estos deben compartir un mismo espacio en determinadas zonas de la planta. Así, nos interesará saber la evolución de cada uno de estos *stocks* intermedios y determinar si coinciden puntas de almacenaje que nos obliguen a reservar un espacio mayor. Esta situación la podremos determinar mediante el análisis de la correlación entre las longitudes de las colas frente a las máquinas: una correlación alta de las longitudes de cola cuando éstas son largas nos indicará el grado de conflicto en el requerimiento del recurso espacio.

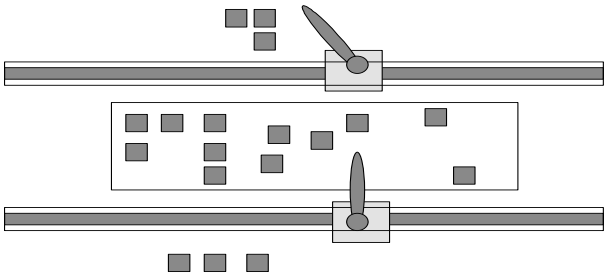


Figura 5.23

EJEMPLO 5.16

Supongamos que una compañía aérea desea dimensionar adecuadamente su servicio de chequeo de pasajes y facturación de equipajes en la nueva terminal del Aeropuerto de Barcelona. Esta compañía asigna puestos de servicio exclusivos a sus clientes preferentes o fidelizados, a quienes desea tratar con esmero y diligencia. El resto de clientes dispone de una batería de mostradores donde van a ser atendidos de forma indistinta aunque también se considera la posibilidad de reservar determinados puestos para vuelos a destinos frecuentes o para viajeros que facturan en grupo.

Se trata de resolver dos cuestiones distintas: por una parte se desea minimizar el tiempo de espera promedio de todos los pasajeros frente a las colas de los mostradores de facturación, aunque se cree conveniente que la espera sea menor para los clientes preferentes. Por otra, y por motivos de imagen, en los momentos de mayor afluencia se pretende que los pasajeros en general tengan la impresión de que los pasajeros preferenciales “también esperan un cierto tiempo” antes de ser atendidos y que ellos no están especialmente mal tratados por la compañía cuando hay congestión.

Para resolver este problema también será conveniente analizar la correlación entre los tiempos de espera en las colas de los distintos tipos de mostradores.

© Los autores, 2002; © Edicions UPC, 2002.

En este caso, si calculamos estimadores puntuales de los parámetros de interés de cada repetición, obtendremos una secuencia de vectores resultado aleatorios, que podrán ser estudiados mediante métodos clásicos de análisis multivariante y obtener así una estimación de las correlaciones entre las componentes del mismo.

Un posible método es, a partir de las distintas repeticiones del escenario de simulación, construir la *región de confianza conjunta* del vector de medias, siempre que asumamos la independencia y la normalidad multivariante del vector de observaciones obtenido tras cada una de las repeticiones. Esto suele ser cierto en el caso de simulaciones terminales en las que realizamos repeticiones independientes, y en las que el cálculo de los promedios a lo largo de la simulación conduce a que los estimadores puntuales estén distribuidos normalmente. Trabajamos observando la matriz de correlación para el vector de medias: cada elemento de esta matriz nos proporciona la correlación entre los distintos estimadores puntuales. Nótese que la región de confianza puede representarse mediante una elipse o un elipsoide si analizamos dos o tres parámetros, respectivamente: de su observación podremos determinar la combinación de los parámetros que nos son, en cada caso, más indicados.

Un segundo procedimiento se basa en la construcción de intervalos de confianza individuales simultáneos sobre la media de cada proceso componente del vector de resultados, para lo que Charnes recomienda dos métodos: el de Scheffé o Roy-Bose y el de Bonferroni [CHAR95].

El método de Scheffé o Roy-Bose construye los intervalos definiendo una región rectangular que circunscribe el elipsoide que define la región de confianza. Bonferroni constituye un método simple para establecer los niveles individuales de confianza para un grupo de procesos univariantes, de forma que nos proporciona una cota inferior del nivel agregado de confianza en el conjunto de inferencias [BALC84].

En general, se cumple que la región de confianza de Bonferroni es más pequeña que la de Scheffé y que no recubre en su totalidad al elipsoide de confianza generado mediante el método de las *regiones de confianza conjuntas*. Este no es más que una generalización del procedimiento clásico de determinación del intervalo de confianza univariante para la media simple, construido mediante la distribución t.

Para el estado estacionario, en el que consideraremos que alcanzamos un equilibrio tal que las medias, las covariancias cruzadas y las autocovariancias del proceso resultado son invariantes a lo largo del tiempo de simulación, procederemos como sigue. En primer lugar, eliminaremos el sesgo que pueda ser introducido por las condiciones iniciales del proceso, no representativas del estado estacionario del mismo, realizando una ejecución suficientemente larga contemplando una fase primera de *warm-up*, los datos de los cuales van a ser truncados y, por lo tanto, no se van a tener en consideración. El punto de corte deberá establecerse a partir de algún método multivariante que nos permita identificar este momento en el que se alcanza el estacionario.

Dadas las características de estos procesos en estado estacionario, y al igual que en el caso univariante, el análisis de los datos debe tener muy en cuenta el problema de la autocorrelación. Aquí, y a diferencia de los procesos terminales, no trabajamos con repeticiones independientes que nos podrían eliminar esta dependencia sino que trabajamos con medidas extraídas de largas ejecuciones, en las que los valores recolectados se ven, en general, muy influenciados por los valores precedentes.

© Los autores, 2002; © Edicions UPC, 2002.

Entre los métodos utilizables podemos destacar: el basado en la *función de autocorrelación de la muestra*, con la que descubrimos la estructura de dependencia de los procesos simulados, el de las *medias de los lotes multivariantes* y, en tercer lugar, al igual que en el caso de simulaciones terminales, construyendo *intervalos de confianza simultáneos* teniendo en cuenta la desigualdad de Bonferroni a la hora de elegir el percentil de la distribución t al construir cada intervalo.

Otros métodos más avanzados pueden basarse en modelos autorregresivos, en el análisis espectral, en series temporales estandarizadas, el método regenerativo, la aplicación de TRV, especialmente variables de control en el caso multivariante, y finalmente, los métodos secuenciales de Raatikainen [CHAR95][RAAT93].

5.4 Ejercicios

1. De los cuatro enfoques a las técnicas de VV&A escoge la técnica que utilizaría el lector para validar un modelo.

a) Construye una tabla comparativa para evaluar los cuatro criterios
b) Justifica la decisión de la técnica adoptada.

2. (Adaptado de [LAW99]) Se quiere modelar y simular una máquina que sigue un esquema M/M/1. Esta máquina es un servidor simple con una cola infinita de tipo FIFO. Los tiempos de llegada y servicio siguen una distribución exponencial.

   La máquina sufre diferentes averías y se ha comprobado que tiene un 90% de eficiencia en su funcionamiento. Para modelar este comportamiento se ha incrementado el tiempo de servicio para que quede incluido el tiempo que está averiada. Las características del sistema se describen en la tabla 5.6

Parámetro	Valor
Tiempo medio entre llegadas	1 min
Tiempo medio de servicio	1 min
Número de clientes servidos	1000

Tabla 5.6 Parámetros del sistema

   Los parámetros reales del sistema con averías son los descritos en la tabla 5.7

Parámetro	**Valor**
Tiempo medio entre llegadas	1 min
Tiempo medio de servicio	0.9 min
Número de clientes servidos	1000
Tiempo activo	270 min
Duración de la averia	12 min

Tabla 5.7 Parámetros del sistema

a) Construye el modelo descrito con los parámetros de la tabla 5.6
b) Construye el modelo descrito con los parámetros de la tabla 5.7

© Los autores, 2002; © Edicions UPC, 2002.

c) ¿Son estos dos modelos el mismo?
d) ¿Qué problemas pueden surgir si no validamos el primer modelo y se incluye éste como submodelo de un sistema de producción?

3. Tras realizar 15 repeticiones de una simulación de la cola de un cajero en un banco se han obtenido los valores medios y máximos descritos en la tabla 5.8.

Replicación	Tamaño medio	Tamaño máximo	Retardo medio	Retardo máximo
1	10.23	35	9.79	28.71
2	5.73	24	5.76	25.56
3	7.27	30	7.18	26.72
4	5.65	33	5.40	31.39
5	8.60	36	8.42	27.88
6	15.66	57	14.52	49.15
7	8.12	29	8.24	26.80
8	8.90	31	8.95	30.42
9	5.05	28	5.19	27.89
10	15.02	51	14.96	51.45
11	4.95	24	5.03	25.01
12	7.07	35	7.66	34.25
13	5.61	35	5.66	33.71
14	5.38	22	5.54	24.30
15	5.42	27	5.34	20.51

Tabla 5.8 Resultados de la simulación

a) Calcula el intervalo de confianza del 90% de los tamaños medios y máximos.
b) Calcula el intervalo de confianza del 95% de los retardos medios y máximos.

4. En las oficinas de atención al público de una sucursal bancaria existen 5 ventanillas de atención, cada una con una cola independiente. El director de la sucursal cree que se reduciría el tiempo de espera en las colas si estas se integraran en una sola cola para las 5 ventanillas.

a) Que experimento(s) de los descritos en este capítulo se utilizaría para evaluar la idoneidad de este cambio.
b) Realiza el modelo y experimento(s) suponiendo que llegan clientes cada medio minuto (distribuidos exponencialmente) y el tiempo de servicio de los cajeros se distribuye normalmente con media de 2.5 minutos y desviación de un minuto
c) Simula el sistema y llega a una decisión.
d) Cómo se ve afectado el espacio por el cambio de política de cola? Que le sucedería al sistema si, aproximadamente, el espacio para las colas es de 10 personas (50 en total)? Como se tendría que organizar el espacio con la nueva situación?
e) Analiza también los tamaños de la(s) cola(s).

© Los autores, 2002; © Edicions UPC, 2002.

5. La misma sucursal está estudiando un cambio radical. Quieren ver que pasaría al sistema si se aplicaran una serie de modificaciones. Cola única versus colas independientes, necesidad de una sexta ventanilla o cinco ventanillas e instalación de dos cajeros automáticos.

a) Diseña los experimentos necesarios.
b) Si un 30% de los clientes utiliza cajeros automáticos y el tiempo de servicio es exponencial de media 1.5 minutos simula los experimentos antes diseñados.

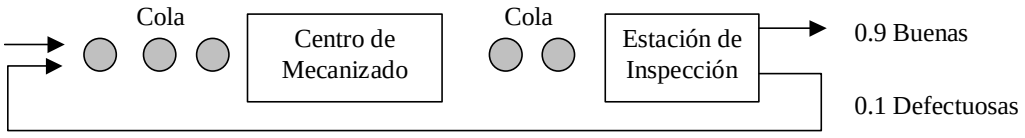
6. En una empresa se ha estudiado el tiempo total de proceso de sus productos. Al final de la línea de producción se encuentra un control de calidad. Si el producto es defectuoso este vuelve a entrar en la línea. La empresa ha encargado un estudio para saber si modificando el funcionamiento del control de calidad y reduciendo su tiempo de proceso se mejora el tiempo total de proceso de los productos. Unos directivos están a favor de la modificación ya que argumentan que reduciendo este tiempo de proceso se reducirá el tiempo global. Otros directivos están en contra y argumentan que al verificar más rápido se puede saturar al sistema con el producto defectuoso. ¿Quien tiene razón?

   En la tabla 5.9 se muestran los valores del sistema original (modelo 1) y los del sistema modificado (modelo 2) del valor medio del tiempo total de proceso.

Replicación	Valores Modelo 1	Valores Modelo 2
1	5.18	3.66
2	6.00	3.44
3	4.98	3.71
4	5.55	3.76
5	5.75	3.64
6	5.72	3.85
7	5.63	3.98
8	5.10	3.62
9	5.82	3.74
10	5.38	3.80

Tabla 5.9 Valores del tiempo total de proceso para las dos soluciones

7. (Adaptado de [LAW99]) Se desea modelar y simular una pequeña factoría. Esta factoría está formada por un centro de mecanizado y uno de inspección es série. En la figura siguiente se muestra un esquema de ésta.



   El problema tiene las siguientes especificaciones:

- El tiempo entre llegadas de piezas es exponencial de media 1 minuto.
- El tiempo de procesado de la máquina es uniforme con el intervalo [0.65,0.7] minutos.

© Los autores, 2002; © Edicions UPC, 2002.

- El tiempo de inspección es uniforme con el intervalo [0.75,0.8] minutos.
- El 90% de les piezas son buenas y el 10% restante son defectuosas y vuelven al centro de mecanizado.
- La máquina sufre averías con una media de una avería cada 6 horas, siguiendo una distribución exponencial. El tiempo que la máquina esta reparándose es uniforme con un intervalo de [8,12] minutos.
- Suponemos la factoría inicialmente vacía.

a) Simula el sistema durante 160 horas y con 10 semillas diferentes.
b) Calcula y representa la producción promedio por horas.
c) Encuentra el instante de tiempo en el que se puede considerar que el sistema ha llegado a su régimen permanente

8. Una empresa tiene subcontratado el mantenimiento de sus 4 estaciones de mecanizado. Una estación requiere de reparaciones y mantenimiento cada 9 días de media. La necesidad de mantenimiento se distribuye exponencialmente. La empresa subcontratada ofrece dos tipos de mantenimiento, el estándar que tiene unos costes de 180,30 € a la hora de trabajo y con un tiempo de reparación normal de media 4 horas y desviación de 1 hora y con un tiempo de desplazamiento exponencial de media 1 hora o un mantenimiento experto con unos costes de 300,50 € a la hora y un tiempo de reparación normal de media 2,6 horas y desviación de 45 minutos y un tiempo de desplazamiento exponencial de media 45 minutos. La empresa subcontratada factura el tiempo de reparación y el desplazamiento de ida y el de vuelta.

a) Simula el sistema durante 2 meses y con 10 semillas diferentes.
b) Estudia los costes de mantenimiento. ¿Existe alguna ventaja en contratar el equipo experto?
c) Si a la empresa le cuesta 625 € cada hora que una de las máquinas está parada, ¿qué contrato de mantenimiento escogeríamos?

9. Al director de la factoría de la empresa del ejercicio anterior le ha interesado mucho el estudio que se ha realizado, pero no está muy seguro de los costes que le hemos calculado. Argumenta que los costes tienen una posibilidad de error muy grande y nos ha pedido que volvamos a realizar estas simulaciones pero con un margen de error máximo de 1000 €.

a) Simula el sistema para decidir cual ha de ser la longitud de la repetición necesaria.
b) ¿Qué resultados obtenemos con las longitudes de simulación obtenidas?

10. Un proceso de fabricación flexible fabrica dos productos distintos. Para la fabricación se dispone de 4 células de trabajo para la realización de las diferentes etapas del proceso. Cada celda consta de un número determinado de máquinas y el proceso de fabricación de los dos productos sigue una secuencia diferente de visitas a las células y los tiempos de proceso son diferentes para cada producto y para cada estación. La tabla siguiente muestra el orden de visita a las células para cada producto.

© Los autores, 2002; © Edicions UPC, 2002.

Producto	**Secuencia**
A	a) Fundición
	b) Torno
	c) Pulidora
B	a) Moldeadora
	b) Torno

El sistema esta equilibrado y el número necesario de máquinas para obtener la producción deseada es el indicada por la columna *máquinas actuales* de la siguiente tabla:

Celdas (factores)	Tipo de celda	Máquinas actuales (-)	Máquinas futuras (+)
1	Fundición	3	5
2	Torno	5	6
3	Moldeadora	7	9
4	Pulidora	10	13

11. Se desea incrementar la producción de los dos tipos de productos. Para ello se puede incrementar el número de máquinas de cada celda de acuerdo con la columna *máquinas futuras* de la tabla anterior. Para analizar el impacto de cada uno de los factores (número de máquinas en cada celda) se efectúa un diseño factorial. ¿Cuál de las siguientes afirmaciones es cierta en relación con el efecto de los diferentes factores? (nota: sólo se diferencia si el efecto es 0 o diferente de 0). El parámetro sobre el que se mide el efecto es la producción total del sistema (producción del producto A más producción del producto B).

    a) e1 > 0, e2 > 0, e3 >0, e4 > 0, e12 > 0, e13 = 0, e14 > 0, e23 >0, e24>0, e34=0

    b) e1 > 0, e2 > 0, e3 >0, e4 > 0, e12 = 0, e13 > 0, e14 = 0, e23 =0, e24=0, e34>0

    c) e1 = 0, e2 = 0, e3 =0, e4 = 0, e12 > 0, e13 = 0, e14 > 0, e23 >0, e24>0, e34=0

    d) e1 = 0, e2 = 0, e3 =0, e4 = 0, e12 = 0, e13 > 0, e14 = 0, e23 =0, e24=0, e34>0

© Los autores, 2002; © Edicions UPC, 2002.

6 Aplicaciones

En la actualidad, la mayor parte de organizaciones, empresas y procesos productivos han visto incrementada la presión para adaptarse a los constantes cambios provocados por una elevada competitividad en los mercados internacionales. Posiblemente influenciadas por las nuevas tendencias en el sector, muchas empresas se han visto involucradas en un proceso de reingeniería de sus procesos.

La falta de herramientas analíticas que ayuden y faciliten la toma de decisiones es uno de los principales problemas con los que se encuentran los responsables de mejorar el rendimiento de aquellos sistemas cuyo comportamiento depende de un número elevado de variables de decisión (sistemas logísticos complejos). Determinar cambios en los procedimientos, en las reglas que utilizan los distintos departamentos en la toma de decisiones, en los flujos de información, así como cambios en la organización y en las políticas de funcionamiento tradicionales de la empresa son algunos ejemplos de tareas que deben ser realizadas para reaccionar de modo eficiente ante perturbaciones y adaptarse a los continuos cambios del mercado.

La dificultad que comportan los procesos logísticos en general, se hace evidente en una gran diversidad de campos, en los que prácticamente se desconoce la influencia que pueden tener sobre el rendimiento global de la empresa cambios en algunas variables de decisión. El problema es todavía más complicado cuando se pretende optimizar (o en su caso mejorar) el rendimiento de un sistema cuyo comportamiento es poco previsible ante cambios en la logística del sistema, debido no tan sólo a las fluctuaciones estadísticas, sino también a las distintas situaciones que se derivan de la dependencia entre sucesos.

Las herramientas de simulación orientadas a eventos discretos ofrecen una plataforma que permite abordar con éxito un proceso de mejora continua de sistemas complejos para los cuales las técnicas analíticas clásicas basadas en el uso de cálculo diferencial, teoría de probabilidades y métodos algebraicos, no pueden ser utilizadas para analizar de modo sencillo la complejidad de los procesos.

A continuación se describen un conjunto de campos en los cuales se ha aplicado o se aplica la simulación para ayudar a la toma de decisiones estratégicas u operativas. Dado que el análisis no es exhaustivo, se recomienda al lector que consulte en la bibliografía especializada para conocer la capacidad de estas técnicas para abordar problemas de su interés.

© Los autores, 2002; © Edicions UPC, 2002.

6.1 Fabricación y sistemas de manipulación de materiales

Una de las áreas en donde tradicionalmente se ha aplicado intensivamente la simulación es en los procesos de fabricación y los sistemas de manipulación de materiales. Las motivaciones que han impulsado el empleo de la simulación son:

- Las empresas están forzadas a ser competitivas.
- El nivel de automatización es muy elevado.
- Las nuevas metodologías, como por ejemplo *just in time*, tienen que ser evaluadas.
- Los procesos de fabricación suelen estar bien definidos.
- Estos procesos son generalmente demasiado complejos para ser analizados con técnicas analíticas.

Los aspectos específicos que se pueden estudiar son:

- *Las necesidades de equipo y/o personal*: número y tipo de máquinas para alcanzar un determinado objetivo; número, tipo y distribución física de los elementos internos de transporte; distribución y espacio para los *stocks*; evaluación del impacto del cambio del *product mix* sobre el ritmo productivo; evaluación del efecto de un nuevo equipo en una instalación existente; planificación de las tareas del personal.
- *Análisis de rendimientos*: análisis de producción, análisis del tiempo en el sistema y análisis de los cuellos de botella.
- *Evaluación de estrategias operacionales*: planificación de la producción, políticas de gestión de inventarios, estrategias de control y políticas de control de calidad.

6.1.1 Medidas de comportamiento

Los métodos empleados para medir el comportamiento del modelo deben ser los mismos que los empleados en el sistema real (o a la inversa). En caso contrario puede ser difícil validar el correcto funcionamiento del modelo. Para el conjunto de medidas que se presentan a continuación, es importante registrar los valores promedios y la dispersión respecto a estos promedios:

- régimen de producción
- tiempo en el sistema de los productos fabricados
- tamaño y tiempo de espera en las colas de los distintos puestos de trabajo
- tiempo de respuesta de los sistemas de manipulación de materiales
- grado de utilización de los equipos y trabajadores

Es de destacar que la optimización o mejora de una de estas medidas se consigue generalmente con el empeoramiento de otra medida. Por ejemplo, si se reducen los tiempos de espera promedios en las colas es probable que también se reduzca el grado de utilización de los equipos.

© Los autores, 2002; © Edicions UPC, 2002.

6.1.2 Componentes de un proceso productivo

La mayoría de procesos productivos manejan recursos como *equipos o trabajadores, y material para procesar*. La disponibilidad de estos recursos está generalmente gobernada por horarios regulares de trabajo, por paradas programadas de mantenimiento y por la presencia aleatoria de averías.

En muchos casos, los sistemas de manipulación de materiales pueden afectar de forma significativa el comportamiento del proceso. En este grupo *incluimos los vehículos autoguiados (AGV), las cintas transportadoras, los toros o los puentes grúa*.

Tanto el flujo de materiales, como los recursos o los sistemas de manipulación de materiales están gobernados por *sistemas de control especializados*. El modelado correcto de estos sistemas de control es uno de los retos principales para obtener un buen modelo de un proceso productivo.

En muchos procesos productivos, el *espacio para el almacenamiento temporal* (buffers) es limitado y puede llegar a ser un cuello de botella del proceso productivo.

El material a ser procesado (piezas) suele tener asociado una *orden de fabricación* que indica la ruta que debe seguir a través del proceso productivo. El conjunto de posibles rutas en un proceso productivo puede llegar a formar una red compleja en la que deben formalizarse correctamente nodos de sincronización (operaciones de ensamblado), así como nodos de bifurcación para representar la independencia entre tareas (operaciones en paralelo).

6.1.3 Fuentes de azar

Algunas fuentes comunes de azar son:

- llegada de materia prima de los proveedores, llegada de piezas o llegada de pedidos
- tiempos de operación o tiempos de ensamblado
- tiempos de operación de los equipos entre averías o fallos
- tiempos de reparación de los equipos o corrección de los fallos
- tiempos de preparación de las máquinas

Algunos de los valores anteriores pueden ser constantes. Por ejemplo, los tiempos de operación o ensamblado pueden ser considerados constantes cuando los procesos estan automatizados y presentan una dispersión muy baja.

Una de las fuentes más importantes de azar son los paros no planificados de los equipos, ya sea por averías, rotura de piezas u otras incidencias significativas. Las paradas planificadas, como por ejemplo descansos, cambios de turno o mantenimiento programado son simples de modelar en contraposición con los paros no planificados descritos anteriormente. Es muy importante describir correctamente este conjunto de incidencias no planificadas; un modelado incorrecto puede comportar que el comportamiento del modelo sea muy diferente al del proceso real.

Un equipo sigue una secuencia ininterrumpida de ciclos de duración variable (figura 6.1). Para cada ciclo, encontramos un periodo de tiempo en que el equipo está operativo (Oi) y el resto del tiempo del ciclo en que el equipo no está operativo (Ni).

© Los autores, 2002; © Edicions UPC, 2002.

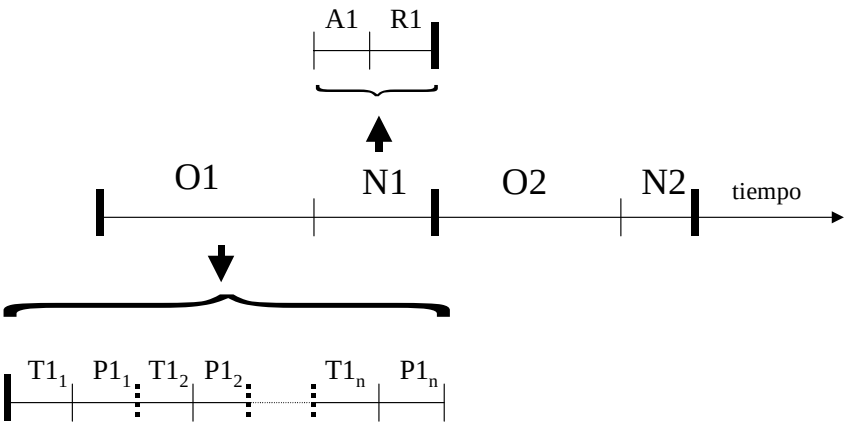


Figura 6.1 Ciclos de las máquinas

El intervalo de tiempo del ciclo en que el equipo está operativo se puede descomponer en ciclos de menor duración. En cado uno de estos ciclos encontramos un primer periodo de tiempo en que el equipo está trabajando (Ti_j) y un segundo periodo en el que la máquina está parada (Pi_j) ya sea por falta de material o bloqueada por la pieza que se acaba de procesar.

El intervalo de tiempo en que el equipo no está operativo se puede descomponer en un primer periodo, en que el equipo está averiado (Ai) y se está pendiente de que se inicie la reparación y un segundo intervalo, en que el equipo está siendo reparado (Ri). Para modelar estos ciclos se puede adoptar dos estrategias:

- *Método del calendario*: parte de la base de que sólo se dispone de muestras de los tiempos Oi y Ni. Si se supone que los valores de cada muestra son independientes y siguen la misma distribución de probabilidad se puede ajustar una distribución de probabilidad a cada una de las dos muestras. El problema principal de este método reside en que el tiempo que tarda una equipo en averiarse depende generalmente del tiempo de funcionamiento total de cada ciclo

$$\sum_j Ti_j$$

  y no del tiempo total de cada periodo operativo (Oi). Por tanto, la cadencia de averías en el modelo representará correctamente la realidad siempre que la relación entre el tiempo total que el equipo está trabajando respecto al tiempo en que el equipo está operativo en cada ciclo sea similar a la realidad. Esto no siempre es así dado que en numerosas ocasiones se emplea la simulación para evaluar configuraciones o formas de trabajar alternativas que cambian la carga de trabajo de los equipos contemplados. No parece sensato que un equipo que trabaja el 90% de su tiempo operativo se averíe con la misma frecuencia que el mismo equipo trabajando el 10% de su tiempo.

- *Método del tiempo de trabajo:* Se define Bi como la acumulación de tiempos $Ti_1+Ti_2+...+Ti_n$. Es decir, Bi es el tiempo total de trabajo del equipo en el ciclo i. Si se dispone de una muestra suficiente de los tiempos Bi, es posible ajustar una distribución de probabilidad a la muestra. Mediante este método el simulador debe provocar la avería cuando el tiempo de trabajo acumulado supera el tiempo de trabajo obtenido de la distribución. El inconveniente principal de

© Los autores, 2002; © Edicions UPC, 2002.

esta aproximación es que la mayoría de fábricas sólo toman estadísticas del tiempo de operación (Oi) entre averías.

Los métodos anteriores se emplean cuando hay suficiente información para ajustar las funciones de distribución de probabilidad. Generalmente, cuanto mayor es el tamaño de la muestra, mejor es el ajuste obtenido. Sin embargo, a efectos prácticos se puede obtener un ajuste suficiente a partir de 20 valores, aunque es preferible tener entre 50 y 100 [ROHR98].

En la práctica real, existen situaciones en las que esta información no está disponible, ya sea por que los equipos todavía no han sido instalados o porque no hay tiempo suficiente para obtener estadísticas. En estos casos, se recomienda emplear el modelo descrito por [LAW91] que supone que se dispone como mínimo del tiempo medio estimado de funcionamiento entre averías (μ_B) y del tiempo medio estimado en que el equipo no está operativo (μ_D). Considerando la eficiencia de un equipo como

$$e = \frac{\mu_B}{\mu_B + \mu_D}$$

Se propone ajustar el tiempo de trabajo a una distribución gamma de parámetro de forma α_B=0.7 y parámetro de escala

$$\beta_B = \frac{e\mu_D}{0.7(1-e)}$$

y el tiempo en que no está operativo a una distribución gamma de parámetro de forma α_B=1.4 y parámetro de escala

$$\beta_D = \frac{\mu_D}{1.4}$$

El tiempo Ai corresponde al intervalo de tiempo desde que el equipo se avería hasta que inicia su reparación. Modela, por tanto, el tiempo de repuesta de la empresa ante incidencias o averías. Es importante destacar que estos tiempos se mantendrán siempre y cuando no cambie el número de operarios responsables del mantenimiento, el número total de equipos o la política de mantenimiento de la empresa.

En los casos en que la información es insuficiente, pero se cree que las paradas por avería pueden tener un impacto significativo, se puede emplear el modelo de simulación para efectuar un análisis de sensibilidad del comportamiento del modelo frente a diferentes hipótesis de tiempos entre averías y duración del tiempo total de no operación.

6.1.4 Sistemas de manipulación de materiales

La simulación se emplea a menudo para el análisis y mejora de los sistemas de manipulación y almacenamiento de materiales. En muchos procesos industriales, el tiempo que se emplea manipulando una pieza es generalmente igual o superior al tiempo de proceso, pudiendo alcanzar valores del 85% del tiempo total de fabricación [BANK96]. Adicionalmente, el elevado coste de estos sistemas combinado con la necesidad de integrar de forma efectiva equipos de diferentes proveedores

© Los autores, 2002; © Edicions UPC, 2002.

hace conveniente el empleo de técnicas de simulación para evaluar la interacción entre los distintos sistemas de manipulación y, por supuesto, asegurar su correcto funcionamiento.

Los aspectos más importantes que se deben tener en cuenta en el modelado y la simulación de sistemas de manipulación de materiales son:

- Tipología del sistema de transporte: AGV (*automated guided vehicles*), cintas, toros,...
- Capacidad de los equipos: número de piezas que se pueden transportar a la vez.
- Velocidad de los equipos: también puede ser de interés incluir la aceleración y la desaceleración.
- Distribución en planta: por ejemplo, ¿cuál es la mejor distribución de una red de transporte de AGV para conseguir transportar un determinado número de *palets*/hora?

Los sistemas de manipulación más empleados son:

- Cintas transportadoras: pueden transportar *palets* o piezas. Se pueden clasificar en acumulativas y no acumulativas. En las no acumulativas, toda la cinta se detiene al cargar, al procesar sobre la cinta o al descargar cada una de las piezas. En las acumulativas, las piezas se mueven mientras no alcanzan un punto de paro u otra pieza.

- Vehículos de transporte: se pueden agrupar en vehículos que siguen una ruta (o conjunto de rutas) prefijadas y los que tiene libertad para escoger el camino. En general, el primer tipo de vehículos están dirigidos por sistemas de control por ordenador mientras que el segundo tipo son conducidos por trabajadores. Un ejemplo del primer tipo de vehículos son los denominados autoguiados (AGV), los cuales son gobernados por complejos algoritmos que tiene como objetivo guiar de forma eficiente la flota de AGV en la planta y evitar posibles conflictos.

- Otros sistemas de manipulación son los sistemas de almacenamiento automático (AS/RS, -*automated storage and retreival systems*-), los puentes grúa o las estaciones de entrada/salida. Los trabajadores también pueden ser considerados dentro del grupo del sistema de manipulación de materiales si, por ejemplo, transportan piezas. En este último caso, se tendrá que contemplar su velocidad y los tiempos de carga y descarga.

El conjunto de estadísticas que se pueden extraer del estudio de simulación para medir el comportamiento del sistema depende principalmente del tipo de sistema empleado. Por ejemplo, las medidas de comportamiento que se emplean usualmente para los vehículos de transporte son:

- Grado de utilización de los vehículos.
- Tiempo total en cada uno de los estados: espera, movimiento con carga, sin carga,...
- Número de viajes efectuados.
- Grado de congestión en cada uno de los caminos.
- Tiempo de espera para el transporte.
- Porcentaje del tiempo a la máxima velocidad.

6.2 Transporte

El sector del transporte es un área con un interés creciente en las técnicas de simulación, puesto que la mayor parte de herramientas disponibles (hojas de cálculo y paquetes de optimización) no son

© Los autores, 2002; © Edicions UPC, 2002.

adecuadas para resolver los problemas de coordinación de los distintos agentes que intervienen en la manipulación, carga, descarga y transporte de material, los cuales inciden de modo determinante en el rendimiento global del sistema.

A continuación se describen algunas de las principales características inherentes a los modelos de simulación de sistemas de transporte:

- Complejidad: el comportamiento de los sistemas suele ser no lineal, y su rendimiento viene determinado por un buen control de las interacciones que aparecen entre los distintos recursos que intervienen en las operaciones de carga/descarga de material, así como por una correcta priorización de las peticiones de transporte en función del estado actual del sistema.
- Dinámica: no es posible realizar una buena estimación sobre el comportamiento del sistema que sea útil para la toma de decisiones en la fase de planificación (modelos en régimen o metamodelos), ya que el rendimiento global del sistema de transporte es muy sensible a las interacciones entre los subsistemas que lo integran, por lo que es imprescindible el desarrollo de modelos de simulación que recojan la dinámica de los distintos subsistemas.
- Azar: los sistemas reales de transporte están sujetos al azar. Algunos ejemplos son: tiempos de tránsito, tiempos de carga/descarga, tiempos entre averías, tiempos de mantenimiento.
- Presión en la toma de decisiones: en ausencia de modelos detallados de simulación, se tiende a tomar decisiones muy conservadoras para evitar problemas por falta de medios logísticos de transporte, lo cual penaliza los beneficios que se pueden obtener.

Ejemplos de simulación se pueden encontrar en todos los modos de transporte. Muchas grandes empresas del sector del transporte, ya sean aéreas, por ferrocarril, barco o camión, emplean en la actualidad la simulación para analizar sus operaciones. Las empresas transportadoras emplean la simulación para racionalizar sus circuitos de transporte y planificar mejor sus operaciones. Los puertos, las compañías de ferrocarril y las terminales aéreas emplean la simulación para determinar su capacidad y mejorar sus operaciones.

6.2.1 Puertos

Un sistema portuario puede ser considerado como la combinación de tres campos tecnológicos diferentes [BRUZ99]:

- Los puertos se encuentran altamente influenciados por el desarrollo y las tendencias de los mercados internacionales, incluyendo el rápido crecimiento de transporte de productos entre países de diferentes continentes, y su comportamiento tiene un peso importante en el transporte a nivel mundial.
- Los constantes cambios en las tecnologías y estrategias logísticas tienen una repercusión directa en el rendimiento de los terminales.
- La eficiencia en la gestión de un puerto depende de la correcta gestión de la información así como de la posible coordinación de los distintos agentes que intervienen en las operaciones de transporte, descarga y carga en los terminales portuarios.

© Los autores, 2002; © Edicions UPC, 2002.

Como consecuencia de los avances tecnológicos en los tres campos mencionados, los puertos se encuentran sometidos a constantes modificaciones tanto en la adaptación y mejora de su infraestructura, como en la logística global del sistema. Por estos motivos, los terminales portuarios deben ser considerados sistemas en constante evolución: los continuos cambios en los mercados nacionales e internacionales repercuten directamente en los sistemas de transporte, los cuales deben tener una gran capacidad de reacción.

Al mismo tiempo, los terminales portuarios también presentan características muy similares a los sistemas de producción intensivos: jornadas de 24 horas diarias los 365 días del año, sometidos a puntas de trabajo y muy sensitivos a las condiciones del tiempo y a los patrones de llegadas, con la particularidad que la infraestructura que se utiliza requiere de grandes inversiones económicas.

Desde el punto de vista organizativo, las actividades que se desarrollan en un terminal portuario, requieren la correcta coordinación de un volumen considerable de agentes que, en general, pertenecen a distintas empresas o grupos, como son por ejemplo: autoridades públicas, autoridades marítimas, operadores de terminales, compañías navieras, compañías de transporte, ferrocarril, etc.

Dentro de este contexto, los terminales de contenedores son uno de los subsistemas más complejos a la hora de gestionar, puesto que la posibilidad de obtener pequeños márgenes de beneficios en la explotación de un terminal de contenedores depende directamente de la toma de decisiones por parte del sistema gestor, al mismo tiempo que la toma de decisiones errónea puede conllevar penalizaciones importantes. A nivel de ejemplo, los puertos de Rotterdam y de Génova solían manipular la misma cantidad de contenedores a principios de los 70, 10 años después, el tráfico de contenedores en el puerto de Rotterdam había crecido 10 veces más que el de Génova.

El uso de herramientas de simulación puede aportar muchos beneficios, tanto para los grandes operadores, como para distintos usuarios potenciales, ya que la simulación de procesos portuarios ofrece una metodología esencial para gestionar y conducir eficientemente y con garantías de éxito las constantes mejoras y adaptaciones que se producen en los puertos. Algunas de las principales ventajas en la aplicación de la simulación en los cambios de los sistemas portuarios son:

- Facilita la comprensión de las interacciones entre los distintos agentes presentes en un puerto, las cuales determinarán el rendimiento de los distintos terminales, ayudando de este modo a un buen diseño de los algoritmos de control que gestionarán y priorizarán los distintos recursos compartidos.
- La detección de errores en las primeras fases de diseño y planificación de las actividades portuarias para las cuales se va a redimensionar la infraestructura en los diversos terminales.
- Evaluar diferentes alternativas de terminal portuario además de la realizada a partir de un estudio analítico en régimen, con el objetivo de conocer los beneficios que se obtendrían mediante otras configuraciones ante distintas perturbaciones.
- Herramienta de ayuda para entender los complejos mecanismos de interacción presentes en sistemas a gran escala.

© Los autores, 2002; © Edicions UPC, 2002.

Requisitos de la simulación

Aunque existen una gran diversidad de herramientas relacionadas con la mejora de la gestión de terminales portuarios, la mayor parte se utilizan para la estimación de operaciones simples, como son, por ejemplo, la estimación de tiempos de ocupación de los atraques en función de la planificación de las operaciones establecidas. Los paquetes tradicionales normalmente soportan tan sólo la planificación de terminales portuarios en horizontes de trabajo relativamente cortos, considerando cada tarea por separado, como suele ser, por ejemplo, la estimación y el análisis de las tareas de carga/descarga de un barco, sin considerar las interacciones con otros barcos atracados en la misma terminal.

Evidentemente, en el sistema real, las interacciones entre los distintos agentes que participan en las actividades portuarias de una terminal son, justamente, los factores críticos de los cuales va a depender el rendimiento global de las actividades portuarias. Algunas de las principales interacciones que conviene tener en cuenta son:

- Interferencias entre las unidades de carga/descarga terrestres.
- Interferencias entre grúas portuarias.
- Interferencias entre puentes grúas.
- Problemas de acceso a los puntos de carga/descarga.

Así pues, las operaciones en los terminales portuarios se ven afectadas por un número elevado de interferencias, debido a las operaciones de descarga/carga concurrentes que se realizan en cada terminal sobre varios barcos a la vez, con varias grúas portuarias en los mismos intervalos de tiempo.

La complejidad de gestión de los terminales portuarios reales viene en parte provocada por el comportamiento no lineal de las operaciones de manipulación, por lo que el uso de planificadores de cargas de trabajo aisladas del contexto real (interacciones entre los distintos elementos) tan sólo ofrece estimaciones sobre tiempos de ocupación de la infraestructura de los terminales, que no es útil para llevar el sistema al máximo rendimiento.

La simulación puede ser una herramienta muy útil tanto para realizar correctas estimaciones de los distintos parámetros de rendimiento del sistema, como para la toma de decisiones en las cuales es muy difícil prever cómo afectarán al resultado final. El principal problema en el uso de simuladores es la complejidad de desarrollo de modelos de los distintos elementos que interactúan en los terminales portuarios, por lo que es muy difícil generar un simulador que contemple todas las actividades portuarias. Un aspecto importante en el desarrollo de estos simuladores es la capacidad gráfica y de animación, la cual debe ser considerada prácticamente como prerrequisito para dar garantías a los usuarios finales sobre la consistencia y validez del modelo.

Algunos de los problemas que pueden ser abordados de modo efectivo mediante el uso de las herramientas de simulación en los terminales portuarios son:

- Redes de distribución:
    - Planificación de los flujos de entrada/salida de material por vía terrestre y por vía marítima de un grupo de terminales.

© Los autores, 2002; © Edicions UPC, 2002.

  - Mejora de la infraestructura de las terminales para explotar los beneficios del transporte multimodal.

- Terminales:
  - Diseño de nuevas unidades para la manipulación y ubicación de cargas.
  - Planificación y coordinación de las distintas operaciones que se deben realizar en el puerto y en la vía terrestre.
  - Entrenamiento para la toma de decisiones en planificación.

Otro aspecto que se debe tener en cuenta, es que los simuladores de terminales portuarios deben acceder con frecuencia a grandes bases de datos dinámicas, lo cual requiere de modelos de simulación muy eficientes en que no hayan esperas activas, ya que el acceso a bases de datos dinámicas por parte de un simulador es un aspecto crítico en cuanto el simulador debe ir coordinado en el tiempo con el sistema real.

6.2.2 Ferrocarriles

El modelado, la simulación y el análisis de estaciones de cabecera y estaciones de paso con el objetivo de planificar la gestión de los ramales y muelles de carga ante perturbaciones en las operaciones que deben ser atendidas, está considerado una tarea difícil que comporta una inversión considerable de tiempo y de personal especializado.

La coordinación y sincronización efectiva de toda la infraestructura ferroviaria en una estación cabecera es un factor decisivo para poder ofrecer con éxito sus funciones ante las distintas perturbaciones a las que se ve constantemente sometida. La reasignación de recursos de modo seguro y eficiente es una tarea compleja que comporta la optimización de un problema multicriterio y multiobjetivo.

Tradicionalmente, el diseño, análisis y control de la infraestructura ferroviaria se había abordado con herramientas de optimización (investigación operativa), pero dada la evolución a la que se han visto sometidas las estaciones cabeceras durante la última década, éstos se han convertido en sistemas altamente complejos (asignación eficiente de los terminales de pasajeros, coordinación de recursos en los muelles de carga y de las unidades de almacenamiento, clasificación de vagones, paso de convoyes, etc.), que deberían ofrecer más flexibilidad para poder responder de modo eficiente a las constantes perturbaciones a las que se ven sometidos.

Los modelos analíticos utilizados tradicionalmente no sirven para abordar de modo eficiente los problemas de *routing*, *schedulling* y *dispatching* ante cargas de trabajo reales (patrón de llegadas aleatorio en lugar de determinista) para evaluar estrategias alternativas de gestión en una estación, para prever y analizar soluciones sobre el rendimiento de los recursos ante retrasos en las llegadas de convoyes y/o cabezas tractoras, ni para evaluar metodologías que permitan minimizar los efectos de las perturbaciones.

La reasignación (*re-schedulling*) de recursos juega un papel muy importante cuando aparecen perturbaciones que fuerzan al sistema a situarse en condiciones anormales de funcionamiento, las cuales suelen conllevar situaciones de conflicto en el tiempo en el uso de un mismo recurso (ramales). Puesto que el volumen del transporte aumenta, el tráfico de convoyes es tan denso en algunas vías,

© Los autores, 2002; © Edicions UPC, 2002.

que el retraso de trenes tiene tendencia a verse incrementado ante la aparición de perturbaciones. De hecho, una vez que el flujo programado de los convoyes ha sido perturbado, cuesta mucho esfuerzo poder recuperar el régimen de trabajo previsto, de tal modo que una perturbación sobre una cierta zona puede propagarse por toda una línea férrea, y repercutir con retrasos considerables y problemas de congestión. La posibilidad de tomar decisiones rápidas (pero seguras) en un corto plazo de tiempo es imprescindible para minimizar algunos de los efectos colaterales que provocan ciertas perturbaciones.

Los modelos de simulación de sistemas ferroviarios ofrecen el marco ideal para entender la propagación de los efectos de las perturbaciones, y diseñar algoritmos de control así como evaluar distintas alternativas que se podrían implementar ante la aparición de dichas perturbaciones. Por otro lado, el uso de los modelos de simulación como herramienta de ayuda a la toma de decisiones no tan sólo permite mejorar el rendimiento de la infraestructura ferroviaria y de las condiciones de trabajo, sino que también permite en algunos casos reducir el tiempo total de transporte así como la ocupación de ramales en las estaciones y, de modo más indirecto, permite reducir los costes asociados al uso de locomotoras y vagones, así como los costes asociados al personal (maquinistas y conductores). En la literatura se encuentran numerosos trabajos relacionados con la simulación ferroviaria [DALA01, MIDD01].

6.2.3 Gestión de aeropuertos

A pesar que el transporte es un factor clave en el desarrollo económico de cualquier región, la aviación recibe en la actualidad poco crédito como factor que contribuye a conectar suministradores, industrias y consumidores con un servicio eficiente y competitivo. Hoy en día, cualquiera conoce los importantes beneficios que las autopistas ofrecen a los sistemas de transportes, pero no es así en el caso de aeropuertos.

Adicionalmente a los beneficios directos que genera un aeropuerto (puestos de trabajo directamente relacionados con las actividades aeroportuarias), existen un conjunto de actividades indirectas como son: uso de servicios locales para el transporte de paquetes, suministro de alimentos cocinados a la tripulación, mantenimiento de la flota de aviones, transporte de pasajeros para acceder o salir del recinto aeroportuario, compra regular de combustible o equipos y servicios de los suministradores locales. Por otro lado, ciudades con buenos recursos aeroportuarios se benefician tanto del turismo como de convenciones de negocios, lo que repercute indirectamente en hoteles, restaurantes, tiendas, deportes, espectáculos, alquiler de coches, y transportes públicos entre otros.

Existen también un conjunto amplio de actividades en los aeropuertos de menor dimensión que todavía no son tan explotados como sería de esperar, como son, por ejemplo: operaciones de búsqueda y rescate, transporte médico, vuelos de negocios y de ejecutivos, vuelos chárters, taxi aéreo, entrenamiento de pilotos, transporte de personal, y usos industriales, comerciales y/o de recreación.

Necesidad de modelos de simulación para la gestión de la infraestructura aeroportuaria

La coordinación y sincronización efectiva de todos los recursos de un aeropuerto es un factor decisivo para poder ofrecer con éxito su funciones ante las distintas perturbaciones a las que se ve

© Los autores, 2002; © Edicions UPC, 2002.

constantemente sometido. En general, debido a una falta de metodología así como de herramientas para poder gestionar de modo eficiente los recursos aeroportuarios actuales, se suele abordar el problema de capacidad de los aeropuertos realizando inversiones considerables de capital en la adquisición de nuevos recursos, o aumentando la capacidad de los ya existentes. Sobredimensionar un aeropuerto no conduce a la solución de una mejor gestión de sus recursos, sino que más bien repercute en forzar un crecimiento de recursos justamente donde más difícil será absorber de modo eficiente las cargas de trabajo no esperadas. Cabe notar, por otro lado, que no se ha invertido suficientemente en métodos para hacer más efectivos los recursos existentes.

Considerando el elevado coste y el largo horizonte en cuanto a tiempo, requeridos para las mejoras de un aeropuerto, la planificación es un factor esencial para determinar tanto los recursos necesarios así como para una correcta priorización en el tiempo de los recursos actuales, para poder obtener mientras tanto un rendimiento máximo de los elementos disponibles.

Un elemento muy importante para poder atender de modo eficiente las actividades que deben ser realizadas en un aeropuerto es poder “anticiparse y estudiar soluciones efectivas” tanto para los picos de trabajo (flujo de pasajeros en época turística, congresos, etc.) como para las necesidades futuras de transporte aéreo. Estas previsiones son particularmente difíciles de evaluar, puesto que las cargas de trabajo futuras depende de muchos factores, de entre los cuales cabe considerar factores socioeconómicos de las áreas a las que el aeropuerto da servicio: demografía, renta per capita, factores geográficos, capacidad competitiva, factores sociológicos y finalmente, factores políticos.

A pesar de que la planificación de un aeropuerto debe ser considerada como parte integrada de la planificación de todo el sistema de transporte, la descripción (formulación de un modelo matemático) del conjunto de tareas que deben ser realizadas sobre los recursos (infraestructura aeroportuaria) para optimizar el flujo de pasajeros, permite modelar cada uno de los aeropuertos en particular como un sistema de toma de decisiones.

Tradicionalmente, el diseño, análisis y control de la infraestructura aeroportuaria se había abordado con herramientas de optimización (investigación operativa) pero, dada la evolución a la que se han visto sometido los aeropuertos durante la última década, éstos se han convertido en sistemas altamente complejos (asignación eficiente de los terminales de pasajeros, gestión óptima de los terminales de facturación, coordinación de las unidades de almacenamiento, clasificación y transporte de maletas, etc.) que deberían ofrecer más flexibilidad para poder responder de modo eficiente a las constantes perturbaciones a las que se ven sometidos.

Los modelos analíticos utilizados tradicionalmente no pueden ser utilizados para abordar de modo eficiente los problemas de *routing*, *schedulling* y *dispatching*, ante cargas de trabajo reales (patrón de llegadas aleatorio en lugar de determinista), para evaluar estrategias alternativas para operar un terminal, para prever y analizar soluciones sobre el rendimiento de los recursos ante retrasos en las llegadas de pasajeros, maletas y/o aviones, ni para evaluar metodologías para minimizar los efectos de las perturbaciones.

© Los autores, 2002; © Edicions UPC, 2002.

Desarrollo de modelos de simulación

La simulación digital ofrece una herramienta indispensable en la búsqueda de nuevos caminos para hacer más eficiente el uso de los recursos de que actualmente dispone un aeropuerto, o evaluar los cuellos de botella que deberían ser solucionados en función de los beneficios que se obtendrían.

Para poder utilizar las ventajas que ofrecen los actuales entornos de simulación en el campo de la gestión de infraestructura aeroportuaria, es necesario disponer de modelos de simulación orientados a eventos discretos, que describan en el dominio temporal tanto los recursos de los que se dispone y las restricciones lógicas impuestas por la secuencia de operaciones a ser realizadas, así como una formulación matemática de las cargas de trabajo a las que se va a someter el sistema.

A modo de descripción, los modelos deben formularse teniendo en cuenta las siguientes actividades, así como su distribución a lo largo del día:

1. Flujo de pasajeros: se suelen identificar por separado los pasajeros que llegan, salen o se encuentran en tránsito en el aeropuerto. El registro histórico de estos datos por temporadas permite realizar extrapolaciones a futuro.
2. Flujo de carga aérea: se clasifica el número total de toneladas, en paquetes prioritarios, no prioritarios, urgentes y mercancías varias. El uso industrial de este servicio suele preverse no tan sólo a partir de modelos estadísticos históricos, sino también a partir del crecimiento del tejido industrial del área a la que el aeropuerto da servicio.
3. Movimiento de aviones: el número total de movimientos repercute directamente en el uso de los recursos aeroportuarios que determinan de modo considerable la capacidad de un aeropuerto. La previsión futura de movimientos suele obtenerse a partir de una correlación entre el número de pasajeros, las toneladas de carga y la capacidad de los aviones.
4. Aviones con base en el aeropuerto: número total de aviones monomotores, multimotores o de turbinas que utilizan o podrían utilizar en el futuro el aeropuerto como base.
5. Número de operaciones en las horas punta: la predicción de este parámetro es esencial dada su relación directa tanto con la capacidad del aeropuerto, como en la generación de retrasos en las operaciones aeroportuarias.
6. Flota de aviones: en la evaluación de las necesidades de nuevos recursos aeroportuarios, es necesario prever el tipo de aviones que utilizarán el aeropuerto, puesto que va a repercutir directamente en la capacidad de los grupos de pasajeros que pueden ser transportados en bloque, así como en el número de operaciones que se realizan.

Una vez se dispone de modelos que permitan preveer la cantidad de actividades que deben realizarse en el aeropuerto, así como de un modelo que describa los distintos recursos de los que se dispone, y teniendo en cuenta que se desea gestionar del modo más eficiente posible los recursos disponibles para absorber las cargas de trabajo, las técnicas de simulación suelen ser utilizadas para evaluar las distintas alternativas (cambios en la logística) de gestión de recursos compartidos, con el objetivo de minimizar los retrasos, los cuales se acaban traduciendo en un incremento en el coste de las operaciones por parte de las líneas aéreas (y en definitiva en los usuarios finales) y en un pérdida de tiempo en los pasajeros.

© Los autores, 2002; © Edicions UPC, 2002.

En cuanto a previsión de inversiones a futuro, las herramientas de simulación permitirán identificar los recursos que deben ser actualizados y/o ampliados para poder absorber el incremento de flujo aéreo, el incremento en el número de pasajeros, el incremento en toneladas de carga, mejoras en los mecanismos de acceso al aeropuerto, así como la evaluación de las repercusiones en las mejoras de otros recursos.

El modelado y simulación de los recursos aeroportuarios se puede efectuar desde dos perspectivas complementarias:

- Recursos aéreos: son aquellos recursos utilizados por los aviones para realizar sus operaciones. Se consideran dentro de este primer grupo: las pistas de salida y de aterrizaje, las pistas utilizadas en los movimientos de los aviones entre los terminales y la pista de salida/aterrizaje, las pistas y terminales donde los pasajeros realizan las operaciones de embarque/desembarque y las pistas y terminales donde se realizan las operaciones de carga.

- Recursos terrestres: incluyen todos aquellos recursos del aeropuerto que dan servicio a los pasajeros, tales como: terminales, áreas de espera, taquillas, transporte de maletas, restaurantes, tiendas, alquiler de coches y terminales de facturación. En la actualidad, también se consideran los recursos de aparcamiento, estación de ferrocarril y unidades de transporte. El objetivo principal que se persigue en la gestión de estos recursos es alcanzar un compromiso aceptable entre los intereses (confort) de los pasajeros, el rendimiento de los recursos, las inversiones realizadas y los factores físicos y psicológicos de confort.

El modelo conceptual deberá formalizar los siguientes aspectos:

- Proceso de llegadas y salidas de aviones, mercancías y pasajeros.
- Asignación de terminales a aviones.
- Asignación de terminales de facturación a compañías aéreas.
- Generación de mercancías y uso de los recursos necesarios hasta ser almacenados en el avión.
- Descripción del flujo de pasajeros, mercancías así como de la tripulación asignada a cada uno de los aviones.

La experimentación con los modelos de simulación desarrollados desde esta perspectiva, permitirán satisfacer objetivos tales como:

1. Evaluar la capacidad de absorción por cada recurso de las peticiones de trabajo actuales y futuras.
2. Flexibilidad que ofrece cada recurso para absorber cargas no esperadas [HUTC01].
3. Coordinación con recursos de tierra: flujo de trenes, capacidad del aparcamiento, flujo de autobuses, etc.
4. Coordinación con otros aeropuertos: cambios logísticos para minimizar las repercusiones de retrasos generados en otros aeropuertos.
5. Evaluar los beneficios obtenidos en la gestión de recursos ante mejoras tecnológicas.
6. Evaluar las inversiones en recursos respecto a los beneficios que se obtendrían.
7. Evaluar alternativas de gestión de los recursos actuales disponibles para absorber tanto perturbaciones como futuras cargas de trabajo.

© Los autores, 2002; © Edicions UPC, 2002.

6.3 Redes de distribución (*supply chain*)

En el mundo de las corporaciones virtuales, ya no son las empresas productoras las que compiten entre sí, sino las redes de distribución. Un ejemplo de corporación virtual es Nike. Esta corporación no tiene fábricas, camiones o tiendas. Se caracteriza, sin embargo, por ser una de las corporaciones con más éxito en el área de las prendas deportivas.

La mayoría de estas corporaciones dependen de un conjunto de suministradores, recursos de transporte, fábricas y almacenes para el correcto funcionamiento de sus redes. Como consecuencia del uso combinado de empresas propias y empresas externas es a menudo difícil conocer el impacto de los cambios o las vías de actuación para mejorar el comportamiento de las redes. Por ello, es necesario disponer de herramientas para la ayuda a la planificación estratégica, táctica y operacional. La planificación estratégica y la táctica tienen como objetivo la toma de decisiones con una perspectiva de medio y largo plazo respectivamente; generalmente contempla la adquisición de recursos o inversiones de magnitud elevada. La planificación operacional tiene como objetivo la toma de decisiones en el corto plazo, principalmente busca la optimización de los recursos disponibles.

Las técnicas de simulación proporcionan mecanismos para examinar y mejorar las redes de distribución de una forma que no podría ser alcanzada con otras técnicas. La complejidad y variabilidad asociada con estos sistemas hace prácticamente imposible su análisis de forma precisa con métodos analíticos convencionales [DOUG97, INGA99].

6.3.1 Componentes de una red de distribución

Una red de distribución está compuesta por:

- *centros* geográficamente dispersos en los que la materia prima, los productos intermedios o los productos acabados son adquiridos, transformados, almacenados o vendidos. Los centros se pueden agrupar en plantas, centros productivos que efectúan transformaciones a los productos, y centros de distribución, en los cuales se reciben, almacenan o distribuyen los productos. Por supuesto, también existen centros híbridos que tienen tanto capacidad de transformación como de distribución.
- *redes de transporte* que conectan los centros a través de los cuales fluyen los productos. Estas redes pueden ser representadas como grafos; los nodos representan los centros y los arcos entre nodos representan las conexiones directas de transporte permitidas o habilitadas para la gestión de la red.

El objetivo tradicionalmente perseguido en la gestión de la red de distribución es minimizar el coste total de una determinada petición. El coste total puede incluir aspectos tales como:

- Coste de la materia prima y otros costes de adquisición.
- Costes de transporte.
- Costes directos e indirectos de fabricación.
- Costes directos e indirectos asociados a los centros de distribución.
- Costes de mantenimiento de inventarios.

© Los autores, 2002; © Edicions UPC, 2002.

Cuando se construye un modelo para la ayuda a la planificación, se puede decidir, en función de los objetivos planteados, contemplar toda la red de distribución o bien solo una parte.

Se puede argumentar que la minimización de los costes es un objetivo inapropiado dado que en general las empresas tienen como objetivo principal maximizar el beneficio neto (ingresos menos costes totales). La razón principal de este planteamiento es que los modelos son capaces de contemplar de forma rigurosa los costes de las diferentes alternativas de funcionamiento y aportar información sobre las ventajas e inconvenientes de cada alternativa.

6.3.2 Tipos de modelos

Son muchos los modelos que se pueden emplear para la toma de decisiones en el ámbito de las redes de distribución, por ejemplo:

- Modelos de predicción: se utilizan para describir la demanda futura de productos a ser producidos por la empresa, así como aspectos relacionados con los costes de las materias primas en función de datos históricos.
- Modelos de costes: describen la evolución de los costes directos e indirectos en función de los parámetros más significativos que actúan sobre éstos.
- Modelos de simulación: describen de forma muy precisa el comportamiento a lo largo del tiempo de toda o partes de la red en función de los parámetros más significativos y estrategias de funcionamiento seleccionadas.
- Modelos de optimización: permiten determinar el punto de diseño o el punto de funcionamiento óptimo de un sistema.

Aunque la distinción entre simulación y optimización es evidente desde el punto de vista metodológico, es mucho menos claro desde el punto de vista empresarial. En la práctica real, la simulación puede ser el único método capaz de proporcionar información suficiente para mejorar el rendimiento del sistema cuando el impacto de los efectos aleatorios es significativo [SHAP01].

Un aspecto todavía no resuelto de forma efectiva, siendo por tanto un campo de investigación abierto, es la integración de los modelos de optimización con los de simulación de eventos discretos para optimizar diseños o procesos complejos.

6.3.3 Modelos de simulación

Los tipos de problemas apropiados para ser abordados con técnicas de simulación se pueden agrupar en tres categorias:

- Nuevo diseño.
- Evaluación de diseños alternativos.
- Rediseño y mejora de procesos existentes.

Los modelos de simulación son generalmente desarrollados para la ayuda en:

- la toma de decisiones estratégicas: distribución geográfica de los centros productivos y de los almacenes de distribución; diseño de los centros; modos de transportes;...

© Los autores, 2002; © Edicions UPC, 2002.

- la toma de decisiones tácticas: políticas de gestión de inventarios; niveles de servicio; diseño de políticas de gestión *pull* o *push* de ordenes; ...
- la mejora operacional: cambios en los modos de transporte; adecuación de los medios a la fluctuación de la demanda; otros cambios operacionales; ...

Algunas medidas de comportamiento que se pueden emplear para analizar el comportamiento del sistema son: utilización promedio de almacenes y recursos de transporte; niveles de inventario de los centros productivos y de los almacenes; retardos que se incurren en los diferentes medios de transporte; nivel de cumplimentación de las ordenes de los clientes; ...

Los trabajos de simulación que aparecen en la bibliografía relacionados con este campo son innumerables. Abarcan desde el estudio completo de la red [JAIN01, PARS01] hasta la gestión de elementos particulares como puede ser la gestión de un almacén.

6.4 Otras aplicaciones

La simulación se ha aplicado con éxito en otros muchos campos, como son por ejemplo, la mejora del servicio de urgencias de un hospital, el diseño de planes de evacuación (desde edificios hasta zonas geográficas), el dimensionamiento de redes de comunicación para garantizar un nivel de flujo de información en función de las necesidades concretas de cada proyecto, la mejora de procesos administrativos, así como la atención al cliente en las empresas de servicio.

A continuación se describen muy brevemente algunos aspectos relacionados con el uso de los simuladores en algunos campos particulares de aplicación.

6.4.1 Sanidad

La presión sobre el entorno sanitario para controlar los costes es cada vez más fuerte. En consecuencia, uno de los restos principales con los que se enfrenta en la actualidad el sistema sanitario es como incrementar la eficiencia de las actividades que se realizan en centros sanitarios como son los hospitales. Es decir, determinar la mejor forma de organizar los múltiples recursos necesarios para proporcionar un servicio de calidad manteniendo o reduciendo los costes. La distribución o planificación de estos recursos no puede ser efectuada con la ayuda de simples fórmulas dado que las interelaciones entre los diferentes recursos son generalmente complejas. La simulación es una herramienta adecuada para el análisis y la ayuda a la toma de decisiones por su capacidad para modelar estas relaciones y los factores aleatorios inherentes a estos sistemas. La simulación permite explorar múltiples opciones, sin la necesidad de una inversión importante de recursos, de tiempo, de equipos y, de modo especial, sin degradar el nivel de servicio actual:

- Una de las aplicaciones clásicas de la simulación en el entorno sanitario es la determinación de los recursos necesarios (camas, salas de operación, de Rayos-X, ‘boxes’, médicos, enfermeras,…) para un centro determinado dado un conjuntos de restricciones y de parámetros (tipología de pacientes, flujos de pacientes, horarios, tiempos de servicio …) [CENT00, ROSS99, RAMI01].

© Los autores, 2002; © Edicions UPC, 2002.

- Otro ámbito de aplicación es el relacionado con la coordinación de los servicios hospitalarios. Por ejemplo, la coordinación necesaria para los transplantes de órganos [TARA00].

6.4.2 Planes de emergencia

Un plan de emergencia efectivo junto con una buena preparación del personal son los elementos esenciales para la gestión correcta de las emergencias. Desafortunadamente, evaluar ambos elementos es a menudo difícil en ausencia de emergencias reales.

Existen varios métodos para entrenar al personal y evaluar la capacidad de respuesta frente a las emergencias. Algunos de estos métodos se basan en una aproximación académica (videos, libros, conferencias) mientras otros son más experimentales (simulacros). No obstante, son varias las deficiencias asociadas a estos métodos:

- *Complejidad*: La gestión de las emergencias es una tarea compleja, que requiere la coordinación de numerosas personas y actividades.
- *Coste*: La realización de simulacros tiene generalmente un coste elevado y su alcance es, a menudo, limitado.
- *Falta de flexibilidad*: Con las herramientas tradicionales o mediante simulacros es muy difícil modificar o adaptar el escenario para evaluar diferentes hipótesis.

Los modelos de simulación ofrecen una solución potencial a muchos de estos problemas. Por esta razón, numerosos instituciones y países han acudido a estas técnicas para el análisis de los planes de emergencia. Las herramientas de simulación permiten el modelado de escenarios complejos con múltiples actividades que actúan de forma concurrente. El objetivo de estudio puede ser la evacuación de un solo edificio [THOM95] o la evacuación de una amplia zona geográfica.

6.4.3 Telecomunicaciones

Las redes de comunicación han experimentado un crecimiento acelerado paralelo a la popularización de Internet y de la telefonía móvil. El análisis de estos sistemas tiene interés para evaluar y comparar diseños alternativos. Las alternativas más comunes para evaluar el comportamiento de estos sistemas son el empleo de modelos analíticos, modelos de simulación o la medida directa de su comportamiento.

En la literatura se pueden encontrar numerosos artículos que abordan el problema de análisis y diseño de redes con la ayuda de técnicas de simulación. Es de destacar que la revista *Simulation* ha dedicado varias ediciones especiales a este tema en los últimos años (vol. 75, n.1, julio 2000; vol. 72, n. 5, mayo 1999; vol.70, n. 1, enero 1996; vol. 68, n. 1, enero 1997).

6.4.4 Procesos administrativos

Los recortes en los presupuestos a los que se ven sometidas las empresas, han puesto de manifiesto la necesidad de mejorar los procesos administrativos minimizando los costes asociados sin que ello repercuta en una degradación del servicio. Para ello, es necesario un examen riguroso de las distintas estrategias propuestas que permita cuantificar los beneficios que se obtendrían en la implantación de cada una de ellas.

© Los autores, 2002; © Edicions UPC, 2002.

gestión administrativa impuestos por los distintos departamentos ligados al mismo tiempo a presupuestos cada vez mas bajos.

Algunas ventajas que aporta la simulación en este campo son:

- Mejor comprensión del funcionamiento de los departamentos administrativos de las empresas.
- Mejor comprensión de los procesos burocráticos.
- Identificación del volumen de personal necesario, así como su funcionalidad para dar respuesta a las distintas actividades administrativas de cada unidad de negocio.

6.4.5 Actividades logísticas diversas

Son numerosos los campos en los que las técnicas de modelado y simulación se han empleado y se emplean para analizar y mejorar los procesos, por ejemplo:

- La industria de la *minería* está sujeta a fuertes presiones para incrementar su productividad, reducir los costes y reducir el impacto ambiental. Para conseguir estos objetivos, a menudo contradictorios, es necesario mejorar tanto los aspectos tecnológicos como los logísticos. La simulación puede ayudar a determinar que diseños o modos de operación son los más adecuados [RUNC99, LEBE98].
- Muchas empresas industriales y de transporte tienen que soportar costes muy elevados de *mantenimiento*. La simulación se puede emplear de forma efectiva para analizar la gestión del servició de mantenimiento [BOWM10, JONG97].
- En general, puede decirse que la mayor parte de actividades logísticas ligadas a la gestión de residuos tambien pueden ser simuladas [TRON00, BARN00], obteniendo beneficios tanto en los servicios de recogida de material, como en su posterior distribución.

6.5 Ejemplos

6.5.1 *Drive-Through simulation*

El caso de estudio que se presenta ha continuación ha sido reproducido con la autorización de los profesores E.J. Williams y M. Zottolo de la Universidad de Michigan, Deaborn [WILL01]. Agradecemos a ambos su oferta y autorización para presentar este caso.

El objetivo perseguido con el estudio es evaluar diferentes alternativas para mejorar el servicio en una oficina de una cooperativa bancaria (*credit union*) cercana a la universidad. Más concretamente, el estudio se centra en mejorar el servicio *drive-through* en los picos de los viernes por la tarde. Este servicio consiste en que los clientes pueden efectuar sus operaciones bancarias sin necesidad de salir de su vehículo. Con anterioridad al estudio, el servicio ofrecido era claramente inadecuado en estos periodos de tiempo, lo que provocaba colas excesivamente largas de espera, incidencias en el tráfico, quejas de los clientes y quejas de los propios trabajadores.

© Los autores, 2002; © Edicions UPC, 2002.

La oficina estudiada es la que realiza más operaciones *drive-through* del total de seis oficinas que dispone la cooperativa bancaria en el área metropolitana de Detroit. Trabaja seis dias a la semana: de 9:20 a 19:00 los días laborables y de 9:30 a 14:00 los sábados. Los picos de demanda ocurren los viernes desde las 16:00 hasta las 18:00. Dado que el estudio sólo tiene como objetivo mejorar el servicio en los picos de los viernes, el estudio de simulación se restringe al intervalo desde las 16:00 a las 18:00 de los viernes.

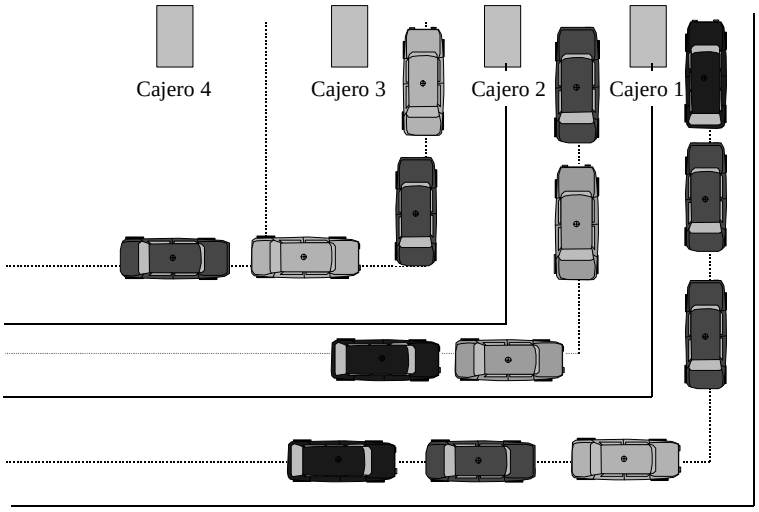


Figura 6.2 Estructura del drive-through

La sección de interés del *drive-through* está formada por ventanillas de servicio (4 cajeros) pero sólo tres líneas de entrada (figura 6.2). La línea más interior conduce a los cajeros 3 y 4. Dado que el terreno anexo ya está ocupado por otras empresas, la ampliación del área de acceso tuvo que ser descartada. Las líneas 1, 2 y 3 de entrada están separadas por guarda raíles, de manera que una vez un vehículo ha entrado en una de las colas no puede abandonarla ni cambiar de línea. Los cajeros 1 y 3 permanecían abiertos en los picos del viernes y el 4 cerrado. Una de las primeras tareas del estudio de simulación consistió en definir explícitamente el objetivo del estudio. En este caso particular el objetivo planteado fue analizar si la apertura de la línea 4 en los picos de los viernes incrementaría de forma significativa la calidad del servicio. Las estadísticas de interés escogidas para evaluar la calidad del servicio fueron el tiempo promedio en el sistema, el tamaño promedio y el tamaño máximo de las colas.

La siguiente etapa del proyecto consistió en obtener muestras significativas de los tiempos entre llegadas y de los tiempos de servicio. Las muestras se obtuvieron de tres viernes considerados representativos de la carga de trabajo normal en los picos de los viernes. Tal como se esperaba, la distribución de tiempos entre llegadas se ajustaba a una distribución exponencial. Sorprendentemente, el tiempo de servicio también se ajustaba a una distribución exponencial de media 2.3 minutos. La justificación se encontró después de una entrevista con los directivos. La mayoría de las operaciones permitidas en el *drive-through* son muy fáciles de efectuar y en general cada vehículo sólo efectuaban una operación. Con menos probabilidad, los vehículos efectuaban más de una operación u operaciones

© Los autores, 2002; © Edicions UPC, 2002.

que requerían el uso de recursos compartidos (ie. uso de fax para validación de firmas). Otro aspecto que fue necesario analizar consistió en averiguar si el tiempo de servicio dependía de la longitud de la cola. En muchos procesos de servicios, las personas encargadas de efectuar el servicio trabajan más deprisa si la cola es elevada y aprovechan los periodos con menos cola para relajarse y, por tanto, el tiempo de servicio es más largo. En este caso particular, los análisis por correlación de los tiempos de servicio demostró que el tiempo de servicio era independiente de la longitud de la cola. La razón principal que justifica la independencia de los datos es que los cajeros que trabajan en cada una de las ventanillas no ven la cola. Un último aspecto que se incluyó en el modelo fue la velocidad del vehículo en la cola dado que hay un tiempo muerto no despreciable desde que un vehículo finaliza el servicio hasta que el siguiente vehículo inicia el servicio en un mismo cajero.

Si no se tiene en cuenta la velocidad de los vehículos en las líneas y la selección de las líneas, se puede obtener una solución analítica empleando teoría de colas. Usando la notación de Kendall [KEND53], el modelo es del tipo M/M/3/30/∞. Los resultados iniciales obtenidos con el modelo de simulación verificado eran parececidos a los precedidos teóricamente, lo que permitió validar el buen comportamiento del modelo de simulación. Las lógicas discrepancias eran en parte originadas por las simplificaciones efectuadas en el modelo teórico. Un trabajo posterior de validación efectuado conjuntamente con los cajeros y los directivos permitió detectar un error significativo en el modelo. El modelo contemplaba que el cajero 1 se dedicaba exclusivamente a la línea 1, y de forma similar para los cajeros 2 y 3. En la realidad, los tres cajeros trabajan como un *pool* de recursos, cada uno de los cajeros puede trabajar con cualquiera de las líneas dado que los documentos pueden ser transferidos vía tubos neumáticos.

Para representar correctamente el estado del modelo a las 16:00 horas en la que empezaba la simulación, se optó por inicializar el modelo con 5 vehículos en cada una de las colas en lugar de definir un periodo de calentamiento previo al inicio del periodo de interés. Se optó por esta estrategia dado que se observó que en el inicio del periodo siempre había un mínimo de 5 vehículos en cada línea.

Se efectuaron 15 simulaciones para cada uno de los dos casos de interés. El primer caso corresponde a la situación inicial y en el segundo se tenía un cajero adicional. La tabla 6.1 presenta los resultados obtenidos, en los que se muestra una clara mejora con la configuración propuesta de 4 cajeros. Uno de los aspectos más valorados fue la escasa dispersión en los tiempos de permanencia en el sistema. En la configuración inicial, un cliente tanto podía permanecer 5 como 15 minutos en el sistema. La nueva configuración garantizaba a los clientes que el tiempo de permanencia en el sistema sería siempre muy parecido. Desaparece, por tanto, la impredectibilidad en el tiempo de permanencia, aspecto psicológicamente muy molesto para los clientes.

Número de cajeros	Tiempo medio en el sistema	Desviación estándar	Número de personas que no entran
3	9.08 min.	4.11 min.	1.6
4	4.18 min.	0.90 min.	0

Tabla 6.1 Resultados del estudio

© Los autores, 2002; © Edicions UPC, 2002.

6.5.2 Sistema de servicios a clientes

Este caso muestra un sistema complejo de servicio a clientes. Este tipo de sistemas los podemos encontrar en hospitales, centros médicos de consulta, oficinas de atención al ciudadano de la administración pública, parques de atracciones, y en sistemas más pequeños como supermercados, multicines, restaurantes, etc. El estudio de estos sistemas se dirige a la optimización de diferentes parámetros, como pueden ser la reducción o eliminación del tiempo de espera en las colas, el dimensionamiento de los espacios, el número de trabajadores necesarios, etc. Estos parámetros influyen en la sensación de frustración del cliente, en la reducción de costos, en la utilización de los equipamientos, etc. En definitiva, se desea una mejora en la calidad del servicio (QOS) con un coste mínimo. El siguiente caso muestra uno de estos sistemas.

En una importante ciudad europea es costumbre de sus ciudadanos ir a los baños públicos. El usuario de estos baños puede escoger entre una variedad de servicios disponibles, como son:

- Baño estándar
- Baño tonificante
- Baño de relajación
- Baño a presión
- Sauna
- Masaje de recuperación
- Masaje circulatorio
- Masaje de relajación

Los baños están divididos en seis zonas:

1. Recepción y venta de entradas
2. Vestuarios
3. Piscina para el baño estándar
4. Cabinas para baños especiales
5. Sauna
6. Cabinas de masajes

En la zona de recepción y venta de entradas se encuentran las taquillas, las cuales disponen de dos ventanillas con colas independientes donde el usuario compra su entrada. Los usuarios llegan a la zona de recepción siguiendo una distribución exponencial de media 0.5 minutos. De estos usuarios un 60% tienen algún tipo de abono que les permite entrar sin tener que comprar su entrada. El otro 40% restante hace cola en las taquillas; los clientes escogen en qué taquilla hacen cola según su tamaño y compran su entrada. Una vez el usuario está en disposición de entrar se dirige al control de acceso donde una persona se encarga de comprobar la validez de su entrada o abono.

Los vestuarios están divididos, hay un vestuario para hombres y otro para mujeres, pero comparten la zona de guardarropía. Cuando un usuario llega, entra primero en el guardarropía, donde un trabajador entrega al usuario una percha. El usuario se cambia de ropa y devuelve la percha al trabajador y éste le entrega un número identificador mediante el cual se le devolverá la ropa.

El 80% de los usuarios utilizan la piscina general para un baño estándar, un 50% utiliza la sauna, un 10% el baño de relajación, un 25% el baño tonificante, un 5% el baño a presión, un 7% el masaje de

© Los autores, 2002; © Edicions UPC, 2002.

recuperación, un 10% el masaje de relajación y un 6% el masaje circulatorio. Un usuario no utiliza más de un baño especial en una sesión ni más de un masaje y siempre pasa primero por la piscina general, los baños especiales y finalmente los masajes. Sólo se puede utilizar la sauna una vez ya se ha utilizado algún otro servicio.

El funcionamiento de la piscina y de la sauna son idénticos. En la entrada está un trabajador que comprueba la validez de la entrada o abono para el uso de este servicio. Para los masajes existen 10 cabinas, a cargo cada una de ellas por un masajista, que comprueba el masaje solicitado por el cliente y se lo realiza.

Para los baños especiales hay 10 cabinas para relajación y tonificación y 5 especiales para el baño a presión. Estas cabinas están a cargo de 3 trabajadores. En los baños de relajación y tonificación el trabajador comprueba el baño solicitado, ayuda al cliente y le prepara el baño; después queda el trabajador libre para atender a otro cliente. Para el baño a presión es necesario que el trabajador accione los mecanismos para el baño durante su duración.

En primer lugar se modelará bajo el formalismo de RdPC el subsistema de recepción y control de acceso. Este subsistema está formado por los siguientes elementos

- Colores:

  t: número de taquilla (taquilla 1=1, taquilla 2=2)
  a: posesión de abono (sin abono=0, con abono=1)
  s: sexo (hombre=1, mujer=2)
  n1: tamaño de la cola número 1
  n2: tamaño de la cola número 2
  ub: uso del baño general en la piscina (baño no realizado=1, baño realizado=0)
  ube: uso de baños especiales (tonificante=1, relajación=2, presión=3, ninguno=0)
  um: uso de masajes (circulación=1, relajación=2, recuperación=3, ninguno=0)
  idp: identificador de percha
  yab: el cliente ya ha tomado algún baño (sí=1, no=0)
  us: el cliente ya ha utilizado la sauna (sí=1, no=0)
  cp: cabina con baño a presión (con baño a presión=1, sin baño a presión=0)

© Los autores, 2002; © Edicions UPC, 2002.

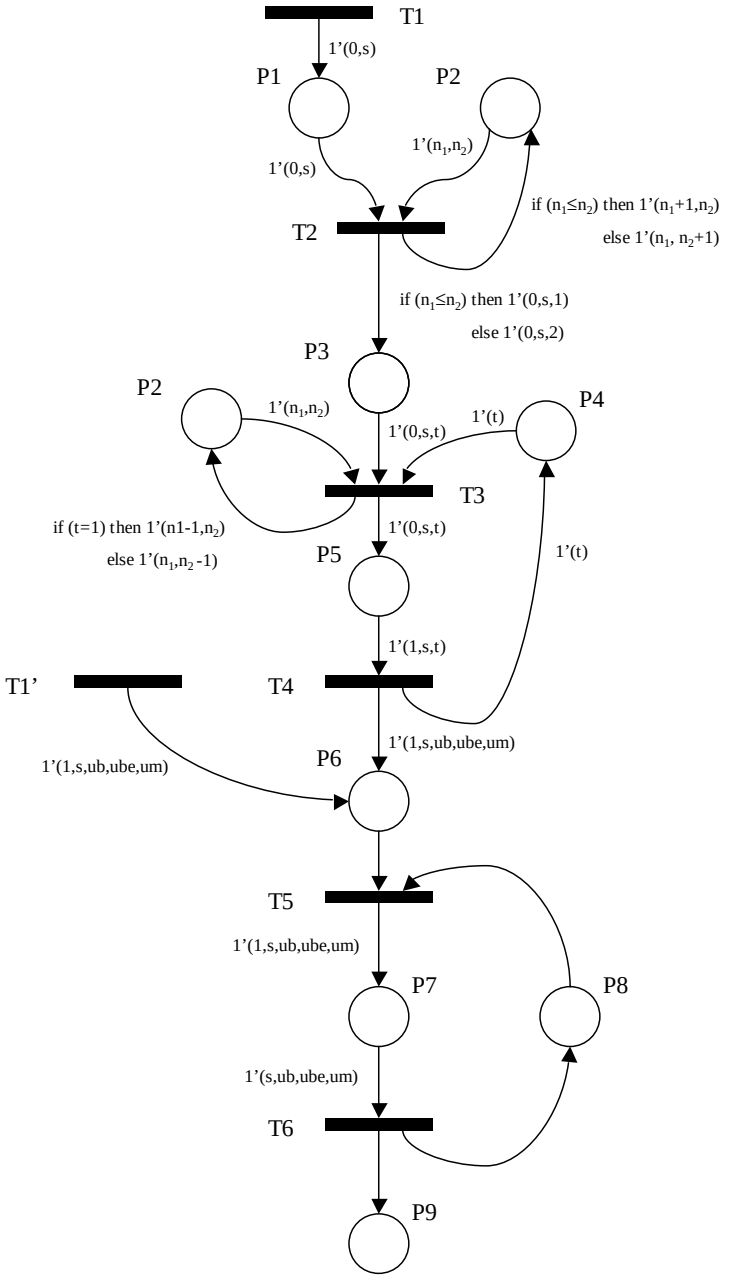


Figura 6.3 RdPC del subsistema de recepción y control de acceso

- Transiciones:

  T1: llegada de un cliente a la recepción; dependiendo de si posee o no abono (a=1) se dirige directamente a P6 (control de acceso) o a P1 (compra de entradas)

  T2: el cliente elige una taquilla en función del tamaño de las colas

© Los autores, 2002; © Edicions UPC, 2002.

T3: el cliente finaliza su espera y pasa a ser atendido en la taquilla
T4: finaliza la compra de las entradas o abonos
T5: el cliente entra en el control de acceso
T6: el cliente entra en los baños

- Lugares:

P1: decisión de la cola más corta (a,s)
P2: control de tamaños de las colas en las taquillas (n_1,n_2)
P3: espera en las colas de las taquillas (a,s,t)
P4: vendedor libre en taquilla (t)
P5: compra de la entrada o abono por el cliente (a,s,t)
P6: cola de espera para el control de acceso (a,s,ub,ube,um)
P7: control de acceso (a,s,ub,ube,um)
P8: controlador libre

El *token* del nodo lugar P2 contiene información del tamaño de las colas: cuando entra un cliente en la cola de la taquilla 1 el color n1 se incrementa y, cuando un cliente pasa a ser servido el color n1 se decrementa. La gestión del tamaño de la cola de la taquilla 2 se realiza de la misma forma. Cuando un cliente compra su abono este pasa a la cola de espera del control de acceso para acceder al recinto. En la figura 6.3 se muestra la RdPC de este subsistema.

Una vez dentro de los baños, el cliente se dirige a la zona de vestuarios. Los vestuarios también se pueden modelar como otro subsistema independiente. En este subsistema los elementos más importantes son:

- Transiciones:

T7: al cliente se le ha entregado la percha y entra en el vestuario
T8: el cliente sale del vestuario para entregar su ropa
T9: al cliente devuelve la percha al encargado
T10: el cliente sale de los vestuarios

- Lugares:

P9: una vez el cliente entra en el guardarropía, espera a que el trabajador esté libre y le entregue una pecha para colgar su ropa
P10: empleado de guardarropía libre
P11: el empleado del guardarropía le entrega una percha al cliente
P12: utilización del vestuario
P13: el cliente entrega al empleado del guardarropía la percha con su ropa para guardarla
P15: armario de ropa de los clientes

El funcionamiento de este subsistema es muy simple. Las personas llegan al guardarropía, un empleado les entrega una percha y los clientes se dirigen al vestuario donde se cambian de ropa. Cuando están a punto, vuelven al guardarropía y entregan la percha con la ropa al empleado y éste la guarda. En la figura 6.4 se ha representado este subsistema.

© Los autores, 2002; © Edicions UPC, 2002.

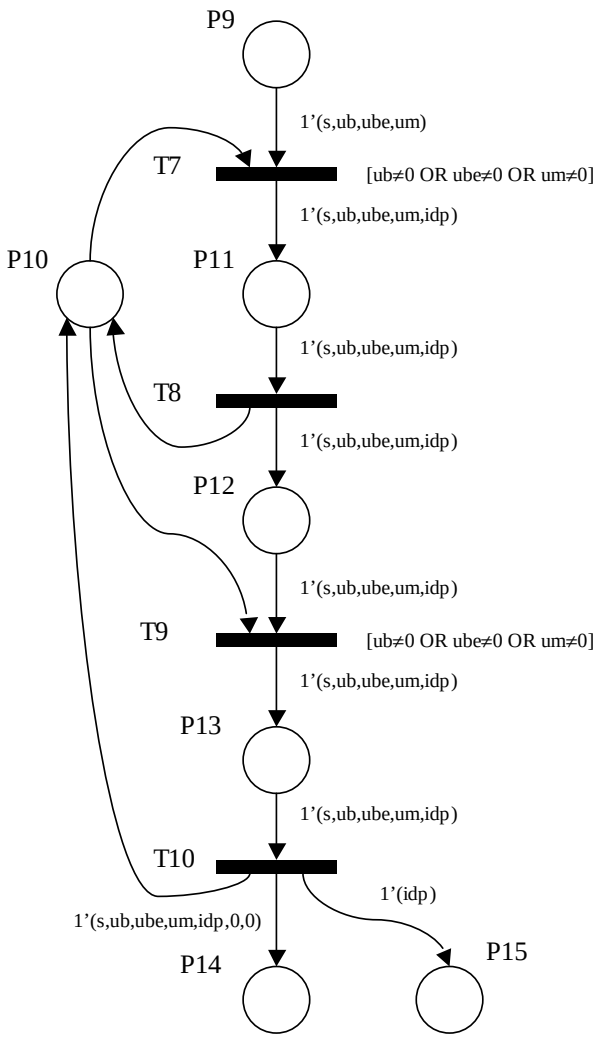


Figura 6.4 RdPC del subsistema de guardarropía y vestuarios

Una vez el cliente sale del vestuario decide a qué instalación se dirige. El siguiente subsistema representa el proceso de decisión. Para modelar esta parte se destacan los siguientes elementos:

- Transiciones:

  T11: el cliente se dirige a las cabinas de masajes
  T12: el cliente se dirige a la piscina general
  T13: el cliente vuelve a los vestuarios
  T14: el cliente se dirige a la sauna
  T15: el cliente se dirige a la zona de baños especiales

- Lugares:

© Los autores, 2002; © Edicions UPC, 2002.

P14: punto de decisión

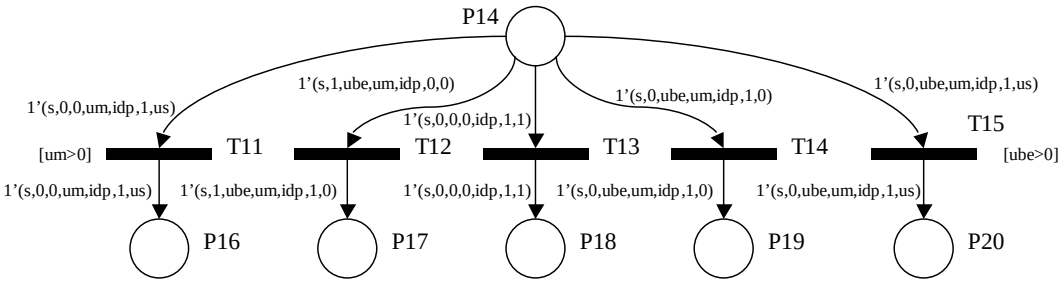


Figura 6.5 RdPC del subsistema de decisión de servicio

Cuando los clientes salen del vestuario se van a recibir los servicios solicitados. Para modelar esta decisión se ha creado un nodo lugar específico P14. Como más prioritario está el baño en la piscina, T12. Cuando se entra en la piscina se activa el color yab para indicar que el usuario ya ha tomado un baño y por lo tanto ya puede ir también a la sauna. Para utilizar los baños especiales T15 se tiene que haber utilizado antes el baño general o no haberlo solicitado (ub=0). Para ir a las cabinas de masajes T11 se tiene que haber utilizado algún baño o no haber solicitado estos servicios (los colores de baños igual a 0). Para utilizar la sauna se tiene que haber utilizado algún servicio anteriormente, yab=1 y no haber usado este servicio, us=0. Finalmente, si el cliente ya ha utilizado todos los servicios vuelve a los vestuarios y marcha del sistema T13.

La inclusión de este nodo lugar, que representa la decisión del próximo servicio que se utilizará, no es estrictamente necesaria, pero se ha decidido incluir este nodo para facilitar la comprensión de la RdPC del sistema.

El siguiente subsistema que se estudia es el baño general en la piscina. Este subsistema es muy parecido a un M/M/1 con la única diferencia que, cuando entra un cliente el color yab se activa (indicando que ya ha utilizado un servicio y por tanto podrá utilizar la sauna) y cuando sale del baño se desactiva el color ub (indicando que el cliente ya se ha bañado en la piscina).

En la figura 6.6 se representa la RdPC coloreada de este subsistema. Es de remarcar el significado de los nodos lugar:

P17: espera del empleado para el control de acceso
P21: empleado de control de acceso libre
P22: control de acceso
P23: baño

© Los autores, 2002; © Edicions UPC, 2002.

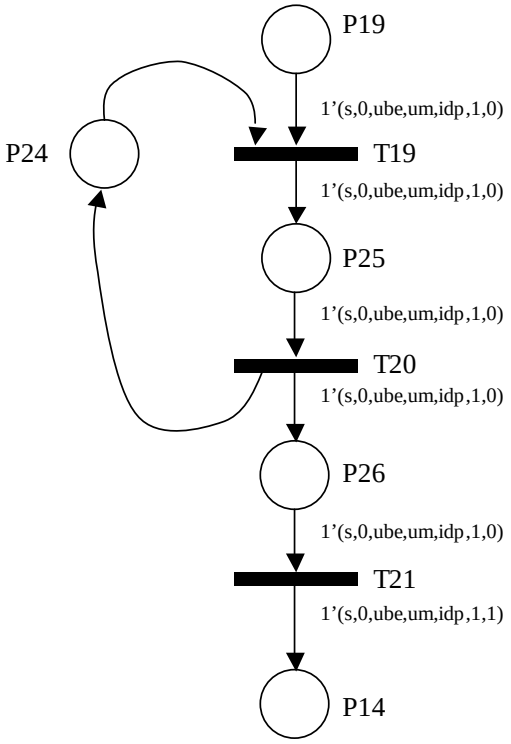


Figura 6.6 RdPC del subsistema de baño general

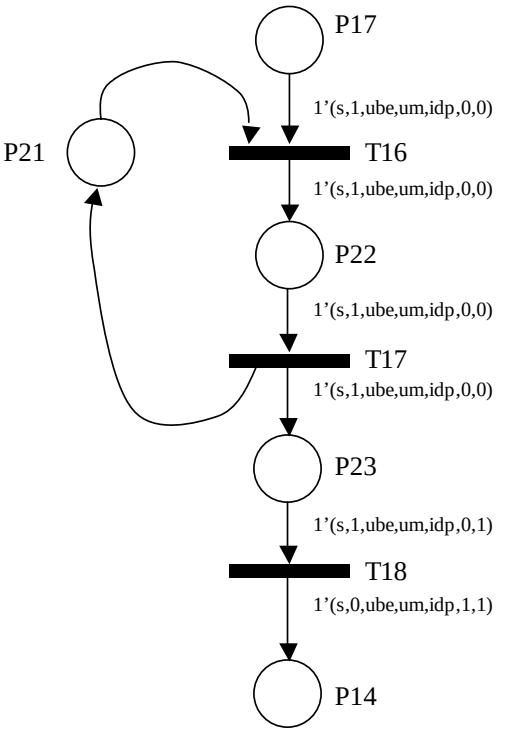


Figura 6.7 RdPC del subsistema de la sauna

El funcionamiento de la sauna es muy similar al del baño general, tal como muestra la figura 6.7.

Para la gestión de los baños especiales se necesita diferenciar las cabinas estándar de las cabinas de baño a presión. También dependiendo del tipo de baño que desee el cliente el operario de baños realizará unas tareas u otras.

© Los autores, 2002; © Edicions UPC, 2002.

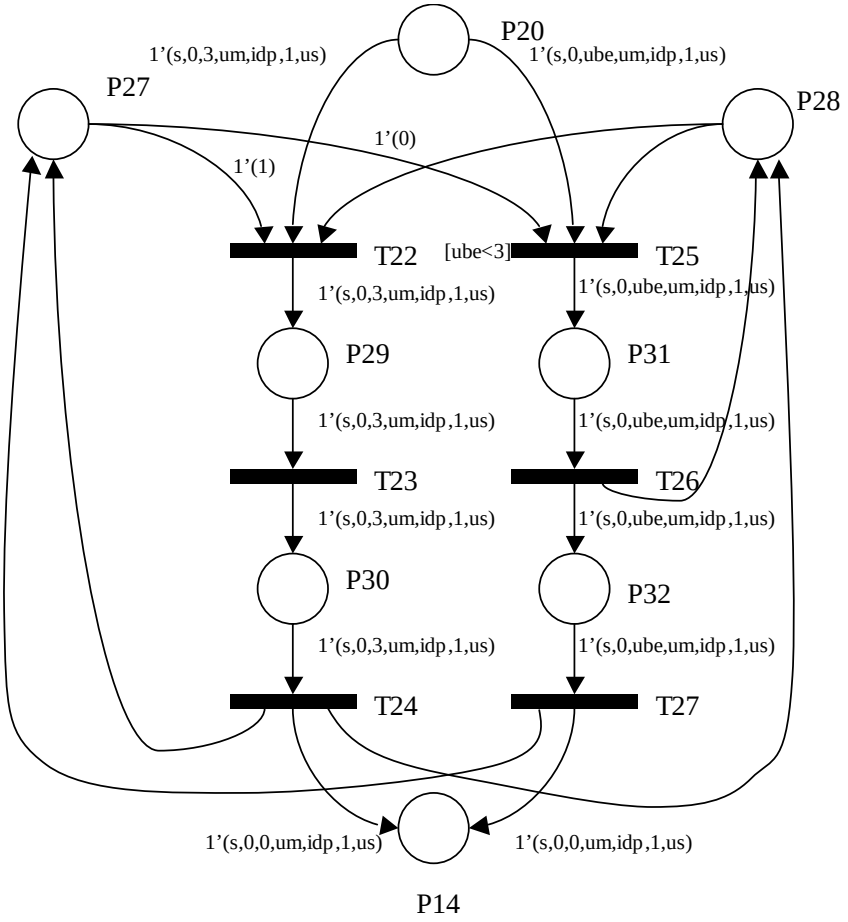


Figura 6.8 RdPC del subsistema de baños especiales

Los elementos más importantes en este subsistema son:

- Transiciones:

  T22: el usuario puede entrar en la cabina de baños a presión, con lo que existe una cabina libre y un trabajador libre
  T23: el usuario, ayudado por el trabajador, está a punto para el baño a presión
  T24: el baño a presión ha finalizado
  T25: el usuario puede entrar en la cabina para baños especiales (no presión), con lo que se cumple la existencia de una cabina libre y un trabajador libre.
  T26: el usuario, ayudado por el trabajador, está a punto para el baño solicitado
  T27: el baño especial, no a presión, ha finalizado

© Los autores, 2002; © Edicions UPC, 2002.

- Lugares:

  P20: el cliente espera a tener libre una cabina para el baño y a que un trabajador lo atienda
  P27: cabinas libres
  P28: trabajadores libres
  P29, P31: el trabajador ayuda al cliente para prepararse para el baño
  P30, P32: se realiza el baño

Cuando el usuario llega espera a que un trabajador lo atienda y a que haya una cabina acorde con el tipo de baño que él desea. Esta selección se hace mediante el color del tipo de baño ube y el color del tipo de cabina cp. Dependiendo del tipo de baño el trabajador ayuda al cliente a entrar en el baño o acciona los mecanismos. Finalmente se libera la cabina, o la cabina y el trabajador si era un baño a presión. Se asigna también el valor 0 al color ube para indicar que ya se ha realizado el baño especial.

Finalmente sólo queda por modelar el subsistema de la zona de masajes. Este subsistema es muy simple y funciona como un M/M/10 ya que las 10 cabinas y los 10 trabajadores se modelan en un mismo *token*. En la Figura 6.9 se puede observar la RdPC que modela este subsistema.

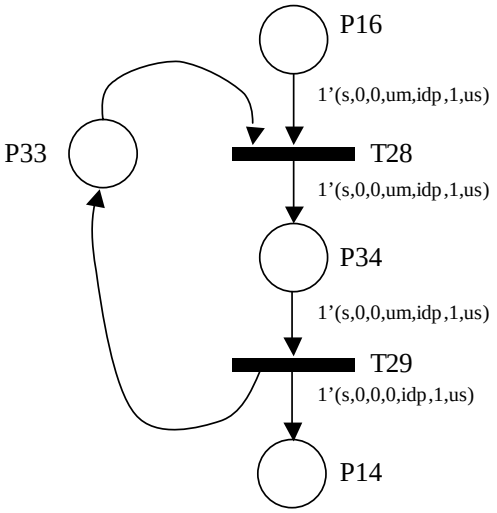


Figura 6.9 RdPC del subsistema de masajes

El diseño de estas RdPC se ha realizado intentando seguir el flujo que tienen los clientes dentro del sistema y separando todos los servicios que éstos utilizan. Estas RdPC se pueden diseñar utilizando aún más la potencia de este formalismo coloreando el servicio y uniendo todos los servicios en un único nodo lugar. También se pueden unir los recursos que representan a los trabajadores añadiendo un color que represente el tipo de servicio que realizan. Esta opción se ha descartado para mejorar la legibilidad del modelo. El lector interesado en este tipo de formalismo puede encontrar un ejercicio, al final del capítulo, donde realizar estas mejoras.

Cuando un usuario ha realizado sus baños o masajes, este vuelve a los vestuarios. Este caso carece de interés ya que es idéntica a la entrada a vestuarios pero en orden inverso.

© Los autores, 2002; © Edicions UPC, 2002.

6.5.3 Modelado de la asignación de barcos a atraques

A continuación se describirán algunos de los principales aspectos en el desarrollo de un simulador para estimar la capacidad de movimiento de material de un puerto, en función de los patrones de llegada de barcos deterministas, del espacio disponible para el atraque de barcos, así como del número y rendimiento de grúas portuarias asignadas a cada terminal. El caso particular de estudio que se presenta ha sido desarrollado conjuntamente J. Canudas, D. Verges y S. Ametller de la empresa SENER[11].

El modelo de simulación ha sido desarrollado siguiendo el formalismo de RdPC para facilitar su mantenimiento atendiendo a futuros objetivos. En la actualidad se limita a describir a nivel macro las distintas actividades que se realizan en la vía terrestre y, pormenoriza las distintas secuencias de acciones que realizará un barco desde su llegada al puerto hasta su salida, las cuales pueden resumirse en:

- T1: proceso de llegada del barco al puerto
- T2: solicitud de permiso de atraque del barco en un terminal
- T3: espera del barco hasta que se le permite realizar la operación de atraque
- T4: asignación de grúas portuarias para realizar las operaciones de manipulación de material
- T5: descarga del barco en el almacén terrestre
- T6: carga de material desde el almacén terrestre al barco
- T7: solicitud de partida del barco
- T8: espera del barco hasta que se permite acceder al área de paso de entrada/salida del puerto
- T9: salida del puerto

Los eventos que describen dicha secuencia de actividades corresponden al siguiente conjunto de transiciones:

$$T=\{T1,T2, T3,T4, T5,T6, T7,T8, T9\}.$$

El modelo considera el puerto como un conjunto de recursos los cuales se encuentran en alguno de los siguientes nodos lugar:

$P = \{B, C, A, M\}$

donde:

– B: representa la información asociada a los barcos. El conjunto color utilizado para describir el estado de cada barco es: Σ = { b_id, b_ti, b_es, b_pa, b_idm, b_m, b_o, b_cc, b_cd, b_l, b_g, b_gd, b_p, b_rend, b_ida, b_ta, b_ts, b_tca, b_vta, b_pca}, donde:

  - b_id: especifica el identificador del barco

[11] Financiado parcialmente gracias a la ayuda recibida del Ministerio de Ciencia y Tecnología (referencia COO1999AX032).

© Los autores, 2002; © Edicions UPC, 2002.

- b_ti: especifica el tiempo de llegada del barco al puerto
- b_es: especifica el estado en el que se encuentra el barco: 0= en espera de asignación de atraque # 1= en espera de proceso de atraque # 2= proceso de atraque # 3= asignación de grúas # 6= sin grúas # 7= en espera de proceso de desatraque # 8= proceso de desatraque # 9= proceso de partida.
- b_pa: especifica el patrón de llegada del barco: 1= conocido # 2= aleatorio
- b_idm: especifica el módulo destino del barco en aquellos casos que se trate de un barco con patrón de llegadas determinista. En caso contrario, este atributo especifica el módulo destino del barco aleatorio
- b_m: especifica la mercancía que transporta el barco
- b_o: especifica el tipo de operación que debe realizar el barco: 1= carga # 2= descarga #3= carga y descarga
- b_cc: especifica en toneladas la cantidad de mercancía que debe cargar el barco
- b_cd: especifica en toneladas la cantidad de mercancía que debe descargar el barco
- b_l: especifica la eslora del barco
- b_g: especifica el número de grúas con las que es posible trabajar con el barco
- b_gd: especifica el número de grúas que tiene a su disposición
- b_p: especifica la prioridad con que debe ser atendido el barco
- b_rend: especifica el rendimiento total de las grúas que tiene a su disposición
- b_ida: especifica el identificador del área donde va a realizar las operaciones
- b_ta: especifica el momento de tiempo de atraque del barco
- b_ts: especifica el tiempo previsto de salida del barco del puerto
- b_tca: especifica el último tiempo que el barco realizó una operación de carga y descarga
- b_vta: especifica el tipo de aviso que ha activado la asignación del barco al módulo
- b_pca: especifica el número de periodos de tiempo (en minutos) de simulación de las operaciones de carga y/o descarga que le faltan al barco para terminar todas las operaciones de carga y descarga

– C: indica el estado de cada grúa portuaria, así como toda la información necesaria para su correcto uso. El conjunto color utilizado para describir el estado de las distintas grúas es Σ = { c_id, c_rend, c_tc, c_e, c_mc, c_m, c_idb}, donde:
  - c_id: especifica el identificador de la grúa

© Los autores, 2002; © Edicions UPC, 2002.

- c_rend: especifica el rendimiento de la grúa (toneladas/hora)
- c_tc: especifica la mercancías que son compatibles con la mercancía
- c_e: especifica el estado de la grúa: 1: libre # 2: ocupada # 3: trabajando
- c_mc: lista de módulos a los que puede desplazarse la grúa
- c_m: módulo en el que se encuentra la grúa
- c_idb: especifica el barco que está utilizando la grúa

– A: representa la información asociada a cada una de las áreas de la vía terrestre donde se almacena el material.. El conjunto color utilizado para describir el estado de las distintas áreas de almacén es: Σ = { a_id, a_m, a_me, a_cp, a_cpae, a_cpai }, donde:
  - a_id: especifica el identificador del área de almacenaje
  - a_m: especifica el identificador del módulo que puede acceder a dicha área de almacenaje
  - a_me: especifica la mercancía que puede acceder a dicha área de almacenaje
  - a_cp: especifica la capacidad del área de almacenaje
  - a_cpae: especifica la capacidad actual del área de almacenaje de exportación
  - a_cpai: especifica la capacidad actual del área de almacenaje de importación
– M: representa la información asociada a cada uno de los módulos en los cuales se descompone el puerto. El conjunto color utilizado para describir el estado de los distintos módulos Σ ={ m_id, m_pe, m_idm, m_pep, m_pec, m_l, m_la, m_m, m_h, m_ida, m_bm, m_tm }, donde:
  - m_id: especifica el identificador del módulo
  - m_pe: especifica la política de explotación del módulo
  - m_idm: especifica el identificador del muelle al que pertenece el módulo
  - m_pep: especifica la priorización en la política de explotación
  - m_pec: especifica la cota de espera máxima que puede demorar un barco conocido por culpa de estar atendiendo a un barco aleatorio
  - m_l: especifica la longitud total de atraque del módulo
  - m_la: especifica la longitud de atraque libre del módulo
  - m_m: especifica las mercancías que son compatibles con el módulo
  - m_h: especifica los tramos horarios (Hi-Hf) durante los cuales hay manos para realizar las operaciones. Este color deberá codificarse como un vector (M,D,HI,HF): M indica

© Los autores, 2002; © Edicions UPC, 2002.

la mercancía, D indica el día de la semana, y Hi y Hf indican la hora de inicio y la hora final

- m_ida: especifica el identificador del área de almacenaje a la que puede acceder el módulo. Este color deberá codificarse como un vector (M,A): M indica la mercancía y A el área de almacenaje.
- m_bm: especifica si hay algún barco realizando operaciones: 0: No # 1. Sí
- m_tm: especifica el tiempo de maniobras para atracar o desatracar en el módulo

Con el objetivo de poder evaluar los beneficios que se obtienen ante diferentes políticas de priorización de barcos para realizar la operación de atraque, el modelo de simulación debe ser parametrizado atendiendo a las siguientes políticas de explotación:

1. Cola FIFO. El primer barco que atraca en un módulo es el que lleva más tiempo en espera para acceder a dicho módulo.
2. Cola semi FIFO. Es una política similar a la FIFO en cuanto a que en primera instancia se prioriza el barco que lleva más tiempo esperando atracar en un determinado módulo si y sólo si la longitud de eslora es inferior a la longitud disponible en el módulo; sin embargo, cuando por razones de longitud de eslora, de utillaje disponible y de capacidad de los depósitos de almacenaje, el primero no tiene cabida en el módulo y el segundo o posteriores si, en este lugar no se respeta la norma FIFO y atraca el primer barco que cumpla las condiciones para atracar.
3. Cola por prioridades. Los barcos que atracarán en primer lugar en el módulo serán los que tengan la prioridad mayor. En caso de empate va a regir la norma semi FIFO anteriormente expuesta.

A continuación, se muestran las RdPC correspondientes al evento T2 en función de las tres políticas de priorización expuestas.

a1: 1'(**v_mid**,pe,idm,pep,pec,m_l,m_la,**v_m**,h,ida,**1,**tm)
a2: 1'(**v_mid**,pe,idm,pep,pec,m_l,m_la-b_l,**v_m**,h,ida,**2,**tm)

a3: 1'(b_id,ti,0,1,**v_mid**,**v_m**,o,cc,cd,b_l,g,gd,p,rend,ida,ta,ts,tca,vta,pca)
a4: 1'(b_id,ti,**1**,1,**v_mid**,**v_m**,o,cc,cd,b_l,g,gd,p,rend,ida,ta,ts,tca,vta**,**pca)

G (T2)=[(pe=1)&&(pep=1) &&
(ti=Min{(*,ti,0,1,v_mid,*,*,*,*,*,*,*,*,*,*,*) ? B}) &&
(b_l<m_la)]

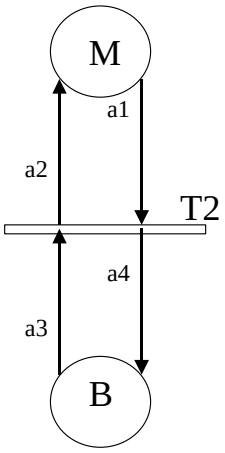


Figura 6.10 Permiso de atraque de un barco en un módulo específico si se sigue una política FIFO

© Los autores, 2002; © Edicions UPC, 2002.

En la figura 6.10 se han representado las precondiciones y los efectos sobre los atributos del barco y del módulo en el cual se permite el atraque, cuando la priorización se realiza siguiendo una política FIFO. Como puede observarse, la asignación de atraque tan sólo puede realizarse si existe un barco conocido en el fondeo exterior y demanda atracar en el módulo v_mid, tiene una eslora inferior al espacio libre de la alineación (Guarda: b_l < m_la), transporta la mercancía v_m y es el barco que lleva más tiempo esperando atracar en dicho módulo (ti = Min(*,ti,0,1,v_mid,...). Asimismo, para disparar las transiciones de asignación de barcos es necesario que no haya ningún barco en el módulo realizando operaciones de maniobra (m_bm=1). Obsérvese que no se evalúa el número de grúas disponibles para trabajar con el barco al ser la prioridad de cola FIFO.

En la figura 6.11 se han representado las precondiciones y los efectos sobre los atributos del barco, del módulo en el cual se permite el atraque, así como sobre el estado de las grúas portuarias cuando la priorización se realiza según una política semi FIFO. Como puede observarse, la asignación de atraque tan sólo puede realizarse si existe un barco conocido en el fondeo exterior y demanda atracar en el módulo v_mid, tiene una eslora inferior al espacio libre de la alineación (Guarda: b_l < m_la), transporta la mercancía v_m, es el barco que lleva más tiempo esperando atracar en dicho módulo (ti = Min(*,ti,0,1,v_mid,...)), y existe como mínimo una grúa disponible para atender las operaciones de manipulación de la carga. Nótese, que el valor n asignado al atributo b_gd corresponde al número de grúas disponibles (ver expresión de arco a7 y a8) en el módulo v_mid, que pueden manipular la mercancía de tipo v_m, limitando el valor n al número máximo de grúas que pueden trabajar concurrentemente con el barco (atributo g). Asimismo, para disparar las transiciones de asignación de barcos es necesario que no haya ningún barco en el módulo realizando operaciones de maniobra (m_bm=1).

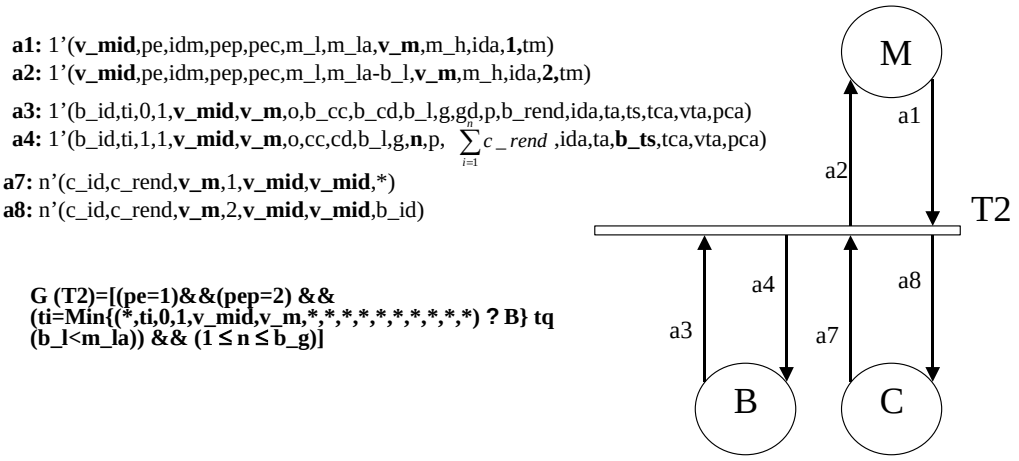


Figura 6.11 Permiso de atraque de un barco en un módulo específico si se sigue una política semi FIFO

© Los autores, 2002; © Edicions UPC, 2002.

6.5.4 Célula flexible de fabricación

Aunque algunos autores [QMIP93] indican que para modelar un sistema de eventos discretos es necesario considerar todos los posibles estados, así como los eventos que rigen la evolución del sistema como paso previo a su formalización en RdPC, en opinión de los autores, el número de estados que puede alcanzar los sistemas logísticos o bien de producción puede ser muy elevado, por lo que se recomienda simplemente determinar todos aquellos eventos cuya aparición repercute en un cambio sobre las variables de interés del sistema.

En la figura 6.12 se ha representado una célula flexible integrada por los siguientes subsistemas:

1. Una unidad de carga/descarga, integrada por un robot (Ro=1), y dos almacenes (i=1,i=6).
2. Dos células de trabajo, integrada cada una de ellas por un robot (Ro=3 y Ro=5), un *stock* (i=3, e i=5), una máquina CNC (j=10 y j=11) y una máquina ensambladora (p=3 y p=5) con tres posiciones para cargar los tres tipos de piezas.
3. Una unidad de transporte que consiste en una cinta transportadora con 8 posiciones (y=1...8) donde parar los 5 *palets*.

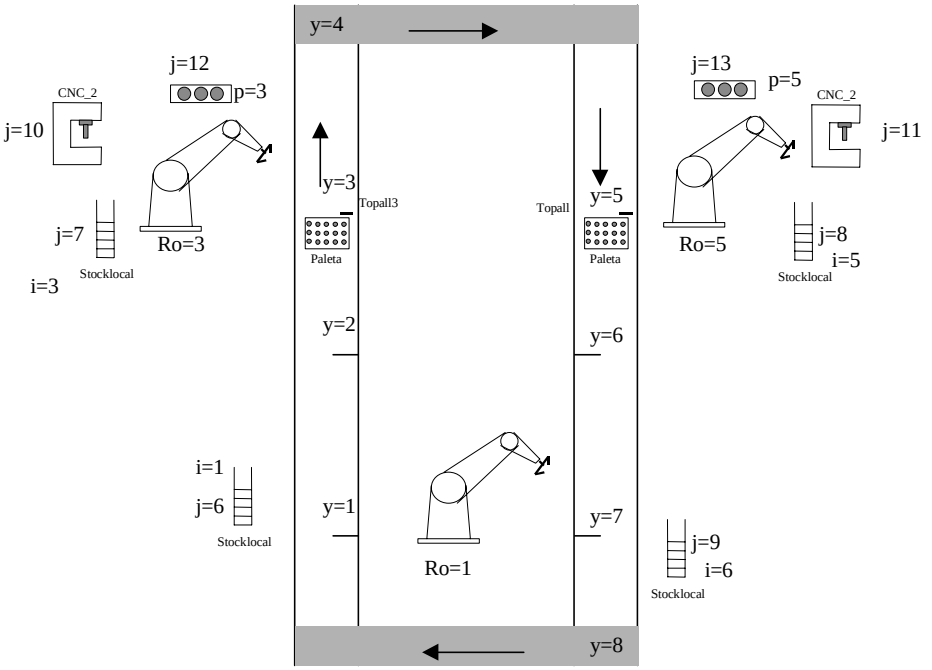


Figura 6.12 Célula flexible de fabricación

El sistema dispone de un almacén de materia primera (i=1) con 3 tipos de piezas que deben ser procesadas en las máquinas CNC (tan sólo las de tipo 2 y 3 que, una vez procesadas se identifican como piezas de tipo 4 y 5 respectivamente) y, finalmente, ensambladas en piezas de tipo 6 (piezas de tipo 1 + piezas de tipo 2 + piezas de tipo3) o de tipo 7 (piezas de tipo 1 + piezas de tipo3).

© Los autores, 2002; © Edicions UPC, 2002.

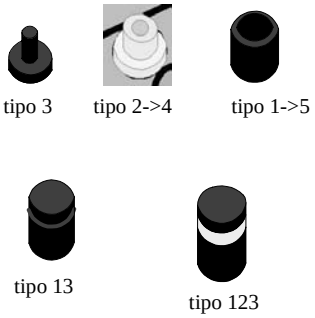


Figura 6.13 Tipos de piezas

Sobre dicho sistema de producción, los únicos eventos que pueden provocar un cambio de estado son los descritos en el siguiente conjunto transición:

$T=\{T_1,T_2, T_3,T_4, T_5,T_6, T_7,T_8, T_9,T_{10}, T_{11},T_{12}, T_{13},T_{14}, T_{15}\}$

donde:

- T_1: movimiento de un *palet* entre dos posiciones consecutivas de la cinta
- T_2: carga de piezas del almacén de materia prima en el *palet*
- T_3: transporte de una pieza del *palet* al *stock* de una de las células de trabajo
- T_4: transporte de una pieza del *palet* a la máquina ensambladora de una de las células de trabajo
- T_5: transporte de una pieza del *stock* de una de las células de trabajo al *palet* situado en el área de trabajo de la célula de trabajo
- T_6: transporte de una pieza del *stock* de una de las células de trabajo a la máquina ensambladora
- T_7: transporte de una pieza del *stock* de una de las células de trabajo a la máquina CNC
- T_8: finalización de la operación de procesamiento en la máquina CNC
- T_9: transporte de una pieza del *palet* a la máquina CNC de una de las células de trabajo
- T_{10}: transporte de una pieza del *palet* al almacén de salida
- T_{11}: transporte de una pieza procesada de la máquina CNC al stock de una de las células de trabajo
- T_{12}: transporte de una pieza procesada de la máquina CNC al *palet* situado en el área de trabajo de una de las células de trabajo
- T_{13}: transporte de una pieza de la máquina CNC a la máquina ensambladora de la célula de trabajo
- T_{14}: transporte de una pieza finalizada de la máquina ensambladora al *stock* de la célula de trabajo
- T_{15}: transporte de una pieza finalizada de la máquina ensambladora al *palet* situado en el área de trabajo de la célula de trabajo

© Los autores, 2002; © Edicions UPC, 2002.

Para poder codificar correctamente dichos eventos en un entorno de simulación será necesario dotar a las entidades que fluyen en el sistema, así como a los recursos compartidos, de ciertos atributos que les permitan mantener actualizada la información sobre el estado presente de la célula flexible. El siguiente conjunto color especifica los distintos atributos que deben ser correctamente codificados en un entorno de simulación.

Σ = { x, y, z, j, k, i, n, p, b, Pa, Ci, Ro, Pe, St, Mb, Mn, Cn, Pr}

donde:

x: identificador del *palet*
y: posición del *palet* (entero 1...8)
z: nº de posiciones libres en el *palet* (entero 1...12)
j: almacén o *stock* donde se encuentra la pieza (entero 1...13)
k: tipo de pieza o posición en la unidad de montaje
i: identificador del *stock*
n: nº espacios libres en el *stock*, o nº espacios ocupados en la ensambladora
p: unidad ensambladora
b: estado: libre, ocupado
Pa: información del *palet*. Producto cartesiano $x \times y \times z$
Ci: posición libre en la cinta transportadora
Ro: robot libre en estación
Pe: información de la pieza. Producto cartesiano $k \times j$
St: información del *stock*. Producto cartesiano $i \times n$
Mb: espacios ocupados en la ensambladora. Producto cartesiano $n \times p$
Mn: información de espacio libre en la ensambladora. Producto cartesiano $k \times p$
Cn: información de la máquina CNC. Producto cartesiano $p \times b$
Pr: información de la pieza en la máquina CNC. Producto cartesiano $p \times k$

Finalmente, los nodos lugar que van a permitir representar todos los recursos y almacenes que van a ser necesarios para el desarrollo del modelo de simulación, junto con los atributos asociados a las entidades almacenadas en cada nodo lugar se describe en el siguiente conjunto lugar.

P = {A, B, C, D, E, F, G, H}

donde:

A: representa la información asociada a los *palets*: identificador del *palet*, posición en la cinta, nº de espacios libres. Color utilizado Pa
B: indica las posiciones libres en la cinta. Color utilizado Ci
C: indica el estado de cada robot. Color utilizado Ro
D: representa la información asociada a cada pieza. Color utilizado Pe: tipo de pieza=(1...3: tipo de pieza original, 4: tipo 2 procesada, 5: tipo 1 procesada, 6: pieza ensamblada 123, 7: pieza ensamblada 1,3), y posición (1...5: en *palet*, 6: stock_0, 7: stock_1, 8: stock_2, 9: stock_3, 10: CNC1, 11: CNC2, 12: ensambladora_1, 13: ensambladora_2

© Los autores, 2002; © Edicions UPC, 2002.

E: representa información asociada al *stock*. Color utilizado St: identificador (1,3,5,6) y nº de espacios libres (1...20)

P: información asociada a la unidad de ensamblado. Color utilizado Mb: nº de piezas, unidad de ensamblado (3: Unidad_1, 5: Unidad_2)

F: información asociada a la unidad de ensamblado. Color utilizado Mn: posiciones libres (3: posición P1, 4: posición P2, 5: posición P3), unidad de ensamblado(3: unidad_1, 5: unidad_2)

G: información asociada a la máquina CNC. Color utilizado Cn: unidad (3: unidad_1, 5: unidad_2), y estado (0: libre, 1:ocupada)

H: información asociada a la pieza que se está procesando en una unidad CNC. Color utilizado Pr: unidad (3: unidad_1, 5: unidad_2), y tipo de pieza (2: tipo de pieza 2 procesándose, 3: tipo de pieza 3 procesándose, 4: tipo de pieza 2 procesada, 5: tipo de pieza 1 procesada)

A continuación se describen cada una de las redes de Petri coloreadas que formalizan los eventos que pueden aparecer en la célula flexible de fabricación bajo una aproximación modular.

T1 Movimiento de un palet de una posición de la cinta transportadora a la siguiente posición

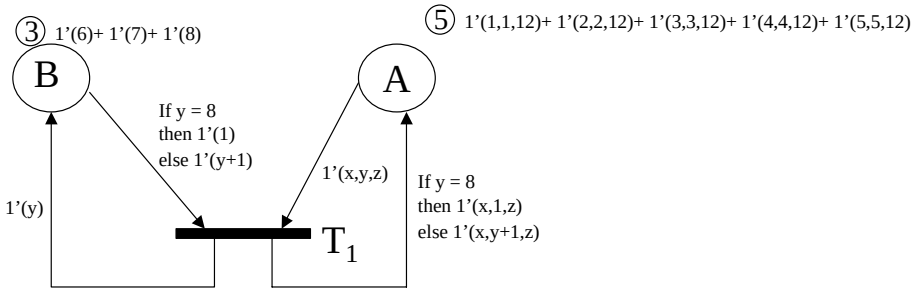


Figura 6.14 Red de Petri del subsistema cinta transportadora

Como puede observarse, en el lugar B se indican las posiciones libres de la cinta transportadora (inicialmente posiciones 6, 7 y 8), y en el lugar A se indica la información relativa a los *palets* (inicialmente hay 5 *palets* totalmente vacíos en las posiciones 1, 2, 3, 4 y 5 de la cinta transportadora). Para que pueda producirse una operación de avance de un *palet* de una posición a la siguiente, es necesario que la posición siguiente se encuentre vacía (expresión de arco de B a T1: 1'(y+1)) donde y corresponde a la posición actual del *palet* que va a realizar el movimiento (expresión de arco de A a T1: 1'(x,y,z). Como consecuencia de dispararse la transición T1, se indica que la posición libre de la cinta es la posición donde se encontraba el *palet* (expresión de arco de salida de T1 a B: 1'(y)), y que el *palet* ya se encuentra en la siguiente posición (expresión de arco de salida de T1 a A: 1'(x,y+1,z)).

Nótese que puesto que tan sólo hay 8 posiciones en la cinta transportadora, la siguiente posición a la posición nº 8 no es la posición nº 9, sino que vuelve a ser la posición inicial (nº 1).

© Los autores, 2002; © Edicions UPC, 2002.

T2 y T10 Unidad de carga/descarga

Para que pueda producirse una operación de carga (transición T2) desde el almacén de materia prima hasta el *palet*, es necesario que un *palet* se encuentre en la posición inicial (expresión de arco A→T2: 1'(x,1,s)), exista alguna pieza en el almacén de entrada (expresión de arco D→T2: 1'(k,6) donde el valor 6 indica el almacén de materia prima), y que como mínimo haya una posición libre en el *palet* (expresión de guarda s>0, donde s es el color que indica el número de posiciones libres del *palet* escogido). Adicionalmente, para que pueda producirse la operación de carga, se comprueba que el manipulador se encuentre libre (expresión de arco C→T2: 1'(1), donde el valor 1 hace referencia al robot de carga/descarga. Como consecuencia de haber realizado una operación de carga el número de posiciones libres en el *palet* se decrementa en 1 (expresión de arco T2→A: 1'(x,1,s-1)), y se actualiza la información de la entidad que representa la pieza para indicar que se encuentra en el *palet* (expresión de arco T2→D: 1'(k,x) donde x corresponde al identificador del *palet*).

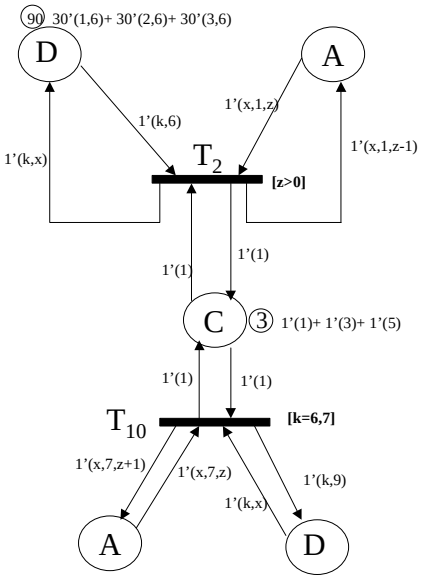


Figura 6.15 Subsistema unidad de carga

En este ejemplo puede observarse que, aunque la mayor parte de simuladores comerciales ofrecen bloques funcionales para describir cintas transportadoras (*conveyors* en Arena), las cuales ya gestionan de modo automático el cambio de posición de un *palet*, el modelador debe tener mucho cuidado en codificar los cambios de estado en los *palets* (información sobre las piezas que transporta, así como posiciones libres en cada *palet*), los cuales pueden aparecer como efectos de los eventos de carga/descarga en las distintas unidades. Así pues, la formalización de dichos eventos en RdPC es imprescindible para una correcta codificación del modelo de simulación en un entorno comercial y garantizar su futura reutilización.

© Los autores, 2002; © Edicions UPC, 2002.

De modo análogo puede comprobarse que para una operación de descarga (transición T10) en la posición 8 de la cinta transportadora, se requiere que haya en un *palet* en dicha posición y que, como mínimo, tenga una pieza.

T3 Transporte de una pieza del palet *al* stock *local*

Para que pueda producirse una operación de carga de una pieza que se encuentra en el *palet* al almacén local asociado a alguna de las dos celdas de trabajo, es necesario que se cumplan las siguientes condiciones:

- Guarda [y=3 || y=5]: el *palet* debe estar situado en la posición y=3 (si se va a descargar la pieza en la primera celda de trabajo) o bien en la posición y=5 en el caso que se descargue la pieza en el almacén local de la segunda celda de trabajo. El valor del color y se obtiene del arco de salida del nodo A, que recoge toda la información asociada a los *palets*.
- Guarda [n > 0]: debe haber espacio libre en el almacén. El valor del color n que se obtiene del arco de salida del nodo E indica el número de posiciones libres en el almacén, y en el caso que se produzca el evento, dicho valor se decrementa en 1 (arco de entrada al nodo E).
- Guarda [k<= 5]: tan sólo se permite descargar piezas que todavía necesitan una operación. El valor del color k que se obtiene del arco de salida del nodo D, indica el tipo de pieza.

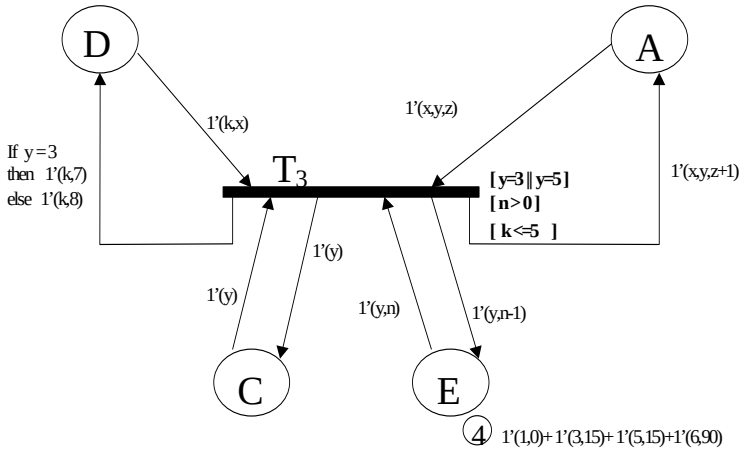


Figura 6.16 Transporte de palet *a* stock

Adicionalmente, para que pueda producirse la operación de transporte, se comprueba que el manipulador se encuentre libre (expresión de arco $C \rightarrow T_3$: 1'(y), donde el valor y corresponde al robot situado delante del *palet*) y que la pieza que se quiere transportar se encuentre en el *palet*, lo cual se indica mediante el color x del arco de salida del nodo D, cuyo valor debe coincidir con el valor x del elemento utilizado del nodo A. Como consecuencia de haber realizado una operación de carga, el número de posiciones libres en el almacén se decrementa en 1 (expresión de arco $T_3 \rightarrow E$: 1'(y,n-1)), se incrementa el número de posiciones libres en el *palet* (expresión de arco $T_3 \rightarrow A$: 1'(x,y,z+1)), y se

© Los autores, 2002; © Edicions UPC, 2002.

actualiza el atributo que indica la nueva posición donde se encuentra la pieza (expresión de arco $T_3 \rightarrow D$: 1'(k,7) || 1'(k,8)).

T4 Transporte de una pieza del palet *a la máquina ensambladora.*

Para que pueda producirse una operación de carga de una pieza que se encuentra en el *palet* a la máquina de ensamblado asociada a alguna de las dos celdas de trabajo, es necesario que se cumplan las siguientes condiciones:

- Guarda [y=3 || y=5]: el *palet* debe estar situado en la posición y=3 (si se va a descargar la pieza en la primera celda de trabajo) o bien en la posición y=5 en el caso que se descargue la pieza en la máquina ensambladora de la segunda celda de trabajo. El valor del color y se obtiene del arco de salida del nodo A, que recoge toda la información asociada a los *palets*.
- Guarda [k=3 || k=4 || k=5]: tan sólo se permite alimentar la máquina de ensamblado con piezas ya procesadas. El valor del color k que se obtiene del arco de salida del nodo D indica el tipo de pieza.

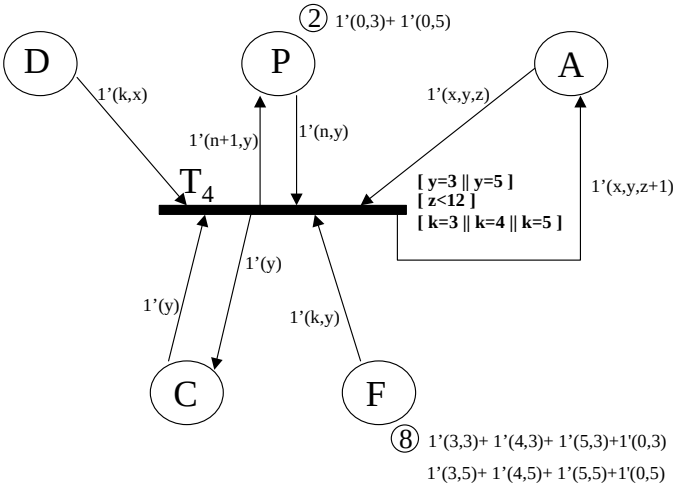


Figura 6.17 Transporte de palet a ensambladora

Adicionalmente, para que pueda producirse la operación de transporte, se comprueba que el manipulador se encuentre libre (expresión de arco $C \rightarrow T_4$: 1'(y), donde el valor y corresponde al robot situado delante del *palet*) y que la pieza que se quiere transportar se encuentre en el *palet*, lo cual se indica mediante el color x del arco de salida del nodo D, cuyo valor debe coincidir con el valor x del elemento utilizado del nodo A. Para evitar que se puedan cargar más de un pieza del mismo tipo, se utiliza la información del lugar F, en el cual se almacena el tipo de pieza que puede ser cargada en cada una de las máquinas (expresión de arco $F \rightarrow T_4$: 1'(k,y), donde el valor k corresponde al tipo de pieza que todavía puede ser cargada. Como consecuencia de haber realizado una operación de carga de la máquina ensambladora, el número de posiciones ocupadas en la máquina ensambladora se

© Los autores, 2002; © Edicions UPC, 2002.

incrementa en 1 (expresión de arco $T_4 \rightarrow P$: 1'(n+1,y)) y, se incrementa el número de posiciones libres en el *palet* (expresión de arco $T_4 \rightarrow A$: 1'(x,y,z+1)).

T5 Transporte de una pieza del stock *local al* palet

Para que pueda producirse una operación de descarga de una pieza que se encuentra en el almacén local asociado a alguna de las dos celdas de trabajo al *palet*, es necesario que se cumplan las siguientes condiciones:

- Guarda [y=3 || y=5]: el *palet* debe estar situado en la posición y=3 (si se va a descargar la pieza de la primera celda de trabajo) o bien en la posición y=5 en el caso que se descargue la pieza desde el almacén local de la segunda celda de trabajo. El valor del color y se obtiene del arco de salida del nodo A, que recoge toda la información asociada a los *palets*.
- Guarda [z > 0]: debe haber espacio libre en el *palet*. El valor del color a que se obtiene del arco de salida del nodo A, indica el número de posiciones libres en el *palet*.

Adicionalmente, para que pueda producirse la operación de transporte se comprueba que el manipulador se encuentre libre (expresión de arco $C \rightarrow T_5$: 1'(y), donde el valor y corresponde al robot situado delante del *palet*) y que la pieza que se quiere transportar se encuentre en el almacén, lo cual se indica mediante el color x del arco de salida del nodo D, cuyo valor debe ser 7 u 8 en función de la posición del *palet* (valor x del elemento utilizado del nodo A). Como consecuencia de haber realizado una operación de descarga el número de posiciones libres en el almacén se incrementa en 1 (expresión de arco $T_5 \rightarrow E$: 1'(y,n+1)), se decrementa el número de posiciones libres en el *palet* (expresión de arco $T_5 \rightarrow A$: 1'(x,y,z-1)) y se actualiza el atributo que indica la nueva posición donde se encuentra la pieza (expresión de arco $T_5 \rightarrow D$: 1'(k,x)).

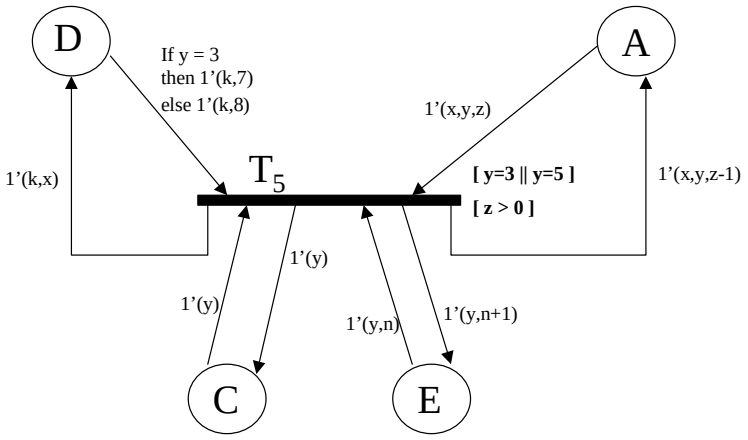


Figura 6.18 Transporte de una pieza de stock *a* palet

© Los autores, 2002; © Edicions UPC, 2002.

T6 Transporte de una pieza del stock *local a la máquina ensambladora*

Para que pueda producirse una operación de carga de una pieza que se encuentra en el almacén local a la máquina de ensamblado asociada a alguna de las dos celdas de trabajo, es necesario que se cumpla la siguiente condición:

- Guarda [k=3 || k=4 || k=5]: tan sólo se permite alimentar la máquina de ensamblado con piezas ya procesadas. El valor del color k que se obtiene del arco de salida del nodo D indica el tipo de pieza.

Adicionalmente, para que pueda producirse la operación de transporte se comprueba que el manipulador se encuentre libre (expresión de arco $C \rightarrow T_6$: 1'(y), donde el valor y corresponde al robot situado delante del *palet*) y que la pieza que se quiere transportar se encuentre en el almacén, lo cual se indica mediante el color x del arco de salida del nodo D, cuyo valor debe ser 7 ó 8 en función de la celda de trabajo a la que se haga referencia (valor p del elemento utilizado del nodo P). Para evitar que se pueda cargar más de un pieza del mismo tipo, se utiliza la información del lugar F, en el cual se almacena el tipo de pieza que puede ser cargada en cada una de las máquinas (expresión de arco $F \rightarrow T_6$: 1'(k,y), donde el valor k corresponde al tipo de pieza que todavía puede ser cargada. Como consecuencia de haber realizado una operación de carga de la máquina ensambladora, el número de posiciones ocupadas en la máquina ensambladora se incrementa en 1 (expresión de arco $T_6 \rightarrow P$: 1'(n+1,p)) y se incrementa el número de posiciones libres en el almacén (expresión de arco $T_6 \rightarrow E$: 1'(p,m+1)).

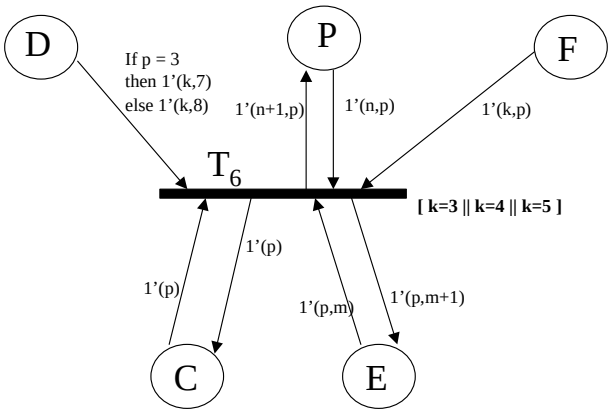


Figura 6.19 Transporte de una pieza de stock *a ensambladora*

© Los autores, 2002; © Edicions UPC, 2002.

T7, T8 Transporte de una pieza del stock local a la máquina CNC y procesamiento de la pieza

Para que pueda producirse una operación de carga de una pieza que se encuentra en el almacén local a la máquina CNC asociada a alguna de las dos celdas de trabajo, es necesario que se cumpla la siguiente condición:

- Guarda [k=1 || k=2]: tan sólo se permite alimentar la máquina CNC con piezas de tipo 1 o de tipo 2, ya que el resto de piezas no necesitan ninguna operación de procesamiento. El valor del color k que se obtiene del arco de salida del nodo D indica el tipo de pieza.

Adicionalmente, para que pueda producirse la operación de transporte se comprueba que el manipulador se encuentre libre (expresión de arco C→T_7: 1'(p), donde el valor p corresponde al robot situado delante de la máquina CNC), y que la pieza que se quiere transportar se encuentre en el almacén, lo cual se indica mediante el color x del arco de salida del nodo D, cuyo valor debe ser 7 ó 8 en función de la celda de trabajo a la que se haga referencia (valor p del elemento utilizado del nodo G). Finalmente, también es necesario comprobar que la máquina CNC se encuentra libre (valor b=0 del elemento utilizado del nodo G). Como consecuencia de haber realizado una operación de carga de la máquina CNC, se incrementa el número de posiciones libres en el almacén (expresión de arco T_7→E: 1'(p,n+1)) y se actualiza el estado de la máquina CNC a ocupado (expresión de arco T_7→G: 1'(p,1)).

El evento de fin de procesamiento de la pieza en la máquina CNC, tan sólo afecta a un cambio sobre el estado de la pieza, la cual se codifica mediante el color k del lugar H.

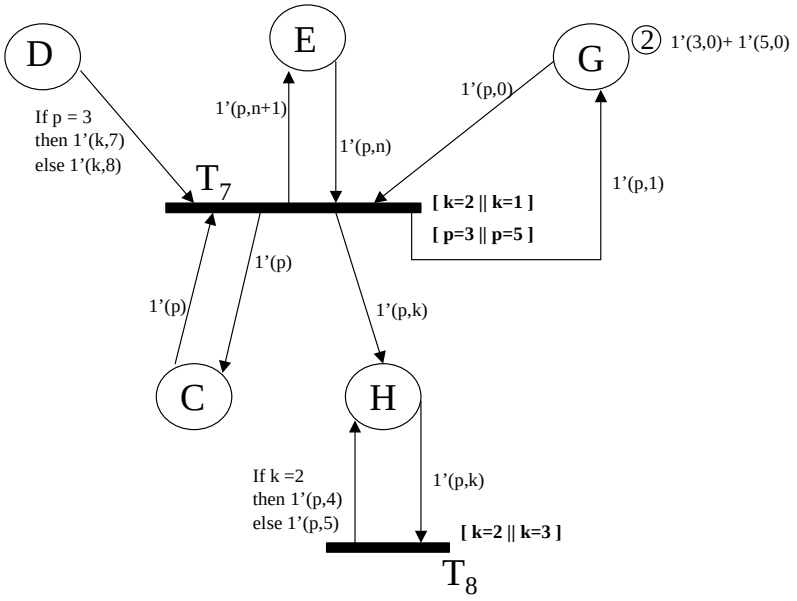


Figura 6.20 Transporte de una pieza de stock *a CNC*

© Los autores, 2002; © Edicions UPC, 2002.

T9 Transporte de una pieza del palet *a la máquina CNC*

Para que pueda producirse una operación de carga de una pieza que se encuentra en el *palet* a la máquina CNC asociada a alguna de las dos celdas de trabajo, es necesario que se cumplan las siguientes condiciones:

- Guarda [y=3 || y=5]: el *palet* debe estar situado en la posición y=3 (si se va a descargar la pieza en la primera celda de trabajo) o bien en la posición y=5 en el caso que se descargue la pieza en la máquina CNC de la segunda celda de trabajo. El valor del color y se obtiene del arco de salida del nodo A, que recoge toda la información asociada a los *palets*.
- Guarda [k=1 || k=2]: tan sólo se permite alimentar la máquina CNC con piezas de tipo 1 o de tipo 2, ya que el resto de piezas no necesitan ninguna operación de procesamiento. El valor del color k que se obtiene del arco de salida del nodo D indica el tipo de pieza.

Adicionalmente, para que pueda producirse la operación de transporte se comprueba que el manipulador se encuentre libre (expresión de arco $C \rightarrow T_9$: 1'(y), donde el valor y corresponde al robot situado delante de la máquina CNC) y que la pieza que se quiere transportar se encuentre en el *palet*, lo cual se indica mediante el color x del arco de salida del nodo D, cuyo valor debe coincidir con el identificador del *palet* (color x del elemento utilizado del nodo A). Finalmente, también es necesario comprobar que la máquina CNC se encuentra libre (valor b=0 del elemento utilizado del nodo G). Como consecuencia de haber realizado una operación de carga de la máquina CNC, se incrementa el número de posiciones libres en el *palet* (expresión de arco $T_9 \rightarrow A$: 1'(x,y,z+1)) y se actualiza el estado de la máquina CNC a ocupado (expresión de arco $T_9 \rightarrow G$: 1'(p,1)).

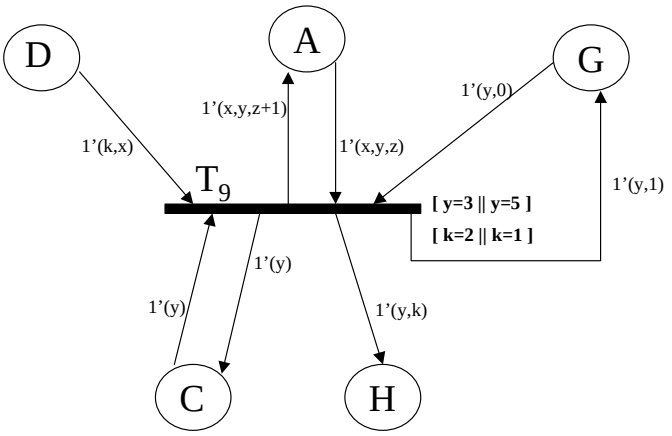


Figura 6.21 Transporte de una pieza de palet *a CNC*

© Los autores, 2002; © Edicions UPC, 2002.

T11, T12, T13 Transporte de una pieza de la máquina CNC al palet *o al* stock *local, o a la máquina ensambladora*

Para que pueda producirse una operación de descarga de una pieza que se encuentra en la máquina CNC de alguna de las dos celdas de trabajo al *palet*, es necesario que se cumplan las siguientes condiciones:

- Guarda [k=4 || k=5]: la pieza que se encuentra en le máquina CNC debe haber finalizado la operación de mecanizado (transición T_8).
- Guarda [z > 0]: debe haber espacio libre en el *palet*. El valor del color a que se obtiene del arco de salida del nodo A indica el número de posiciones libres en el *palet*.

Adicionalmente, para que pueda producirse la operación de transporte se comprueba que el manipulador se encuentre libre (expresión de arco C→T_{12}: 1'(y), donde el valor y corresponde al robot situado delante del *palet*), y que el *palet* se encuentre delante de la máquina CNC que se quiere descargar, lo cual se indica mediante el color y del arco de salida del nodo H, cuyo valor debe coincidir con el color y del *palet* donde se va a almacenar la pieza (valor y del elemento utilizado del nodo A). Como consecuencia de haber realizado una operación de descarga, se decrementa el número de posiciones libres en el *palet* (expresión de arco T_{12}→A: 1'(x,y,z-1)), se actualiza el atributo que indica la nueva posición donde se encuentra la pieza (expresión de arco T_{12}→D: 1'(k,x)) y se indica que la máquina CNC ya se encuentra libre (expresión de arco T_{12}→G: 1'(y,0)).

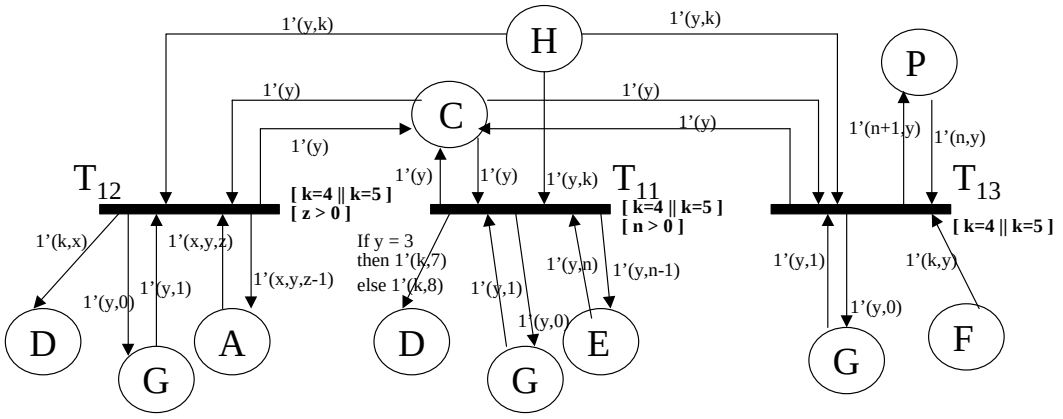


Figura 6.22 Transporte de CNC a palet, stock *o ensambladora*

El transporte de la pieza procesada al almacén local (T_{11}) presenta características muy similares a las comentadas en la transición T_{12}, pero en lugar de comprobar si existe espacio en el *palet*, se comprueba si existe espacio en el almacén (guarda [n>0], donde n es el color que indica el número de posiciones libres en el almacén).

De modo análogo, el transporte de la pieza procesada a la máquina ensambladora (T_{13}) también presenta características muy similares a las comentadas en la transición T_{12}, pero en lugar de

© Los autores, 2002; © Edicions UPC, 2002.

comprobar si existe espacio en el *palet*, se comprueba si existe espacio en la máquina ensambladora para el tipo de pieza procesada (expresión de arco F->T_{13}: 1'(k,y)).

T14,T15 Transporte de una pieza de la máquina ensambladora al stock *local o al* palet

Para que pueda producirse una operación de descarga de una pieza que se encuentra en la máquina ensambladora de alguna de las dos celdas de trabajo al *palet*, es necesario que se cumplan las siguientes condiciones:

- Guarda [y=3 || y=5]: el *palet* debe estar situado en la posición y=3 (si se va a descargar la pieza desde la primera celda de trabajo) o bien en la posición y=5 en el caso que se descargue la pieza desde la máquina ensambladora de la segunda celda de trabajo. El valor del color y se obtiene del arco de salida del nodo A, que recoge toda la información asociada a los *palets*.
- Guarda [q > 1]: como mínimo deben haber dos piezas en la máquina ensambladora para que ésta pueda integrarlas.
- Guarda [z > 0]: debe haber espacio libre en el *palet*. El valor del color a que se obtiene del arco de salida del nodo A indica el número de posiciones libres en el *palet*.

Adicionalmente, para que pueda producirse la operación de transporte se comprueba que el manipulador se encuentre libre (expresión de arco C→T_{15}: 1'(y), donde el valor y corresponde al robot situado delante del *palet*). Como consecuencia de haber realizado una operación de descarga, se decrementa el número de posiciones libres en el *palet* (expresión de arco T_{15}→A: 1'(x,y,z-1)), se actualiza el atributo que indica la nueva posición donde se encuentra la pieza (expresión de arco T_{15}→D: 1'(6,x) || 1'(7,x)), y se indica que la máquina ensambladora ya se encuentra libre (expresiones de arco T_{15}→P: 1'(0,y), T_{15}→F: 1'(3,y)+1'(4,y)+1'(5,y) || 1'(0,y)+1'(3,y)+1'(4,y)+1'(5,y)).

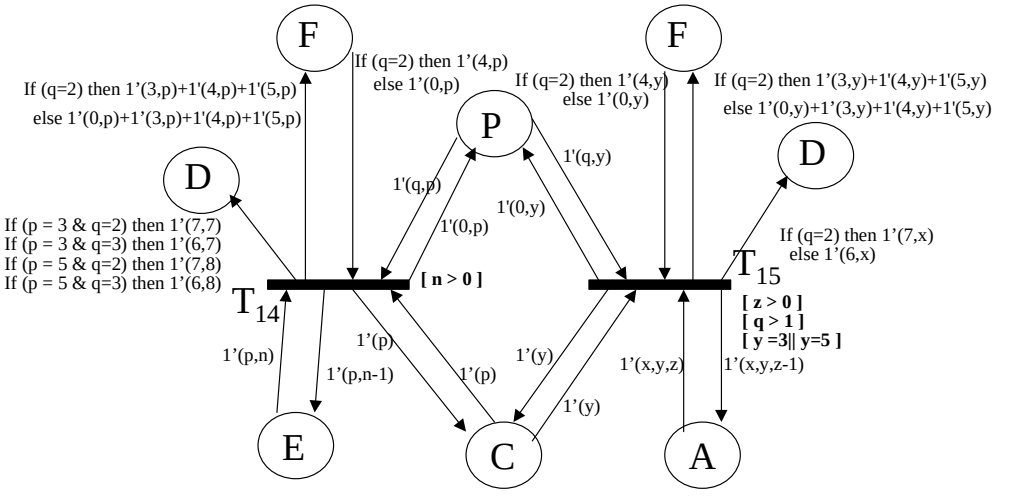


Figura 6.23 Transporte de ensambladora a stock *o* palet

© Los autores, 2002; © Edicions UPC, 2002.

El transporte de la pieza ensamblada al almacén local (T_{14}) presenta características muy similares a las comentadas en la transición T_{15}, pero en lugar de comprobar si existe espacio en el *palet*, se comprueba si existe espacio en el almacén (guarda [n>0], donde n es el color que indica el número de posiciones libres en el almacén).

6.6 Ejercicios

1. El problema de los filósofos se ha presentado en el apartado de ejercicios del capítulo 2. Empleando los tiempos de proceso descritos en el apartado de ejercicios del capítulo 4, construye un modelo de simulación para cada uno de los tres casos descritos en el capítulo 2. Simula el proceso durante un mínimo de 20 horas. Para el caso en que aparezcan bloqueos, simula este caso un mínimo de 20 veces con semillas diferentes y estima la probabilidad de que aparezca un bloqueo.

2. El sistema de canal-esclusa es un problema que ha sido publicado por la EUROSIM (Federación Europea de Sociedades de Simulación) en http://argesim.tuwien.ac.at/comparisons/ [SCHR96]. El objetivo perseguido es forzar el uso de complejas estructuras lógicas y de control. Adicionalmente, se pueden emplear técnicas de reducción de la variancia para el análisis estadístico del comportamiento del sistema.

   La figura 6.24 muestra el sistema canal-esclusa de interés. En una primera aproximación el tiempo de recorrido en el canal oeste y en el este son deterministas de valores 14 y 18 minutos respectivamente. Las barcazas pueden estar un tiempo adicional en el canal a la espera de poder entrar en la esclusa. El tiempo que permanece la barcaza en la esclusa puede variar entre 22 y 34 minutos.

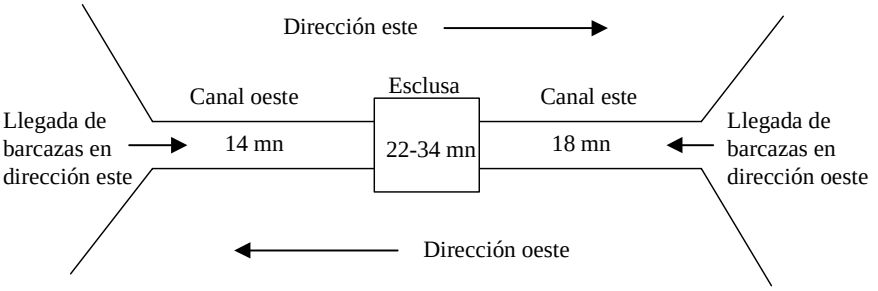


Figura 6.24 Esquema del sistema canal-esclusa

   Se han de tener en cuenta diversas consideraciones:

   - Como los canales son estrechos, sólo se permite la circulación en un sentido.
   - Se limita el máximo número de barcazas en cada sentido, es EsteMax si se dirige del embarcadero oeste al este, y OesteMax si es al contrario.
   - Si no hay barcazas en el sistema, la primera barcaza que llega ocupa el sistema y se inicia un ciclo de movimiento en su sentido de navegación.

© Los autores, 2002; © Edicions UPC, 2002.

- Un ciclo de movimiento termina cuando han entrado el número máximo de barcazas en un sentido, o cuando ha salido una barcaza, no hay más barcazas dentro del canal o la esclusa.
- Cuando un ciclo de movimiento termina y hay embarcaciones esperando para utilizar el sistema en sentido contrario a la embarcación saliente, se inicia un nuevo ciclo, en sentido inverso al de la barcaza que ha utilizado el sistema, tanto si hay como si no hay embarcaciones esperando en el mismo sentido de la barcaza saliente (considerando que se había llegado al máximo, sino habría entrado). Si, en cambio, no hay embarcaciones esperando a iniciar el ciclo en sentido contrario y hay embarcaciones esperando para realizar un ciclo de movimiento en el mismo sentido de la última barcaza, se reinicia el número máximo de barcazas que pueden entrar en el sistema y se inicia el ciclo en el mismo sentido que el anterior.
- Si al finalizar un ciclo no hay embarcaciones para entrar en el sistema, se espera a la llegada de la siguiente embarcación para iniciar un ciclo.

El funcionamiento de la esclusa es el siguiente:

- El nivel de agua del canal oeste, es superior al del canal este.
- El tiempo que tarda un barco en entrar o salir de la esclusa es de 5 minutos.
- El tiempo que tarda la esclusa en subir o bajar el nivel de agua es de 12 minutos, tanto si hay una embarcación dentro como si no.
- La esclusa es inteligente, por lo que si la dirección de movimiento es de este hacia oeste, una vez el barco ha salido en el canal oeste (nivel superior de agua), y hay más barcazas que desean entrar en la esclusa, ésta bajará automáticamente, sin esperar a que llegue a la puerta de la esclusa.

Por tanto, el tiempo mínimo en la esclusa será de 22 minutos si ésta ya está en la posición deseada cuando llega la barcaza. El tiempo máximo de servicio será de 34 minutos si la esclusa está en la posición contraria a la deseada. Obviamente, si cuando llega una barcaza la esclusa está operando con otra, ésta tendrá un tiempo de espera aparte del tiempo máximo de servicio.

a) Obtén la RdP del proceso.
b) Simula manualmente el comportamiento del proceso. Se puede aprovechar esta simulación manual para verificar, si se desea, la RdP anterior. Los casos que debes simular son:
    - El tiempo entre llegadas de barcazas en dirección Este es de 108 minutos y la primera llegada se produce en el instante 0. El tiempo entre barcazas en dirección oeste es de 108 minutos y la primera llegada se produce en el minuto 54. Los valores EsteMax y OesteMax son igual a 1. Simula el proceso hasta que el modelo ha sido completamente actualizado en el tiempo 1458 minutos.
    - El tiempo entre llegadas de grupos de 2 barcazas en dirección este es de 210 minutos y las dos primeras llegadas se producen en el instante 0. El tiempo entre llegadas de grupos de 3 barcazas en dirección oeste es de 210 minutos y las 3 primeras barcazas llegan en el minuto 54. Los valores EsteMax y OesteMax son igual a 2 y 3 respectivamente. Simula el proceso hasta que el modelo ha sido completamente actualizado en el tiempo 1470 minutos.

© Los autores, 2002; © Edicions UPC, 2002.

Las estadísticas de interés son el tiempo de tránsito en minutos tanto para cada dirección por separado como para las dos direcciones conjuntamente. El tiempo de tránsito se define como el tiempo que tarda la barca desde que llega hasta que sale por el extremo opuesto del canal.

c) Construye el modelo de simulación y verifica su correcto funcionamiento simulando los dos casos anteriores. Obtén los intervalos del 95% de confianza del tiempo de tránsito.

d) Analiza el comportamiento del sistema para tiempo entre llegadas para las dos entradas, que siguen una distribución exponencial de media 30, 50 o 70 minutos respectivamente, para 3 casos diferentes. ¿Qué valores consideras apropiados para los límites EsteMax y OesteMax para cada uno de los tres casos para reducir el máximo posible el tiempo de tránsito. Simula el sistema durante una semana para cada uno de los casos considerados.

3. Se emplean 10 barcazas para transportar mercancías en un canal. Las barcazas son cargadas por una de las dos grúas de carga disponibles en la parte baja del canal y descargadas por una de las dos grúas de descargas situadas en la parte alta del mismo. La travesía del canal se divide en dos zonas, la inferior y la superior, separadas por una esclusa. El tiempo de recorrido en cada zona depende de si la barcaza sube o baja. La esclusa sólo puede trasladar, subir o bajar, una barcaza cada vez. En la esclusa, las barcazas que suben cargadas tienen prioridad sobre las que bajan descargadas. Por tanto, si la esclusa no tiene carga de trabajo, se sitúa por defecto es su posición baja.

   El circuito para cada barcaza es, por tanto, cargar, remontar el tramo inferior, subir por la esclusa, remontar el tramo superior, descargar, descender por el tramo superior, bajar por la esclusa y descender por el tramo inferior. Para cada una de las operaciones que se realizan, se conoce el tiempo mínimo, el máximo y el más probable en horas:

Proceso	*T. mínimo*	*T. más probable*	*T. máximo*
Carga	2.00	2.25	3.00
Descarga	1.50	2.25	2.50
Subida o bajada Esclusa	0.40	0.50	0.60
Zona baja del canal			
En subida	2.00	2.50	3.00
En bajada	1.50	1.75	2.00
Zona alta del canal			
En subida	1.00	1.25	1.50
En bajada	0.50	0.75	1.00

Tabla 6.2 Tabla de tiempos

1. Determina teóricamente el número máximo de barcazas que pueden ser descargadas cada día.
2. Obtén la RdP del proceso.
3. Construye el modelo de simulación mediante la herramienta que tengas a mano.
4. Verifica, durante el tiempo que consideres oportuno, que las los eventos del modelo de simulación siguen la RdP obtenida.

© Los autores, 2002; © Edicions UPC, 2002.

5. Determina experimentalmente cuál el número promedio de barcazas que se pueden descargar cada día. Simula el proceso durante 1100 horas, tomando estadísticas después de las primeras 100 horas.

4. Este problema relacionado con el departamento de emergencias de un hospital ha sido desarrollado por el profesor Breiteneker de la Universidad de Viena y publicado por la EUROSIM en http://argesim.tuwien.ac.at/comparisons/.

   El departamento de emergencias de un hospital tiene una sección para efectuar el seguimiento de los pacientes que han sufrido accidentes y han tenido que ser posteriormente enyesados. Esta sección esta formada por:

   - Zona de recepción (zr) con una persona que redirige los pacientes al despacho 1 o al 2. También evalúa la necesidad de efectuar otros tratamientos.
   - Sala para la espera de los pacientes que tienen que entrar en los boxes.
   - Dos salas (S1 y S2) con dos despachos (b) y dos médicos para cada sala (un total de 4 despachos) pero la sala S2 está ocupada por dos médicos júnior (más inexpertos) que atienden los casos más simples.
   - Una sala de rayos X (rx) con dos equipos. Los pacientes esperan en una única cola para acceder a cualquiera de los equipos.
   - Una sala auxiliar (sa) ocupada por una enfermera que quita o pone las escayolas.

   Los pacientes empiezan a llegar y a registrarse a partir de las 7:30 de la mañana. Los médicos empieza a trabajar a partir de las 8:00. Se pueden clasificar cuatro tipos de pacientes según el circuito que siguen, tal como se muestra en la siguiente tabla:

Núm					
1	zr	b	rx	b	alta
2	zr	b	sa	alta	
3	zr	b	rx	b	alta
4	zr	b	alta		

Tabla 6.3 Circuitos asistenciales

   En el circuito 1 se efectúa una placa para seguir la evolución del paciente. En el 2 se le quita el yeso. En el 3, después de una placa, se le renueva el enyesado. En el circuito 5 se cambian los vendajes de las heridas.

	Mínimo	Más probable	Máximo
Recepción	0.2	0.5	1.0
S1	1.5	3.2	5.0
S2	2.8	4.1	6.3
Rayos-X	2.0	2.8	4.1
Sala auxiliar	3.0	3.8	4.7

Tabla 6.4 Distribución de los tiempos de servicio

   Los parámetros estadísticos del problema son:

© Los autores, 2002; © Edicions UPC, 2002.

- El tiempo entre llegada de pacientes sigue una distribución exponencial de media 0.3 minutos.
- El porcentaje de pacientes en cada uno de los circuitos es el siguiente: 1: 35%, 2: 20%, 3: 5% y 4: 40%.
- El 60% de los pacientes son admitidos en la sala S1 y el 40% restante en la sala S2. Se conocen los tiempos mínimos, máximos y más probables en minutos de tratamiento en cada uno de los puntos del circuito:

Se pide,

a) Determina teóricamente el tiempo promedio que permanece cada tipo de paciente en el sistema y el tiempo promedio ponderado para todos los tipos de pacientes.
b) Obtén la RdPCdel proceso.
c) Construye y verifica el modelo de simulación mediante la herramienta que tengas a mano.
d) Determina el tiempo promedio de cada tipo de paciente en el sistema y su intervalo de confianza del 95%. Para ello, simula el comportamiento de los 250 primeros pacientes. Analiza también por separado las colas para S1 y para S2.
e) Repite el caso anterior pero supón que un doctor experimentado de la S1 remplaza a uno de los inexpertos de la S2 tan pronto como en la cola S2 hay más de 20 pacientes. El tiempo de servicio del médico inexperto trabajando en la S1 es el que tenía en la S2 pero incrementado en un 20% debido a la mayor dificultad de los casos. Los médicos vuelven a su despacho original tan pronto como la cola de la S2 se reduce a 5 o menos pacientes. Modifica la RdPC para incorporar este comportamiento.
f) Intenta minimizar la desviación estándar del tiempo en el sistema global de todos los tipos de pacientes introduciendo prioridades. Una opción es priorizar a los pacientes que entran por segunda vez en el despacho respecto a los que entran por primera vez. El objetivo es que el tiempo que esté el paciente en el sistema sea muy parecido con independencia del circuito que siga.

5. El restaurante de la zona universitaria de Barcelona está intentando mejorar su funcionamiento en la franja horaria de 1:30 a las 3 del mediodia. Los clientes llegan en grupos de tamaño 1, 2, 3 y 4, con probabilidades respectivas 0.5, 0.3, 0.1 y 0.1. Los tiempos entre llegadas de los grupos siguen una distribución exponencial de media 30 segundos. Cada cliente, ya sea solo o como parte de un grupo, toma una de las tres siguientes rutas a lo largo del restaurante (en general cada miembro del grupo decide su ruta, por tanto el grupo se rompe):

   a) Comida caliente, a continuación la zona de bebidas y por último paga en caja.
   b) Bocadillos, a continuación la zona de bebidas y por último la caja.
   c) Sólo bebidas y a continuación la caja.

   Las probabilidades de cada una de las rutas son 0.8, 0.15 y 0.05 respectivamente. En el mostrador de comida caliente y en el de bocadillos los clientes son servidos uno a uno. Las bebidas son *self service* y, por tanto, se puede suponer que nadie tiene que hacer cola; es equivalente a pensar que existen infinitos servidores en el mostrador de bebidas. Finalmente, existen 2 o 3 cajeros, con una cola única para todos. Todas las colas en el modelo son FIFO.

© Los autores, 2002; © Edicions UPC, 2002.

En la tabla se indica el tiempo de servicio (TS) en cada mostrador y, adicionalmente, se representa en cada mostrador el tiempo en el cajero (TC). Por ejemplo, si un cliente escoge comida caliente y bebidas el tiempo de pago en caja es la suma de las dos distribuciones uniformes U(20,40)+U(5,10).

	Tiempo de servicio	Tiempo en el cajero
Comida caliente	U(50,120)	U(20,40)
Bocadillos	U(60,180)	U(5,15)
Bebidas	U(5,20)	U(5,10)

Tabla 6.5 Distribución de los tiempos de servicio

Por razones de seguridad, son necesarios un mínimo de dos cajeros y el máximo número de cajeros es 3. También tiene que existir como mínimo una trabajador en el mostrador de comida caliente y otro en el de bocadillos. Por tanto, el número mínimo de trabajadores es 4. Ejecuta el modelo en este caso básico y, a continuación, considerar la incorporación de más empleados. Por tanto, se pide:

a) Cuál es la mejor configuración si se tienen 5 trabajadores.
b) Cuál es la mejor si se tienen 6 trabajadores.

Para ello simula el comportamiento del sistema y evalúa los siguientes parámetros:

a) El retardo medio y máximo en las colas para comida caliente, bocadillos y cajeros.
b) El tamaño medio y máximo de las colas.
c) El tiempo medio y máximo de permanencia en el sistema (desde que entran hasta después de pagar).
d) El número total medio y máximo de personas en el sistema.

Inicialmente, el restaurante está vacío y la simulación se efectuará durante un periodo de 90 minutos.

6. Una barbería con tres barberos tiene que lidiar con clientes que desean sus barberos favoritos, clientes impacientes que marchan si la cola es excesiva y barberos ausentes [IOAN99]. El tiempo medio entre llegadas de clientes sigue una distribución exponencial de media 12 minutos y el tiempo de servicio sigue una distribución exponencial de media 19 minutos. El 30% del total de clientes prefiere el barbero B y el 10% de los clientes prefiere el C. Ninguno de los clientes prefiere el A, el propietario. Hay un 5% de probabilidades de que un barbero esté ausente un día determinado. La barbería trabaja de forma continua 6 horas cada día. A partir de la sexta hora no entra ningún cliente más pero el horario se alarga para servir a los clientes que quedan en la barbería. En los casos en que el cliente tiene un barbero favorito y esté está ausente, el cliente no se espera. Los clientes también miran la cola general: si hay más de 6 clientes en total esperando el cliente que llega se va. Adicionalmente, el cliente también se va si tiene un barbero favorito y éste tiene más de 3 clientes en la cola.

   a) Obtén la RdPC del proceso.
   b) Codifica el modelo de simulación y verifica su correcto funcionamiento.

© Los autores, 2002; © Edicions UPC, 2002.

c) Simula el proceso y obtén estadísticas del número de clientes que se van y la razón, el tiempo de espera en la cola, el estado de la cola y la ocupación de los barberos.

7. (Adaptado de [PEDG95]) El propietario de un restaurante en una zona turística está interesado en estudiar el flujo de clientes para cenar entre las 8 y las 12 de la noche. Los clientes llegan en grupos de 2, 3, 4 o 5 personas con probabilidades de 0.4, 0.3, 0.2 y 0.1 respectivamente. El restaurante tiene 50 mesas, en cada una de las cuales sólo pueden sentarse 2 personas. Las mesas se pueden juntar para acoger a grupos de más de 2 personas. Los grupos que llegan se ponen en cola en la zona de espera, pero si la cola es superior a 5 grupos, los clientes se marchan del local.

   El tiempo entre llegadas es de distribución exponencial con una media de 1.6 minutos en el periodo considerado. Un camarero acompaña a los clientes desde la zona de espera al comedor una vez está disponible el número de mesas requerido por el grupo. Ya en el comedor, el camarero se espera para tomar nota de los platos escogidos (Tria(3,4,5) minutos). El camarero hace el pedido a la cocina y vuelve a la sala de espera. Cuando la orden de la cocina está hecha (Tria(11,15,19) minutos), el mismo camarero que realizó el pedido pasa por la cocina para recoger la comida y servirla a los clientes. Después de servir (se supone que sirve en un solo viaje), vuelve a la sala de espera. El tiempo que tarda un grupo en comer sigue una distribución normal de media 24 minutos y desviación estándar 5 minutos.

   Cuando los clientes terminan de cenar se van a la caja donde pagan la cuenta. El tiempo que tarda el cajero en atender a los clientes sigue una distribución normal de media 1.5 minutos y desviación estándar de 0.5 minutos.

   Las distancias que recorre el camarero son: 30 metros desde la sala de espera al comedor; 30 metros desde la sala de espera a la cocina y 50 metros desde la cocina al comedor. Se supone que el camarero se desplaza a una velocidad de 30 metros por minuto. En la primera configuración que se estudia se supone que hay 3 camareros.

   a) Obtén la RdPC del proceso.
   b) Construye y verifica el modelo de simulación mediante la herramienta que tengas a mano.
   c) Simula el proceso durante todo el periodo de interés. Obtén estadísticas del número de clientes servidos, el número de mesas ocupadas, el número de clientes que se marchan sin comer y el factor de utilización de la caja.
   d) Determina el número máximo de camareros que este restaurante puede requerir como máximo.

8. Una empresa madrileña dedicada a la fabricación de material para la gestión de residuos ha recibido el encargo de fabricar los nuevos contenedores que la ciudad quiere implantar en sus calles. En la tabla siguiente se muestran los tres tipos de contenedores que se han de fabricar y su función:

Tipo	Función
1	Residuos plásticos
2	Residuos de papel y de cartón
3	Residuos orgánicos

© Los autores, 2002; © Edicions UPC, 2002.

Tabla 6.6 Tipos de contenedores

Dado que la demanda de contenedores ha sufrido un incremento muy importante y supone una muy buena posibilidad de negocio, la empresa ha decidido reestructurar la sección de contenedores y dividirla en dos subsecciones: la primera es la subsección de plástico, dedicada a la fabricación, por inyección, de las diferentes partes del contenedor, y la segunda es la subsección de montaje. Para comprobar el funcionamiento de la subsección de montaje, la empresa nos ha encargado su simulación.

La subsección de montaje está formada por una estación de entrada de material, por tres líneas de montaje, una por cado tipo de contenedor, y una estación de salida. Por la estación de entrada llegan 5 componentes:

- El recipiente del contenedor que es común para los tres contenedores que se fabrican.
- La tapa amarilla de los contenedores de residuos plásticos (tipo 1).
- La tapa azul de los contenedores de papel y cartón (tipo 2).
- La tapa bicolor de los residuos orgánicos (tipo 3).
- El pedal que permite levantar la tapa de los residuos orgánicos.

Los contenedores llegan en *palets* de 2 contenedores, los pedales en *palets* de 2 y las tapas en *palets* de 5. Las distribuciones de entrada en minutos de estos componentes se muestra en la tabla siguiente. La estación de entrada también se encarga de la distribución de los componentes hacia las diferentes líneas de montaje.

Elemento	*Distribuición*	*Media*	*Desviación*	*Cantidad en* palet
Contenedor	Normal	6	1	2
Pedal	Normal	18	1.5	2
Tapa plástico	Normal	45	2	5
Tapa papel	Normal	45	2	5
Tapa orgánica	Normal	45	2	5

Tabla 6.7 Distribuciones de llegadas de componentes

Las líneas de montaje de los contenedores de plástico (tipo 1) y papel (tipo 2) están formadas por una estación de trabajo cada una, donde se monta la tapa al contenedor. La línea de montaje de los contenedores orgánicos (tipo 3) está formada por dos estaciones de trabajo: la primera es la encargada de montar el pedal al contenedor y la segunda la encargada de montar la tapa. Los tiempos de proceso de cada estación de trabajo se detallan en la tabla siguiente.

Estación	Distribución
Montaje de pedales	Tria(12,15,15.5)
Montaje tapa plástico	N(10,1)
Montaje tapa papel	N(10,1)
Montaje tapa orgánica	N(45,16)

Tabla 6.8 Distribuciones de los tiempos de proceso

© Los autores, 2002; © Edicions UPC, 2002.

La estación de salida es la encargada de sacar de la planta los contenedores montados.

a) Encuentra analíticamente el número de puntos de trabajo que se precisan en cada estación para cumplir la producción cuando se trabaja de forma ininterrumpida 3 turnos de 8 horas por día.

b) Formaliza el modelo mediante una RdP y una RdPC.

c) Simula el modelo durante 5 días consecutivos y comprueba si el número de puntos de trabajo calculados analíticamente para cada estación es correcto.

9. En la figura siguiente se representa el sistema que se desea estudiar [CARR92]. Consiste en tres máquinas, un vehículo y doce estaciones de carga/descarga. Cada una de las estaciones de carga/descarga sólo manipula un tipo de piezas. La estación se emplea también para el almacenamiento del *palet*.

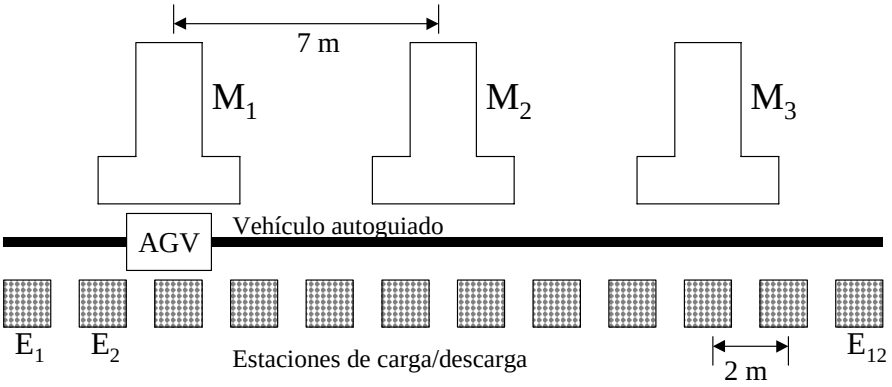


Figura 6.25 Célula de fabricación flexible

Las operaciones de carga/descarga de las estaciones de entrada/salida son realizadas por un trabajador. El tiempo de carga o el de descarga es de 3 mn. El vehículo transporta los *palets* con las piezas de las estaciones de carga a las máquinas o viceversa. La velocidad del vehículo es de 40 m/mn. El tiempo de transferencia de los *palets* entre el vehículo y la máquina o la estación de entrada/salida o viceversa es de 30 segundos.

Mientras que en el proceso real la mayoría de piezas precisan de dos o más operaciones, se supondrá en este ejemplo que cada pieza precisa de una sola operación y de cada *palet* es sólo utilizado por un tipo de piezas. El vehículo real tiene dos posiciones de carga y de transporte pero en este ejemplo se supondrá que el vehículo tiene una sola posición. Por tanto, sólo puede cargar una pieza cada vez. De forma resumida, las hipótesis más importantes son:

- El vehículo tiene una posición de carga y de transporte.
- Cada pieza requiere una sola operación.
- Las operaciones se pueden efectuar en cualquier máquina.
- Hasta que la máquina no ha sido descargada, no se puede cargar con otra pieza.
- Las máquinas no tienen averías.

© Los autores, 2002; © Edicions UPC, 2002.

- Cada estación de carga/descarga sirve a un tipo de pieza. Si hay un tipo de pieza en el sistema, hasta que no ha salido de la estación de carga/descarga no puede entrar una pieza del mismo tipo.
- Las fijaciones están permanentemente montadas en los *palet*.
- Se supone que se trabajan 20 días por mes, 3 turnos por día y 480 mn por turno.

En la tabla siguiente se muestra la previsión de llegadas de piezas de cada tipo así como el tiempo de proceso necesario para cada tipo de pieza.

Número de pieza	Cantidad por mes	Tiempo de proceso (mn)
1	320	18
2	270	60
3	260	36
4	180	40
5	140	36
6	120	24
7	100	32
8	80	28
9	72	20
10	64	20
11	48	30
12	40	32

Tabla 6.9 Previsión de llegadas de piezas

a) Calcula la utilización teórica del trabajador, de las máquinas y del vehículo autoguiado.
b) Obtén la RdPC del proceso.
c) Construye y verifica el modelo de simulación.
d) Simula el proceso durante un periodo de 20 días de trabajo continuo. Obtén estadísticas que permitan comparar los resultados teóricos del apartado a) con los obtenidos con la simulación. En una primera aproximación se puede suponer un tiempo constante de transporte de los *palets* con sus piezas.
e) Determina el número máximo de piezas que se pueden fabricar en el periodo considerado suponiendo que se mantienen las proporciones de la tabla anterior.
f) En el diseño anterior, las máquinas no pueden trabajar en el periodo que transcurre desde que el vehículo se lleva la pieza procesada hasta que vuelve con una pieza para procesar. Una primera opción para recuperar estos tiempos es instalar frente a cada una de las máquinas dos soportes de *palets*. Un soporte para las piezas no procesadas y otro para las procesadas. De esta forma, aunque la máquina esté mecanizando una pieza, el vehículo puede descargar sobre un soporte de entrada otra pieza. De esta manera, cuando la máquina finaliza el proceso descarga la pieza sobre el soporte de salida y coge la pieza no procesada del soporte de entrada. Para este nuevo diseño actualiza la RdPC anterior, actualiza el modelo de simulación, simula el proceso para obtener estadísticas sobre su comportamiento y determina el número máximo de piezas que se pueden fabricar suponiendo que se mantienen las proporciones de entrada.

© Los autores, 2002; © Edicions UPC, 2002.

10. (Adaptado de [LAW99]). En la figura siguiente se muestra un esquema del proceso de interés. Consiste en 5 estaciones de trabajo, una estación de entrada/salida de piezas y uno o más toros de transporte de piezas entre estaciones. Cada estación de trabajo tiene varias máquina iguales. El objetivo final del estudio es determinar cuántas máquinas son necesarias en cada estación y cuántos toros para cumplir de forma adecuada las especificaciones de producción.

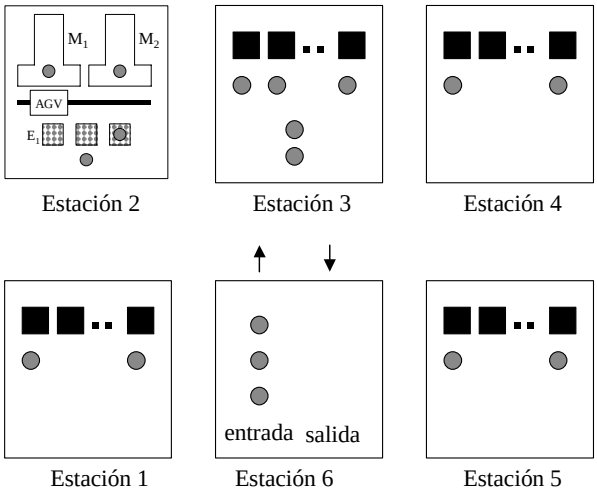


Figura 6.26 Proceso de fabricación

Se suponen que las piezas que se deben procesar llegan a la estación de entrada/salida según una distribución exponencial de media 1/20 horas/pieza. Hay tres tipos de piezas. La probabilidad de que una pieza sea de tipo 1 es de 0.2; de tipo 2, 0.5 y de tipo 3, 0.3. Cada tipo de pieza tiene un circuito diferente tal como queda reflejado en la siguiente tabla:

Tipo de pieza	Circuito	Tiempo medio de servicio (horas)
1	3,1,2,5	0.25, 0.20, 0.10, 0.30
2	4,1,2	0.15, 0.20, 0.30
3	2,5,3,1,4	0.15, 0.10, 0.35, 0.20, 0.25

Tabla 6.10 Circuitos y tiempos medios de servicio

Dado que las piezas entran y salen por la estación 6, hay que tener también en cuenta el desplazamiento desde la estación de entrada/salida hasta la primera estación de trabajo y desde la última estación de trabajo hasta la estación de entrada/salida. El tiempo de proceso de cualquier pieza en cualquier máquina sigue una distribución gamma de parámetro de forma de valor 2 (α=2), cuya media depende del tipo de pieza y de la estación sobre la que se trabaja. La tabla anterior también muestra el tiempo medio de trabajo por pieza y estación.

Cuando una máquina finaliza una operación, queda bloqueada hasta que el toro no coge la pieza y libera la máquina. El toro o toros se desplazan a 1.5 metros por segundo. La tabla siguiente muestra la distancia entre las diferentes estaciones:

© Los autores, 2002; © Edicions UPC, 2002.

	1	**2**	**3**	**4**	**5**	**6**
1	0	45.5	65	102	90	45.5
2	45.5	0	45.5	90	102	65
3	65	45.5	0	45.5	65	45.5
4	102	90	45.5	0	45.5	65
5	90	102	65	45.5	0	45.5
6	45.5	65	45.5	65	45.5	0

Tabla 6.11 Distancia entre estaciones en metros

a) Calcula analíticamente el número de máquinas en cada estación y el número de toros necesarios para procesar 160 piezas en un cada turno de 8 horas.
b) Obtén la RdPC del proceso.
c) Construye y verifica el modelo de simulación.
d) Simula el proceso durante un periodo de 320 horas. Obtén estadísticas que permitan comparar los resultados teóricos del apartado a) con los obtenidos con la simulación. Las estadísticas de interés pueden ser, por ejemplo: proporción de máquinas ocupadas en cada estación, proporción de máquinas bloqueadas en cada estación, tamaño promedio de la cola en cada estación, tamaño máximo de la cola en cada estación, producción total diaria promedio, tiempo promedio en el sistema de las piezas, tiempo promedio total en las colas, tiempo total promedio esperando por el transporte, proporción de toros que se mueven cargados y proporción de toros que se mueven descargados.
e) Determina el tiempo de calentamiento necesario para alcanzar el régimen permanente.
f) Determina un diseño adecuado del sistema, es decir, el número de máquinas necesarias en cada estación y el número de toros necesarios para alcanzar los objetivos.

11. En un centro comercial se quiere abrir una sala multicines. Estos multicines están formados por tres salas de proyección.

    A los multicines pueden llegar dos tipos de clientes: clientes con entrada, que habrán comprado previamente en terminales bancarios, y clientes sin entrada, que deberán comprar su entrada en las taquillas del cine. Después de realizar un estudio previo en otras salas multicines se ha determinado que el 93% de los clientes compran sus entradas en las taquillas y el resto las compran mediante terminales bancarios.

    También se ha analizado la llegada de clientes al cine. Normalmente los clientes llegan en grupos, compuestos entre 1 y 6 personas. Todos los componentes del grupo van a ver la misma película. En la tabla 6.12 se ha detallado la probabilidad que define el tamaño de los grupos.

Probabilidad	0.125	0.275	0.25	0.2	0.1	0.05
Número clientes	1	2	3	4	5	6

Tabla 6.12 Distribución del tamaño de los grupos de clientes

    La llegada de los grupos de clientes al cine no se realiza de manera uniforme. La llegada de clientes depende del tiempo que falta para el inicio de la sesión. Para la modelización de este comportamiento se ha dividido la llegada de clientes en cuatro intervalos, y dentro de cada uno se

© Los autores, 2002; © Edicions UPC, 2002.

modela una probabilidad de tiempo entre llegadas diferente (figura 6.27, donde EXPO(x) es una distribución exponencial de media x).

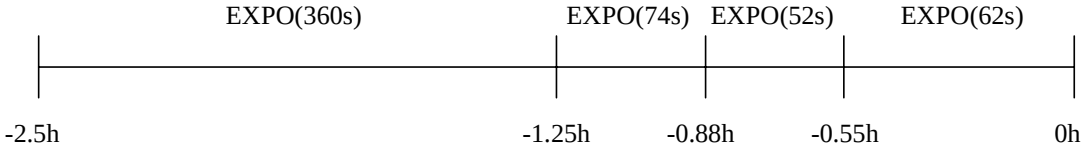


Figura 6.27 Distribución del tiempo entre llegadas de los grupos de clientes

La probabilidad de elección de película depende del tiempo que hace que se proyecta la película. En la sala 1 se proyecta la película más reciente y en la 3 la que más tiempo hace que se exhibe. En la tabla 6.13 se muestran estas probabilidades

Probabilidad	0.4	0.32	0.28
Sala	1	2	3

Tabla 6.13 Distribución de la elección de sala de proyección

Las taquillas están formadas por X puntos de venta de entradas, cada uno de ellos con un trabajador y con una cola independiente.

La zona de acceso está abierta una hora antes del inicio de la sesión. El control de acceso está operado por un trabajador.

Una vez los clientes han entrado en el cine éstos pueden comprar palomitas y golosinas en los Y puntos de venta que existen en el *hall* principal. Cada punto de venta tiene un trabajador y una cola independiente. Un 55% de los espectadores compran algún producto en estos puntos de venta, mientras que el resto va directamente a la sala de proyección.

En las tablas 6.14 y 6.15se especifican los tiempos de servicio y de desplazamiento de los trabajadores y de los clientes.

Tiempo de servicio	*Valor*
Venta de entradas en taquilla	Uniforme 100±15 seg
Control de acceso	Uniforme 10±2 seg
Venta de palomitas y golosinas	Uniforme 140±40 seg

Tabla 6.14 Tiempos de servicio

© Los autores, 2002; © Edicions UPC, 2002.

Tiempo de desplazamiento	*Valor*
De taquillas a control de acceso	Uniforme 40±10 seg
De control de acceso a punto de venta de palomitas y golosinas	Uniforme 30±10 seg
De punto de venta de palomitas y golosinas a sala de proyección	Uniforme 60±30 seg
De control de acceso a sala de proyección	Uniforme 60±30 seg

Tabla 6.15 Tiempos de desplazamiento

a. Calcula analíticamente cuál ha de ser la capacidad de cada una de las salas de exhibición, el número de taquillas y el número de puntos de venta de palomitas y golosinas.
b. Obtén la RdPC que modela este sistema
c. Construye y verifica el modelo de simulación.

Simula el modelo para una sesión 10 veces y obtén las estadísticas necesarias para analizar:

d. ¿Cuál es el número de clientes medio que no puede acceder a la sala deseada?
e. ¿Cuál es el tiempo de espera medio en las colas del sistema?
f. Analiza el sistema modificando la política de colas: ahora las colas son compartidas y sólo hay una cola para las taquillas y una para los puntos de venta de palomitas y golosinas.

12. Un departamento de una administración pública nos ha encargado el estudio de simulación de la evacuación de sus edificios principales. Debido a la importante posibilidad de negocio, nuestra empresa no quiere dejar pasar esta oportunidad, con lo que antes de intentar solucionar el problema con nuestros conocimientos de simulación hemos decidido investigar el estado de este tipo de simulación en el sector.

    a. Busca en Internet simuladores comerciales orientados a la evacuación de edificios.
    b. Busca información sobre métodos de modelización de estos sistemas.
    c. Compara las características de estos simuladores y métodos de modelización con los expuestos en los capítulos anteriores.

© Los autores, 2002; © Edicions UPC, 2002.

A Introducción a la teoría de colas

La mayoría de los estudios de simulación contemplan el modelado de colas. Por esta razón, se considera conveniente adquirir conocimientos básicos sobre la teoría de colas. Para modelos simples, la teoría de colas nos puede ayudar a analizar el comportamiento del modelo sin necesidad de utilizar la simulación. Para modelos complejos, los conceptos ligados a la teoría de colas nos pueden facilitar el análisis de los resultados de la simulación.

Un sistema de colas está formado por uno o más servidores que proporcionan servicio a los clientes que llegan. Si estos encuentran los servidores ocupados, pasan a ocupar una posición de espera frente a los servidores.

A.1 Notación de Kendall

A principios de la década de los cincuenta Kendall [KEND53] definió una clasificación de los sistemas que se caracterizan por tener una sola cola y por que los clientes llegan aleatoriamente. La estructura de la cola se puede definir con seis parámetros:

1. Distribución de los tiempos entre llegadas. Algunos de los valores posibles son:

   M = exponencial (Markoviano)
   G = general (la distribución no ha sido especificada)
   D = determinista (tiempo entre llegadas constante)
   E_k= distribución k-Erlang

2. Distribución del tiempo de servicio.
3. Número de servidores.
4. Capacidad máxima de la cola. Por defecto, la cola tiene capacidad infinita.
5. Número de clientes que llegarán. Por defecto, el número es infinito.
6. Disciplina de la cola: FIFO (*First-in-first-out*), LIFO (*Last-in-first-out*), prioritaria, u otras. Por defecto es FIFO.

El proceso de colas más simple es el M/M/1, en el cual las distribuciones del tiempo entre llegadas y del tiempo de servicio son exponenciales y hay un solo servidor en el sistema. Otro ejemplo puede ser M/M/3/20/1500/FIFO, el tiempo entre las llegadas y el tiempo de servicio siguen una distribución exponencial, hay tres servidores, la capacidad de la cola está limitada a 20 clientes, el número de clientes es 1500 y la política de la cola será FIFO.

En el caso de especificar la cola explícitamente, los valores por defecto se representan mediante el símbolo (·). Así, una sistema M/M/1 con política de cola LIFO se especifica como M/M/1/·/·/LIFO o un sistema con tamaño de cola infinito, 2 servidores a los que llegarán sólo 500 clientes, se especifica como M/M/2/·/500.

Los parámetros de un proceso de colas son:

© Los autores, 2002; © Edicions UPC, 2002.

- λ = frecuencia media de llegadas: Conocido el tiempo medio (esperado) entre llegadas E(A), la frecuencia media de llegadas se calcula como λ=1/E(A).
- ω = frecuencia media de servicio: Conocido el tiempo medio (esperado) de servicio E(S), la frecuencia media de servicio se calcula como ω=1/E(S).
- s = número de servidores.

Se pueden definir dos parámetros adicionales partiendo de los anteriores:

- Factor de utilización del servidor: $\rho = \frac{\lambda}{w \cdot s}$
- Intensidad del tráfico: $u = \frac{\lambda}{w}$

Para que el sistema pueda alcanzar el régimen permanente es necesario que $\lambda \leq \omega$. En caso contrario, el servidor puede quedar saturado y la longitud de la cola puede crecer sin límite. Por tanto, interesa generalmente que u sea menor o igual a 1. De forma equivalente, también interesa que el factor de utilización sea menor que 1.

A.2 Medidas de comportamiento

Son muchas las posibles medidas de comportamiento para un proceso de colas. En este apartado sólo se introducen las medidas más usuales.

Dado

- R_i = retardo promedio en la cola del cliente i
- S_i = tiempo de servicio en el servidor del cliente i
- $T_i = R_i + S_i$, tiempo total en el sistema del cliente i
- Q(t) = número de clientes en la cola en el instante de tiempo t
- L(t) = número total de clientes en el sistema en el instate de tiempo t, es decir, la suma de Q(t) y el número de clientes que se están sirviendo en el instante t

se puede definir el retardo promedio r y el tiempo total promedio en el sistema t en régimen permanente

$$r = \lim_{n\to\infty} \frac{\sum_{i=1}^{n} R_i}{n} \quad \text{y} \quad t = \lim_{n\to\infty} \frac{\sum_{i=1}^{n} T_i}{n}$$

De forma similar se puede definir el tamaño promedio de clientes en la cola Q y el número promedio de clientes en el sistema L (clientes en tránsito):

$$Q = \lim_{T\to\infty} \frac{\int_0^T Q(t)}{T} \quad \text{y} \quad L = \lim_{T\to\infty} \frac{\int_0^T L(t)}{T}$$

© Los autores, 2002; © Edicions UPC, 2002.

Nótese que es necesario que $\rho < 1$ para que r, t, Q y L converjan hacia un valor finito para un sistema G/G/s.

Una de las ecuaciones más interesantes y útiles de la teoría de colas es la denominada ley de Little [LITT61]:

$$L = \lambda t$$

que, conjuntamente con la ecuación,

$$Q = \lambda r$$

constituyen las ecuaciones de conservación. Por ejemplo, si la longitud observada de una cola en régimen permanente es de 2 (Q=2) y el tiempo promedio necesario para pasar por la cola es de 20 minutos (r=20), la frecuencia de llegadas es de λ=2/20 y, por tanto, el tiempo entre llegadas es de 10 minutos.

Una última ecuación de interés práctico es

$$t = r + E(S)$$

que nos indica que el tiempo total promedio en el sistema es el tiempo promedio en la cola más el tiempo de servicio.

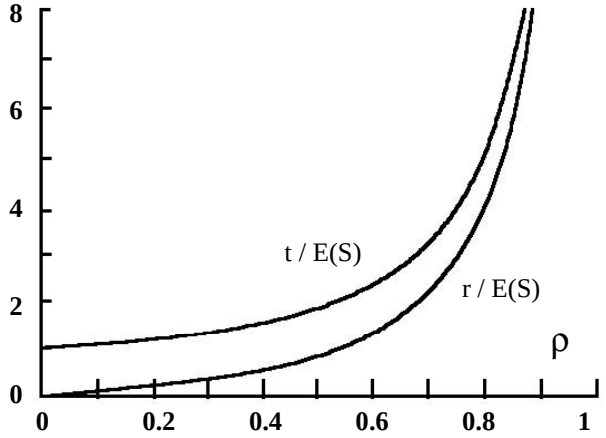


Figura A.1 Ratios de los tiempos de tránsito y de cola para un sistema M/M/1

La figura A.1 muestra la relación entre el tiempo de tránsito y el tiempo en la cola respecto al tiempo de servicio, en función del factor de utilización del servidor para un sistema M/M/1. Se observa que el sistema se deteriora rápidamente a partir de un factor de utilización de entre 0.6 y 0.7. Para factores de utilización bajos, el ratio tiempo total respecto al tiempo de servicio es aproximadamente 1 dado que el tiempo en cola es casi nulo.

© Los autores, 2002; © Edicions UPC, 2002.

B Modelado de sistemas con Arena

En los capítulos anteriores se ha evitado a propósito la descripción del proceso de construcción del modelo de simulación empleando cualquiera de las excelentes herramientas que se pueden encontrar en el mercado. Las razones principales de esta exclusión han sido:

- Se ha querido evitar la dependencia de una herramienta determinada.
- Se ha creído más importante desarrollar con más profundidad los aspectos metodológicos antes que incluir apartados de descripción de una o varias herramientas que hubiesen forzado a la reducción de los contenidos más fundamentales.

Sin embargo, la exclusión deja una laguna de conocimiento que es preciso cubrir: ¿cómo se puede construir el modelo de simulación con una herramienta específica a partir del modelo conceptual descrito con una RdPC? La respuesta más inmediata, desde una perspectiva académica, es emplear un simulador de redes de Petri, dado que acepta directamente el modelo conceptual desarrollado. Si esta es la opción escogida, el problema de construcción del modelo de simulación queda resuelto.

Desde una perspectiva más pragmática, puede interesar emplear una herramienta de simulación convencional pero sacando provecho de todo el potencial descriptivo de la RdPC. La principal razón para inclinarse por esta opción es el hecho que las herramientas convencionales de simulación son mucho más maduras para su aplicación al ámbito empresarial. Nótese que en los libros más importantes a escala internacional relacionados con la simulación de eventos discretos, las referencias a los simuladores de redes de Petri son inexistentes.

El reto que se plantea en este apéndice es, pues, cómo facilitar la construcción del modelo de simulación con una herramienta convencional aprovechando el potencial de la RdPC. Para el desarrollo de este apéndice se ha escogido la herramienta de simulación Arena versión 5.0 [KELT98], aunque también se pretende cubrir con otras herramientas, como por ejemplo, Witness. Todo el material adicional se puede obtener, incluido el código completo de los ejemplos que se muestran a continuación, en http://www.edicionsupc.es/poli118. Este capítulo no pretende ni es un manual de usuario del paquete de simulación Arena. Debe contactarse con el distribuidor (en España, http://www.drilco.es/) si se desea información sobre el uso de la herramienta o una copia de su versión limitada para la realización de estos ejercicios.

B.1 Modelo de un recurso limitado

La figura B.1 representa un proceso simple compuesto básicamente por una estación de trabajo formada por dos máquinas. Las piezas que llegan para ser procesadas se esperan en una única cola FIFO (*first-in-first-out*) para ser procesadas. Finalmente, abandonan el proceso.

© Los autores, 2002; © Edicions UPC, 2002.

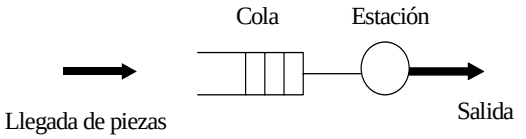


Figura B.1 Estación de trabajo.

La figura B.2 muestra el modelo en RdP del proceso anterior.

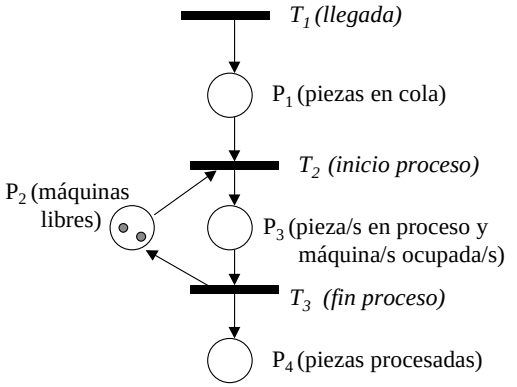


Figura B.2 RdP de la estación de trabajo

Si se analiza este ejemplo desde una *orientación a procesos*, las piezas (entidades dinámicas) en P1 compiten para capturar (*Seize*) un recurso limitado máquina (entidad permanente). Una vez se ha efectuado la captura del recurso máquina (transición T2) se inicia el procesado de la pieza que se modela como el retardo (*Delay*) que sufre la pieza debido al proceso (lugar P3) Una vez finalizado el proceso (transición T3), se libera (*Release*) el recurso máquina y la pieza procesada sale del sistema.

La figura B.3 muestra el modelo de simulación del proceso. El componente *CREATE* genera las llegadas de piezas, el *SIZE* captura el recurso máquina, el *DELAY* modela el tiempo de proceso, el *RELEASE* libera la máquina y el *DISPOSE* elimina la pieza (la pieza sale del modelo). Las etiquetas P1, P1-T2,... que aparecen debajo de cada componente son simples comentarios que nos permiten relacionar el diagrama de bloques con la RdP anterior.

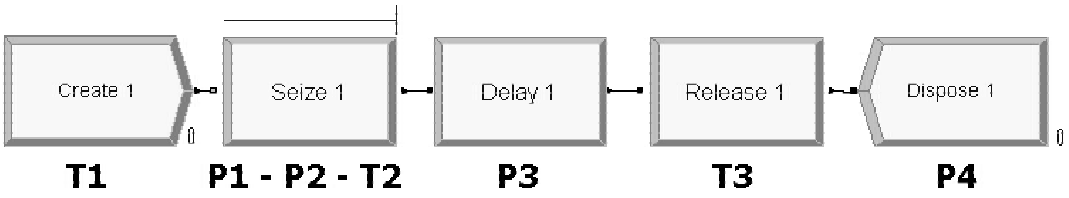


Figura B.3 Modelo B1a

© Los autores, 2002; © Edicions UPC, 2002.

Dado que la captura de recursos, su empleo y su posterior liberación se efectúa continuamente en un modelo de simulación, encontramos en la herramienta el componente *PROCES* que junta la funcionalidad de los bloques *SIZE*, *DELAY* y *RELEASE*.

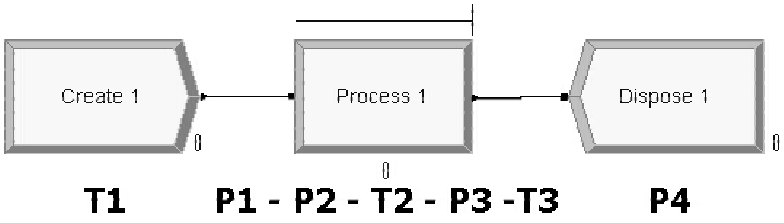


Figura B.4 Modelo B1b

Como se ha comentado anteriormente, los modelo B1a y B1b están *orientados a procesos*, y se construyen desde el punto de vista de las entidades temporales que fluyen por el modelo. En la mayoría de los modelos se empleará esta estrategia de simulación dado que es la estrategia soportada de forma natural por la herramienta que se utiliza (Arena). En algunos casos particulares puede resultar interesante emplear otra estrategia, estos casos son:

- La orientación a procesos es insuficiente para modelar parte del proceso de interés.
- Manteniendo la orientación a procesos es necesario emplear mecanismos sofisticados de la herramienta cuando con otra estrategia el modelo sería mucho más simple.

Una de las alternativas que da buenos resultados consiste en emular directamente la RdP del proceso, con lo cual nos encontraríamos empleando la estrategia de *exploración de actividades* en una herramienta orientada a procesos. Hay que recordar que, como se comentaba en el capítulo 4, en la exploración de actividades no se diferencian de forma explícita las entidades dinámicas (piezas) de las entidades permanentes (máquinas). Por tanto, con esta aproximación, no se emplearán los mecanismos internos de la herramienta para la gestión de recursos. La máquina fluirá por el modelo al igual que las piezas que se quieren mecanizar.

En la figura B.5 se muestra el modelo del proceso considerado, construido a partir de la emulación de la RdP. Con esta estrategia ya no aparecen los bloques para la captura y liberación explícita de recursos. En su lugar, encontramos la emulación de la transición T2 y sus dos lugares incidentes P1 y P2 para la captura del recurso máquina, y la transición T3 para la liberación del recurso. No obstante, desde la estrategia de la RdP, la lectura de la transición T2 es diferente; diremos que el evento asociado a la transición T2 se puede activar si tenemos simultáneamente una entidad en P2 (máquina libre) y una entidad en P1 (pieza en cola). Si recordamos el capítulo 2, la estructura de esta transición modela el concepto de *sincronismo en el tiempo*.

© Los autores, 2002; © Edicions UPC, 2002.

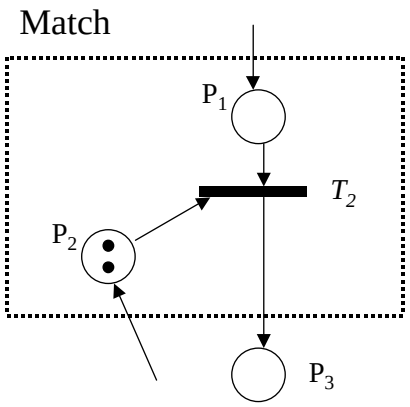


Figura B.5 Sincronización (MATCH)

El componente *MATCH* del Arena nos permite codificar la transición T2 de la RdP y sus lugares incidentes (figura B.5). Otros aspectos que se deben destacar son:

- El componente *CREATE* de nombre Create 3, inicializa el marcado inicial del lugar P2 para indicar que en el instante inicial hay dos máquinas libres en P2.
- Hay dos salidas en el componente *MATCH*, de nombre Match 1 T2, que codifica la transición T2, cuando en la RdP sólo hay una salida. Se ha considerado interesante hacerlo así, aunque no es necesario desde la perspectiva de la RdP, para separar claramente el camino que siguen las piezas del camino de las máquinas. Esta separación puede tener interés para la extracción de estadísticas.
- El componente *MATCH*, de nombre Match 2 T3, fuerza a que la pieza no pueda progresar hasta que la máquina no haya finalizado el proceso (emulado por el componente *DELAY*). Por tanto, es lo mismo imputar el retardo a la máquina que a la pieza.

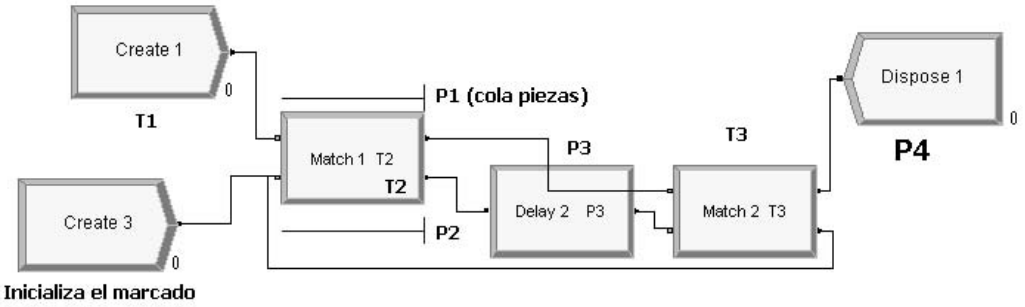


Figura B.6 Ejemplo B1c

Esta claro que el *modelo B1b* presentado es el más simple y fácil de construir. Las alternativas presentadas han tenido como objetivo principal dotarnos de recursos para abordar con más seguridad modelos más complejos.

© Los autores, 2002; © Edicions UPC, 2002.

B.2 Proceso robotizado simple

Este ejemplo muestra un proceso robotizado simple que se caracteriza por que dos flujos diferentes de fabricación compiten por un mismo recurso, dado que el robot tiene que procesar tanto las piezas tipo 1 como las piezas tipo 2.

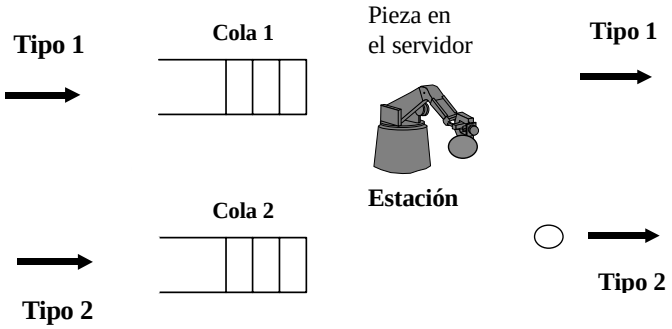


Figura B.7 Ejemplo de proceso robotizado simple

La figura siguiente muestra la RdP del proceso anterior. Obsérvese que la entidad que está en la posición P_1, y que indica que el robot está libre, puede emplearse para activar la transición T_3 o la T_4. Por tanto, en el caso que los lugares P_2 y P_3 tengan simultáneamente más de una pieza, las dos piezas compiten entre sí, para capturar el recurso restringido robot, representado por una marca en el lugar P_1 cuando éste está libre.

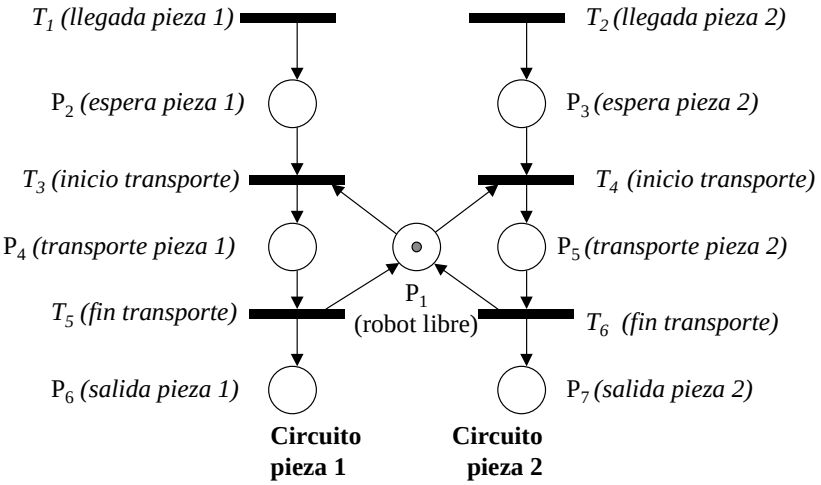


Figura B.8 RdP del proceso robotizado simple.

En la figura B.9 se muestra el código del proceso; se observa, tanto en la RdP como en el código, que los dos tipos de piezas evolucionan de forma concurrente (en paralelo). El único elemento en común

© Los autores, 2002; © Edicions UPC, 2002.

es el robot, recurso limitado por el cual compiten los dos tipos de piezas. Nos encontramos en una situación de conflicto cuando dos piezas de tipo diferente solicitan en el mismo instante de tiempo el único robot del proceso. Para resolver este conflicto el componente *PROCESS* (y también el *SIZE*) nos permiten indicar la prioridad de la petición. Por tanto, podemos optar en priorizar un tipo de pieza respecto a otra, es decir, en caso de conflicto el robot atenderá a la pieza más prioritaria. Otra alternativa es dar la misma prioridad a los tipos de piezas, de forma que se atiendan todas las peticiones por orden de llegada, sin tener en cuenta el tipo.

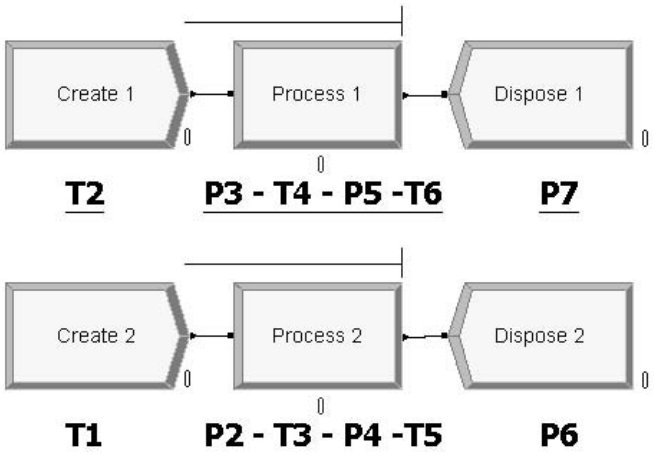


Figura B.9 Ejemplo B2

Obsérvese que los dos bloques *PROCESS* comparten el mismo recurso robot, y por tanto están compitiendo por él.

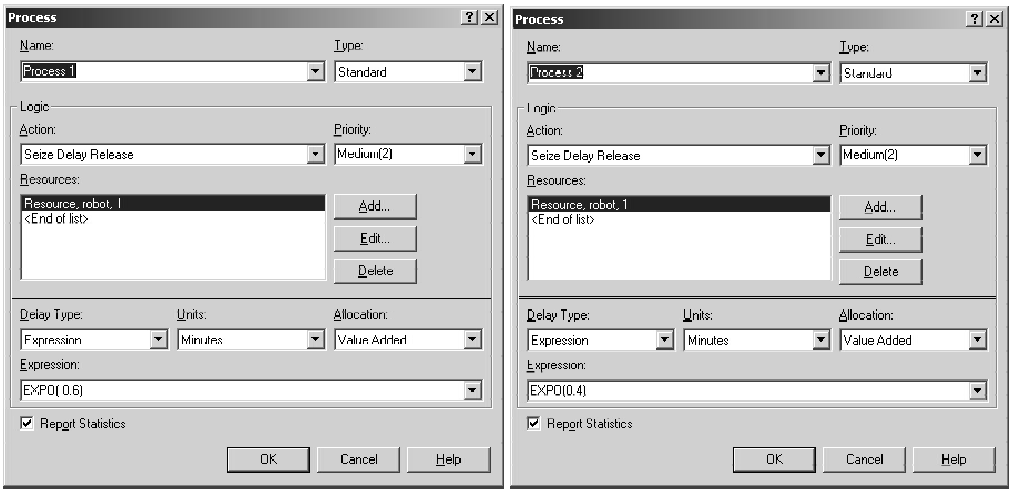


Figura B.10 Comparación de los dos bloques PROCESS

La figura siguiente muestra la evolución de las dos colas de piezas esperando el robot. Cuando los dos tipos de piezas tienen igual prioridad, las desviaciones en los tamaños de la cola de un tipo respecto al

© Los autores, 2002; © Edicions UPC, 2002.

otro sólo son debidas a las variaciones en el ritmo de llegadas. Cuando la pieza tipo 2 es prioritaria se incrementa de forma apreciable la cola de piezas tipo 1. Las piezas tipo 1 sólo son servidas cuando no hay ninguna pieza tipo 2 esperando en la cola.

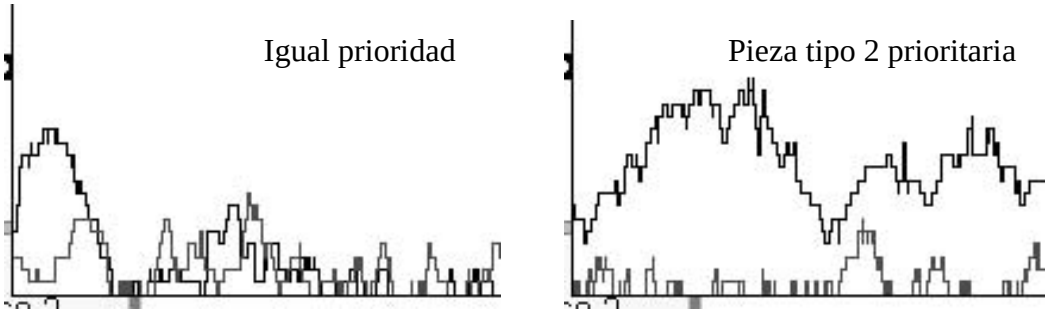


Figura B.11 Priorización de las piezas

B.3 Línea de producción

Considérese la línea de producción descrita en la figura B.12, la cual está formada por las siguientes unidades de producción:

- Dos robots: R_1 y R_2
- Tres máquinas CNC: M_1, M_2 y M_3
- Dos unidades de almacenamiento: S_1 y S_2

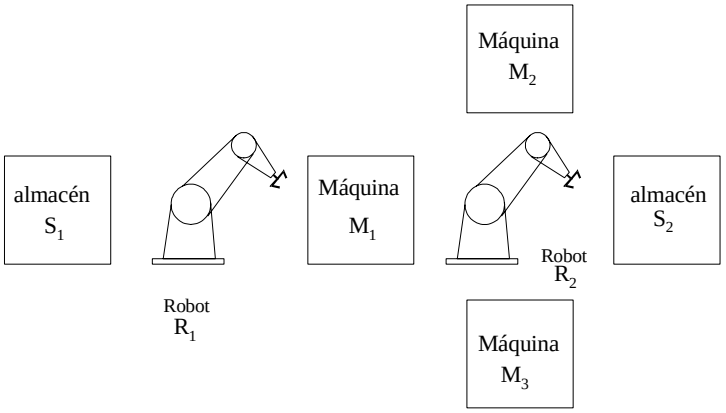


Figura B.12 Línea flexible de fabricación

El funcionamiento de la línea de producción es el siguiente:

9. El robot R_1 realiza el transporte de las piezas almacenadas en S_1 y las carga en la máquina M_1.
10. La máquina M_1 realiza la 1ª operación sobre las piezas.
11. Una vez la máquina M_1 ha finalizado la 1ª operación, el robot R_2 transporta la pieza procesada en M_1 y la carga en la máquina M_2 o en la máquina M_3.

© Los autores, 2002; © Edicions UPC, 2002.

12. Una vez la máquina M_2 o la máquina M_3 ha finalizado la 2ª operación, el robot R_2 transporta la pieza finalizada al almacén de salida S_2.

En la figura B.13 se ha representado la RdP de todo el sistema y en la figura B.14 se muestra el código del modelo. Para contemplar que siempre hay piezas disponibles en el almacén de entrada sólo se genera una pieza en el componente *CREATE*, pero después de que la pieza captura el robot 1 y la máquina 1 en el componente *SEIZE*, de nombre Seize 1, se genera un duplicado de la pieza (entidad temporal) que se envía a la entrada del componente *SEIZE* para indicar que hay otra pieza que desea entrar en el sistema. Otro componente de interés es el *SCAN*, que codifica el conflicto originado por el hecho de que la pieza puede dirigirse a la máquina 2 o a la 3. El componente *SCAN* bloquea el avance de la entidad hasta que la máquina 2 o la 3 estén libres. Si las dos están libres, se escoge el camino de la máquina 2. Una vez, una o las dos máquinas estén libres, el componente *DECIDE* direcciona la pieza a una de las dos máquinas libres.

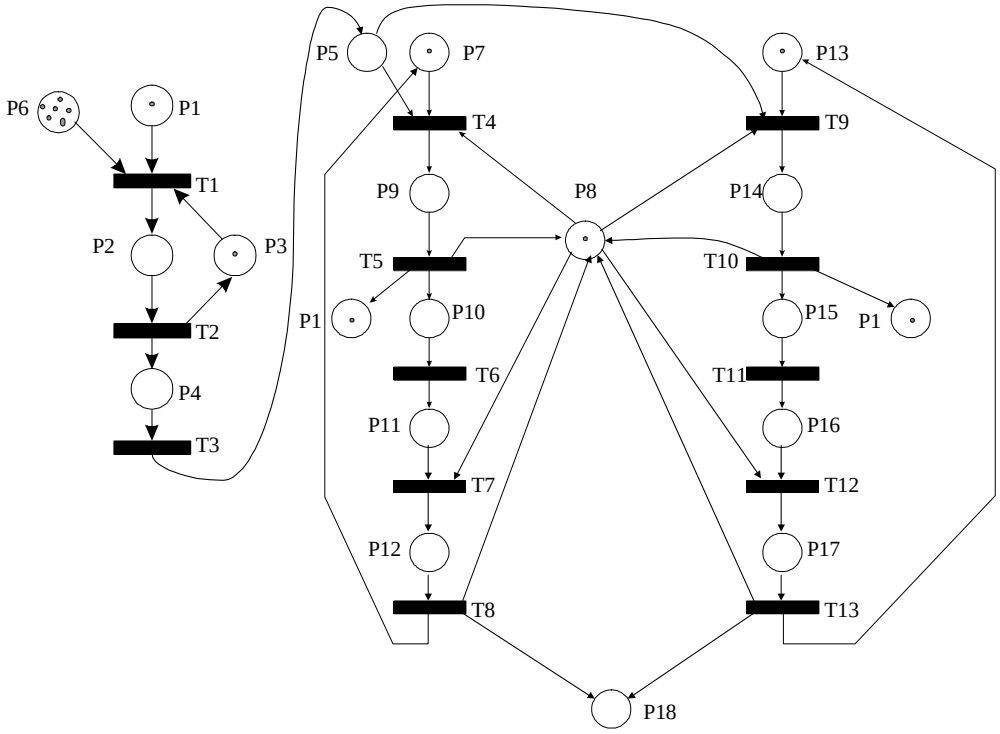


Figura B.13 RdP de la línea de producción

© Los autores, 2002; © Edicions UPC, 2002.

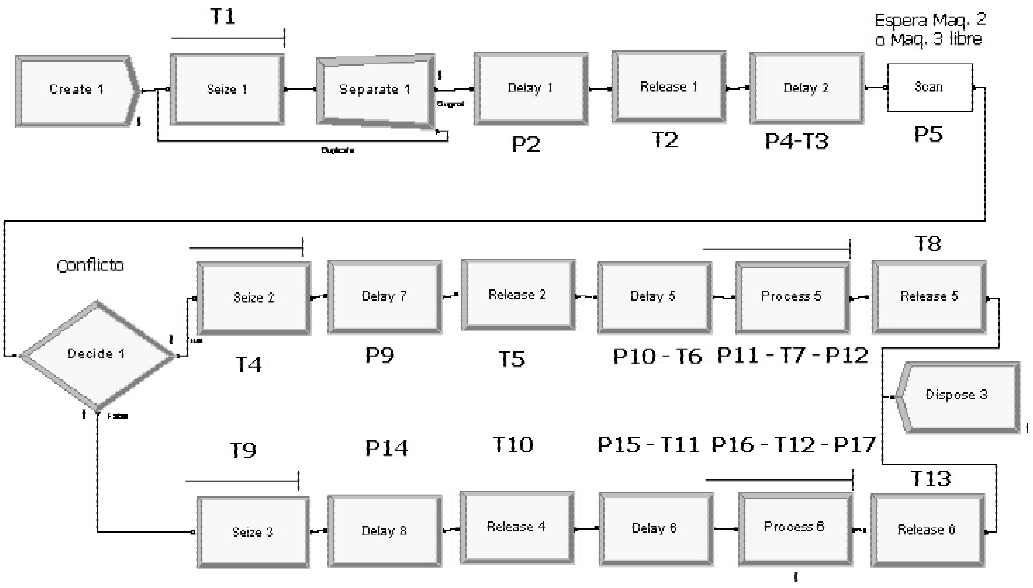


Figura B.14 Ejemplo B3

B.4 Dos máquinas y dos cintas

En la figura siguiente se representa un sistema de producción formado por dos máquinas (M_1 y M_2), dos cintas transportadoras con un *palet* en cada una de ellas (C_1 y C_2), y dos tipos de piezas (a y b) para ser procesadas. La secuencia de operaciones tiene que respetar las siguientes restricciones:

6. La secuencia de operaciones para una pieza de tipo a es: inicialmente se procesa en la máquina M_1; una vez finalizada la operación, se emplea el *palet* C_1 para transportar la pieza a la máquina M_2 donde, una vez procesada, sale del sistema.

7. La secuencia de operaciones para la pieza de tipo b es la misma que la descrita para las piezas de tipo a, pero se emplea el *palet* C_2.

8. Las máquinas no pueden realizar consecutivamente dos operaciones sobre el mismo tipo de pieza. Por tanto, cuando una máquina acaba de procesar una pieza de tipo a, la siguiente tiene que ser de tipo b, y viceversa.

9. Para que la máquina M_1 quede libre una vez ha finalizado el proceso de una pieza de tipo a, es necesario que el *palet* C_1 se encuentre al lado de la máquina M_1.

10. Para que la máquina M_1 quede libre una vez ha finalizado el proceso de una pieza de tipo b, es necesario que el *palet* C_2 se encuentre al lado de la máquina M_1.

© Los autores, 2002; © Edicions UPC, 2002.

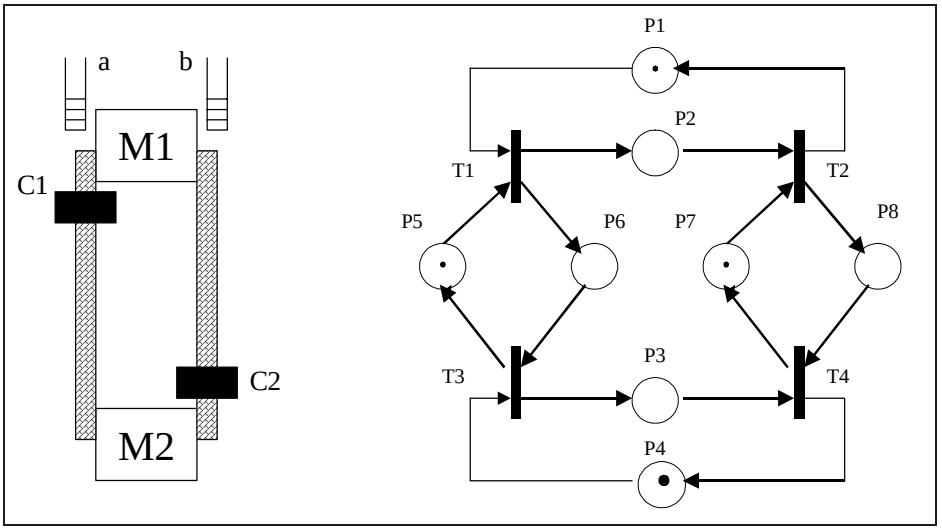


Figura B.15 Sistema de producción y su RdP

En la figura B.15 se ha representado la RdP que modela el sistema de producción descrito. Se considera que el almacén de piezas tipo a y tipo b tiene tamaño infinito. También se ha supuesto que los *palets* C_1 y C_2 tienen sólo dos posiciones cada uno, posición superior y posición inferior.

El significado asociado a los nodos de la RdP es:

- P1: *palet* C_1 en la posición superior (M_1).
- P2: *palet* C_1 en la posición inferior (M_2).
- P3: *palet* C_2 en la posición inferior (M_2).
- P4: *palet* C_2 en la posición superior (M_1).
- P5: máquina M_1 procesando pieza tipo a.
- P6: máquina M_1 procesando pieza tipo b.
- P7: máquina M_2 procesando pieza tipo b.
- P8: máquina M_2 procesando pieza tipo a.

La figura siguiente muestra el código del proceso construido directamente a partir de la RdP. Las transiciones T_1, T_2, T_3 y T_4 son codificadas con el componente *MATCH*. Los lugares P_1 a P_8 de la RdP contemplan tanto el tiempo de operación (del transporte o de la máquina) más la espera para poder iniciarse la siguiente operación. En el código los tiempos de proceso se han codificado con el componente *DELAY* y las esperas son parte de las colas asociadas a los componentes *MATCH*.

Para inicializar el modelo (especificar el marcado inicial de la RdP), se han incluido cuatro componentes *CREATE* (Inicia P_1,..., Inicia P_4) que crean las entidades iniciales del modelo. Otro aspecto de interés es que tanto la RdP como el código no contemplan de forma explícita las piezas procesadas, dado que se supone que siempre están disponibles para el inicio del ciclo de operaciones para cada tipo de pieza. El modelo anterior se puede ampliar para obtener estadísticas de su comportamiento.

© Los autores, 2002; © Edicions UPC, 2002.

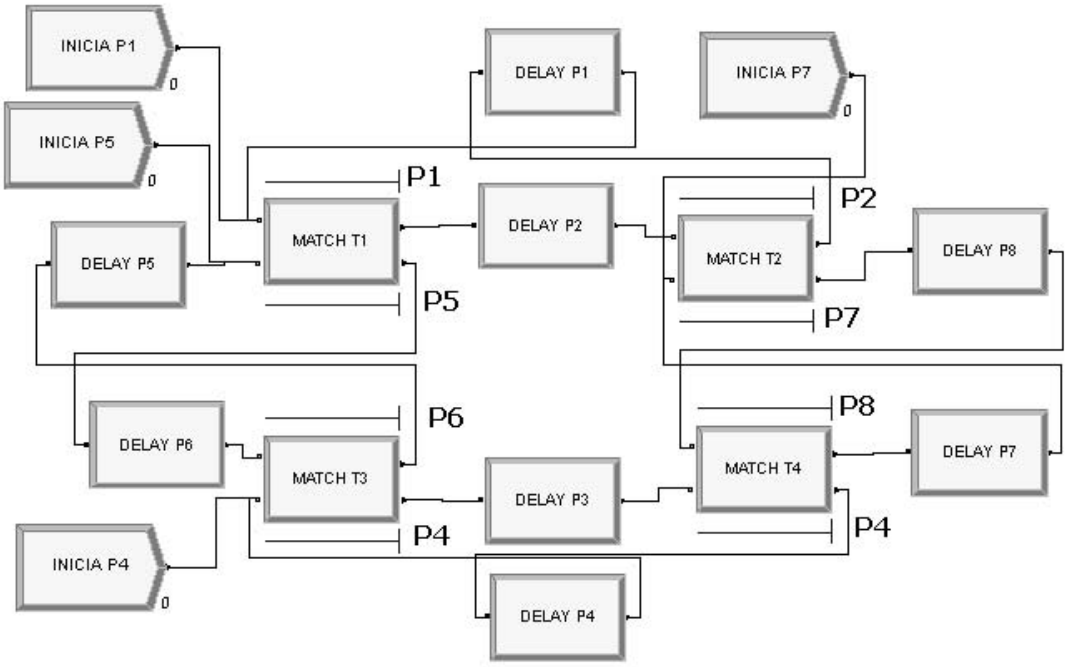


Figura B.16 Ejemplo B4

B.5 Control de vehículos

Los paquetes de simulación actuales trabajan bien con vehículos si no hay diferenciación de tareas entre ellos. Sin embargo, si el control de cada uno o de grupos de vehículos tiene que ser personalizado es necesario diseñar mecanismos propios de control. En la práctica, nos encontramos con los siguientes problemas:

- Si pedimos cualquier vehículo no tendremos control sobre cuál vendrá.
- Si pedimos un vehículo en particular, tal vez esté ocupado cuando uno equivalente puede estar libre.

La figura siguiente muestra un ejemplo que pretende mostrar la forma de abordar este problema. Se tiene un proceso con dos tipos de vehículos. Si no están ocupados permanecen en la zona de aparcamiento de vehículos. Aleatoriamente, pueden llegar contenedores de dos tipos al punto de carga de contenedores. Si el contenedor es de tipo 1, un vehículo libre cualquiera asignado a este tipo de contenedores se traslada al punto de carga y traslada el contenedor a su punto de descarga. De forma equivalente se procede si el contenedor es de tipo 2.

© Los autores, 2002; © Edicions UPC, 2002.

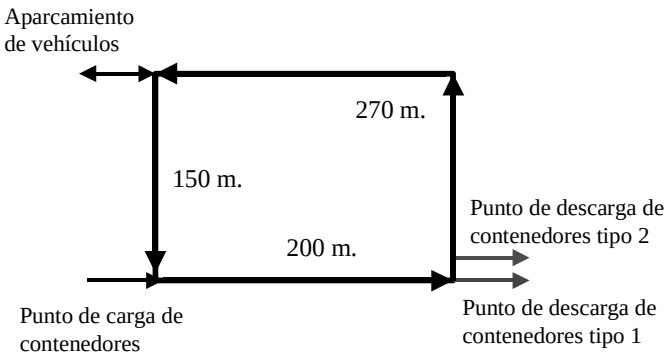


Figura B.17 Ejemplo de control de vehículos

El modelo conceptual del ejemplo anterior se muestra en la figura B.18. Dado que hay que diferenciar los dos tipos de contenedores y vehículos, se emplea un RdPC cuyos elementos son:

- Σ= {ti}, donde ti es el tipo 1 o 2.
- P={P1,...,P7}, donde:

  P1: vehículos libres.
  P2: contenedor en espera para ser cargado.
  P3: el vehículo se desplaza hacia el punto de carga.
  P4: contenedor en vehículo que procede a su traslado.
  P5: contenedores almacenados en su destino.
  P6: petición de vehículo para un contenedor de tipo i.
  P7: el vehículo se desplaza desde el punto de descarga hasta el aparcamiento.

- T={T1,...,T5}, donde:

  T1: llegadas de contenedores al punto de carga según la distribución que se crea conveniente. A cada contenedor se le asigna al azar, en este ejemplo, su tipo.
  T2: si hay una orden de transporte de un contenedor de tipo i y un vehículo libre para transportar contenedores de tipo i, ordena al vehículo que se desplace al punto de carga.
  T3: cuando el vehículo llega al punto de carga, carga el contenedor.
  T4: cuando el vehículo llega a su punto de descarga, el contenedor es descargado y el vehículo queda libre.
  T5: el vehículo llega al aparcamiento

- A={A1,...,A12} es el conjunto de arcos
- C(pl) = t para todo lugar.
- G(t) = verdadero si t ε {T1,T4,T5}
  G(t) = [ti] si t ε {T2, T3}
- I(P1) = 3’(1)+4’(2)
- E(a) = 1’ti si a ε {A1,...,A12}

© Los autores, 2002; © Edicions UPC, 2002.

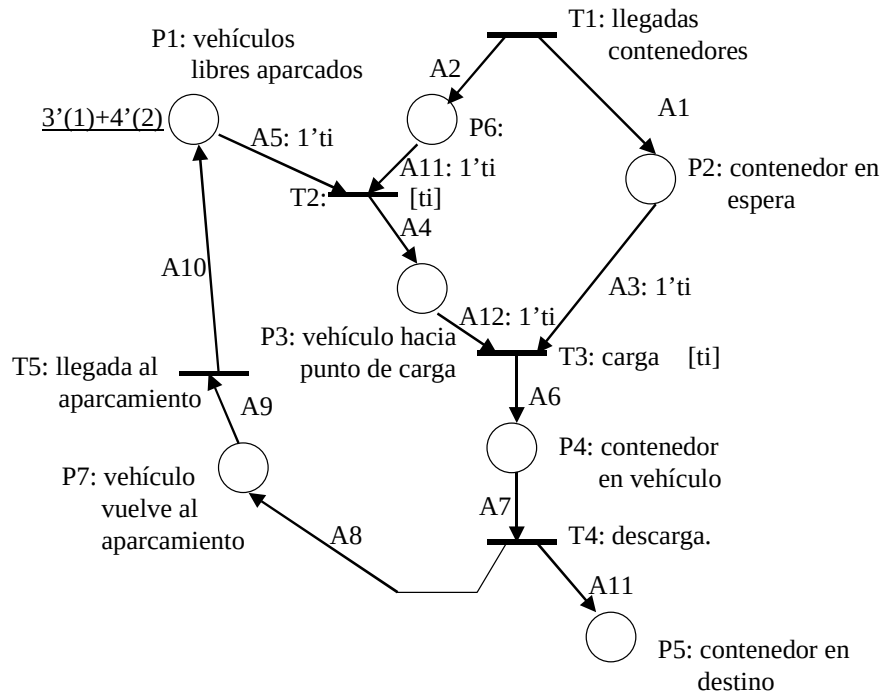


Figura B.18 Modelo del sistema de transporte y gestión de vehículos

En la RdP se separa la petición del vehículo del contenedor físico que genera la petición al llegar al punto de carga. En general, los modelos no sólo contemplan los flujos de elementos físicos, sino también los flujos administrativos.

En http://www.edicionsupc.es/poli118 (ejemplo B5) podemos observar que el código contempla, en los componentes de sincronización *MATCH*, que el vehículo interno de transporte tiene que ser del mismo tipo que el contenedor. El modelo también incorpora componentes adicionales para la obtención de estadísticas.

Los parámetros del modelo son:

- La velocidad de los vehículos es de 10 metros por minuto.
- La distancia desde el punto de aparcamiento al de carga es de 150 metros. Del punto de carga al de descarga, 200 metros. Del punto de descarga al de aparcamiento, 270 metros.
- La llegada de contenedores sigue una distribución exponencial de 10 minutos de media.
- Estadísticamente, se ha comprobado que el 40% de los contenedores es de tipo 1 y el 60% de tipo 2.

© Los autores, 2002; © Edicions UPC, 2002.

Antes de empezar a simular es necesario calcular el número de vehículos de cada tipo teóricamente necesarios. El tiempo medio de llegadas de los contenedores de tipo 1 y de tipo 2 es

$$E(Tmt1) = 10/0.4 = 25\ mn/c$$
$$E(Tmt2) = 10/0.6 = 16.6\ mn/c$$

La frecuencia media de llegadas es

$$\lambda_1 = 1/25 = 0.04\ c/mn$$
$$\lambda_2 = 1/16.6 = 0.06\ c/mn$$

El tiempo de servicio de cada vehículo es el necesario para efectuar todo el recorrido

$$E(s) = 62\ mn/p$$

y la frecuencia de servicio para cada vehículo es

$$w = 1/62 = 0.0161\ p/mn$$

el número teórico de vehículos de tipo 1 y tipo 2 es

$$nvt1 = 0.04/0.0161 = 2.48 \Rightarrow 3 \text{ vehículos tipo 1}$$
$$nvt2 = 0.06/0.0161 = 3.72 \Rightarrow 4 \text{ vehículos tipo 2}$$

El siguiente aspecto que se puede comprobar es que el sistema se comporta correctamente para esta configuración de vehículos y que si el número de vehículos disminuye en uno, el sistema no se comporta tal como se desea. Para evaluar este aspecto se puede analizar el número de contenedores en espera para ser transportados.

Para 3 vehículos de tipo 1 y 4 de tipo 2 la evolución de la cola durante una evolución de 14 días ($20*10^3$ minutos) es

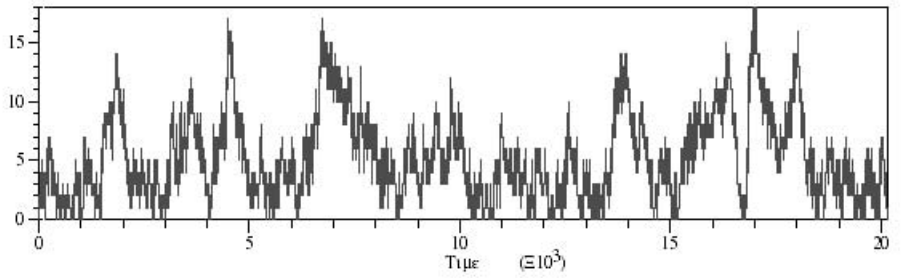


Figura B.19 Evolución de la cola de contenedores en espera para el transporte (3 y 4 vehículos)

El número promedio de contenedores que se esperan en la cola es de 5.2 y el número máximo alcanzado en esta evolución es de 18 contenedores. El número de vehículos parece que es correcto. Si reducimos el número de vehículos de tipo 1, de 3 a 2, obtenemos lan respuesta de la figura B.20.

El sistema de transporte no es capaz de responder a todas las peticiones, por lo que el número de contenedores en la cola se incrementa progresivamente. Este comportamiento es el esperado dado que el análisis teórico precisaba que el número mínimo necesario de vehículos de tipo 1 es de 3.

© Los autores, 2002; © Edicions UPC, 2002.

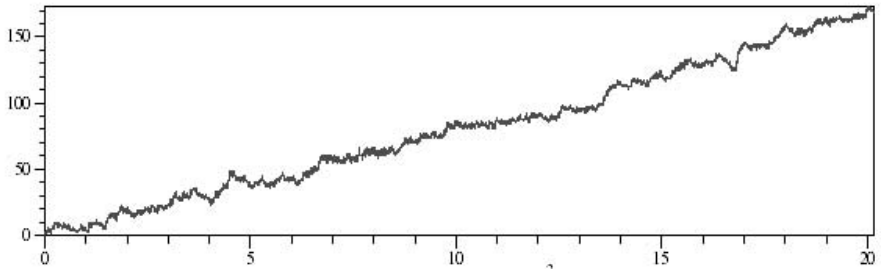


Figura B.20 Evolución de la cola de contenedores en espera para el transporte para 2 y 4 vehículos

Para el análisis de los tiempos de espera para el transporte se han efectuado 10 evoluciones diferentes, de 14 días de duración cada una. Los resultados promedios obtenidos se representan en la siguiente figura:

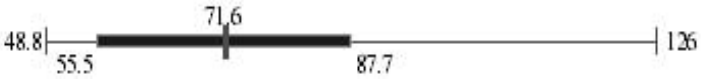


Figura B.21 Intervalo de confianza del tiempo medio de espera

El tiempo medio de espera es de 71.6 mn y hay un grado de confianza del 95% de que el tiempo promedio real se sitúe entre los 55.5 y 87.7 mn. El tiempo máximo de espera ha variado entre un mínimo de 194 mn y un máximo de 445 mn. Por último, el grado de ocupación de los vehículos es del 86%.

B.6 Sistema de vehículos autoguiados (AGV)

La figura B.22 muestra un esquema de un sistema de transporte con 4 vehículos autoguiados [PEDG95]. Los AGV toman las piezas en las intersecciones 1 y 7 y las dejan en las intersecciones 2 y 10. El tiempo entre llegadas de las piezas sigue una distribución exponencial 8 minutos de media, para las que entran por la intersección 1 y 7 minutos de media las que entran por la intersección 7. La probabilidad de que una pieza se descargue en la intersección 2 es de 0.55 y la probabilidad de que se descargue en la intersección 10 es de 0.45. Estas probabilidades son independientes del punto de entrada. El tiempo necesario para cargar o descargar una pieza al o desde el AGV sigue una distribución normal de media 2.5 minutos y desviación estándar 0.5 minutos.

Los AGV trabajan a una velocidad de 50 metros por minuto y están aparcados en el *link* c al empezar la simulación. Cuando un AGV completa la operación de descarga, comprueba si hay otra petición de carga. Si la hay, el AGV se desplaza al punto de carga; en caso contrario, se dirige a la zona de aparcamiento. Si hay peticiones desde ambos puntos de carga pendientes de servir, el AGV se dirige al punto de entrada más cercano (criterio de distancia más corta).

© Los autores, 2002; © Edicions UPC, 2002.

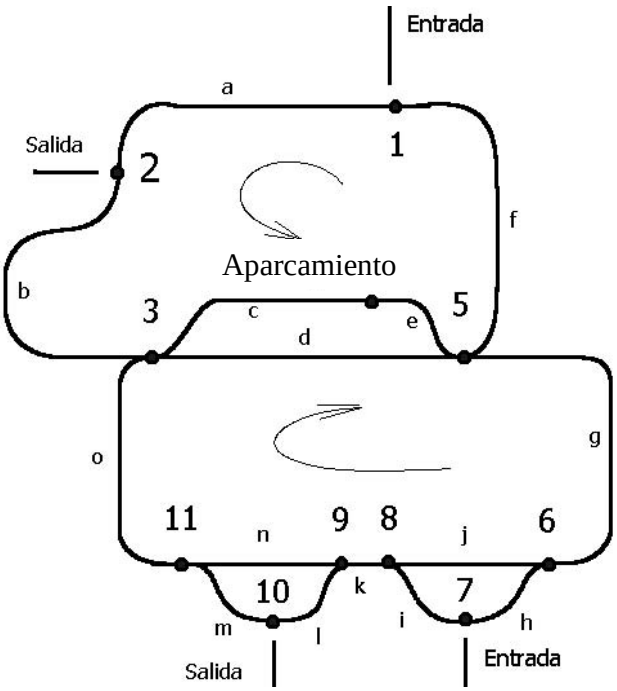


Figura B.22 Sistema de transporte por AGVs

La tabla siguiente muestra *links* que hay entre intersecciones, el número de zonas en cada *link* y la longitud de cada zona. Una zona es un tramo del recorrido que sólo admite un AGV simultáneamente. Por tanto, un AGV sólo podrá acceder a una zona si ésta está vacía.

Link	Número de zonas	Longitud de cada zona (m)
a	3	20
b	5	13
c	4	6
d	2	16
e	1	13
f	4	20
g	4	20
h	1	23
i	1	23
j	1	26
k	1	10
l	1	23
m	1	23
n	1	26
0	3	20

Tabla B.1 Lista de links

© Los autores, 2002; © Edicions UPC, 2002.

En este ejemplo no se pretende emular la RdP anterior dado que el modelo de simulación se complicará en exceso si se desea contemplar el movimiento y las interacciones entre los vehículos a lo largo de toda la red de transporte.

El objetivo perseguido es, en este caso, combinar la potencia de las herramientas de simulación para contemplar el movimiento de los AGV con la potencia de las RdP para controlar el proceso. Más concretamente, en este ejemplo particular se emplea el componentes *REQUEST* para pedir un AGV, el *TRANSPORT* para ordenar donde tiene que transportar la pieza (destino), *MOVE* para pedir que se mueva a una cierta posición y *FREE* para indicar que está libre y a punto de aceptar otra petición. En la figura siguiente se muestra la equivalencia entre la RdP y la mayoría de los componentes anteriores. Los aspectos que se deben destacar son:

- La transición T1 es codificada con el componente *REQUEST*. El flujo de cálculo seguirá cuando el transportador llegue al punto de la red de transporte especificada en el *REQUEST*.
- El lugar P3 es codificado con el componente *TRANSPORT*. Indica al vehículo que inicie el movimiento a la posición de interés de la red de transporte.
- La transición T2 es codificada con los componentes *STATION* y *FREE*. Cuando el vehículo llegue al punto de destino, el flujo del modelo de simulación seguirá a partir del bloque *STATION* referenciado.

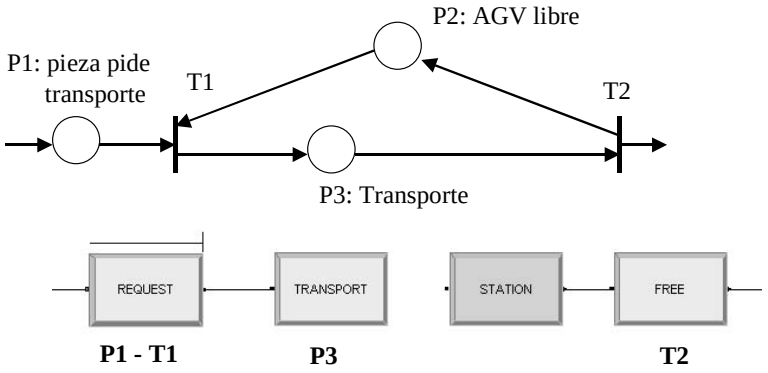


Figura B.23 Petición de vehículos y transporte

La figura siguiente muestra la RdP que controla al conjunto de los AGV. Nótese que esta red no contempla el control del movimiento a través de las zonas ni las interacciones entre los AGV. Se propone al lector construir la RdP según el supuesto de que el movimiento y las interacciones entre los AGV tienen que ser controlados por la red.

© Los autores, 2002; © Edicions UPC, 2002.

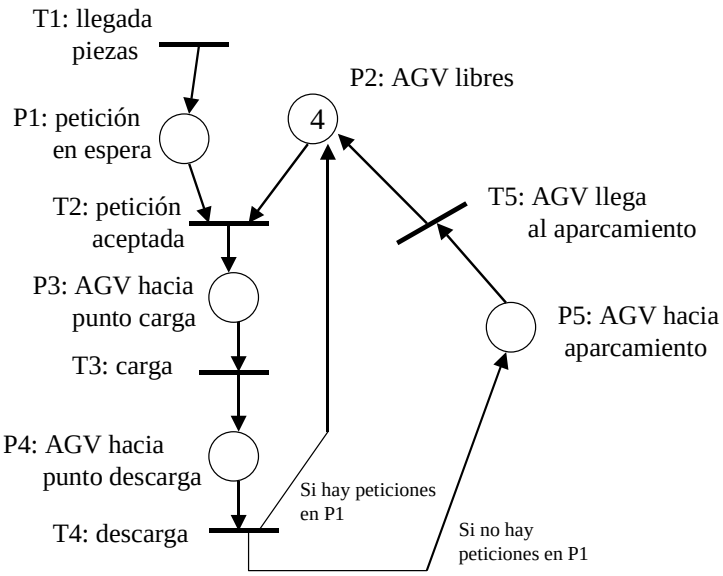


Figura B.24 Ejemplo B6

B.7 Metodología de programación

Hasta el momento hemos visto cómo a partir de una RdP o RdPC se construye un modelo de simulación. A medida que se trabaja con modelos simples el programador adquiere una cierta confianza en el uso de la herramienta de simulación y puede caer en la tentación de programar directamente el modelo sin previamente haber formalizado el problema. Así pues, en este ejemplo se aborda la problemática que surge cuando se realiza una programación *a destajo*.

Supongamos el siguiente sistema: en un subproceso de una planta de fabricación llegan dos tipos de piezas diferentes. Estas piezas, que se identifican como piezas tipo1 y tipo2, siguen dos procesos de mecanización diferentes. Las piezas tipo1 se procesan siendo mecanizadas por la máquina1 y después por la máquina2 y las piezas tipo2 se procesan sólo en la máquina2. Estas piezas llegan a un pulmón de entrada de 8 posiciones y se cargan a la máquina correspondiente por dos robots. El robot1 carga la máquina1 y el robot2 carga la máquina2. Siempre se procesa la pieza (de tipo1 o tipo2) que hace más tiempo que se espera en el pulmón. Así si llega una pieza tipo1, el subproceso sigue la siguiente secuencia:

1. Si hay espacio en el pulmón entra en éste.
2. El robot1 toma la pieza del pulmón y la carga en la máquina1.
3. La pieza se mecaniza en la máquina1.
4. El robot1 deja la pieza en el pulmón.

© Los autores, 2002; © Edicions UPC, 2002.

Para las piezas de tipo2 o las de tipo1 ya procesadas, la secuencia de procesado es más simple:

1. El robot2 toma la pieza tipo2 o tipo1 procesada y la carga en la máquina2.
2. La pieza se mecaniza en la máquina2.
3. El robot2 deja la pieza en la salida del subsistema.

El esquema de esta estación de trabajo se puede ver en la figura B.25 donde se observa el pulmón con los tipos de piezas, los robots y las máquinas.

El proceso de deducción del programador seguiría los siguientes pasos.

1. Existen 4 recursos que son: robot1, robot2, máquina1 y máquina2.
2. El pulmón se representa mediante una cola, ya que la política de servicio es FIFO.
3. El flujo a programar sigue la secuencia:
   - Si existe espacio en el pulmón, la pieza entra en el subsistema.
   - Si no se está procesando ningún elemento por la máquina correspondiente, la pieza entra directamente a procesarse; si la máquina correspondiente está ocupada, se espera en el pulmón.
   - Cuando la máquina1 termina de procesar la pieza que está tratando, el robot1 la devuelve al pulmón y toma otra, si es que hay alguna esperando.
   - Cuando la máquina2 termina de procesar la pieza que está tratando, el robot2 la elimina del subsistema y toma la siguiente pieza que esté esperando en el pulmón, si es que existe alguna.

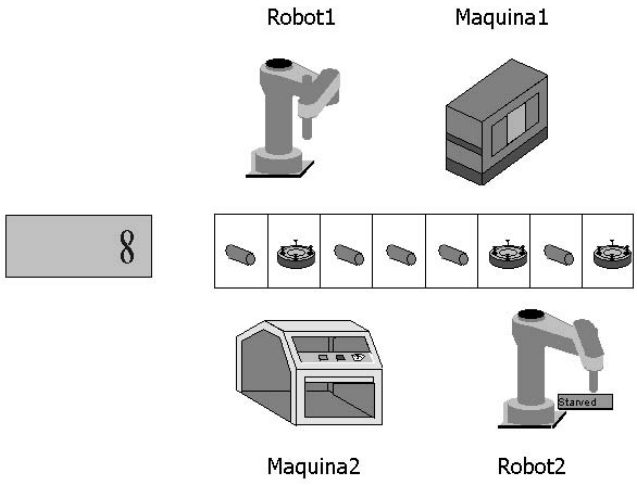


Figura B.25 Subproceso de una planta

© Los autores, 2002; © Edicions UPC, 2002.

Los recursos se programan mediante el bloque *RESOURCES,* tal como muestra la figura B.26 donde se observan los dos robots y las dos máquinas. Todos los recursos tienen capacidad 1 (*Capacity* 1) ya que solo existe un elemento de cada.

Resource - Basic Process

	Name	Type	Capacity	Busy / Hour	Idle / Hour	Per Use	StateSet Name	Failures	Report Statistics
1	Robot1	Fixed Capacity	1	0.0	0.0	0.0		0 rows	☑
2	Robot2	Fixed Capacity	1	0.0	0.0	0.0		0 rows	☑
3	Maquina1	Fixed Capacity	1	0.0	0.0	0.0		0 rows	☑
4	Maquina2	Fixed Capacity	1	0.0	0.0	0.0		0 rows	☑

Double-click here to add a new row

Figura B.26 Recursos del ejemplo B7

El pulmón se modela mediante una cola (bloque *QUEUES*), ya que éste ha de seguir una política de ordenación FIFO. En la figura B.27 se muestra esta cola.

Queue - Basic Process

	Name	Type	Shared	Report Statistics
1	Pulmoln	First In First Out	☐	☑

Figura B.27 Modelización del pulmón del ejemplo B7

El siguiente paso es la comprobación de espacio en el pulmón. Si existe espacio la pieza entra en el sistema, si no sale de éste. La programación de este paso se realiza mediante un módulo *DECIDE* tal como muestra la figura B.28. Si existe espacio (el tamaño de la cola es menor que 8) la pieza entra en el subsistema de fabricación; en caso contrario, no entra en él.

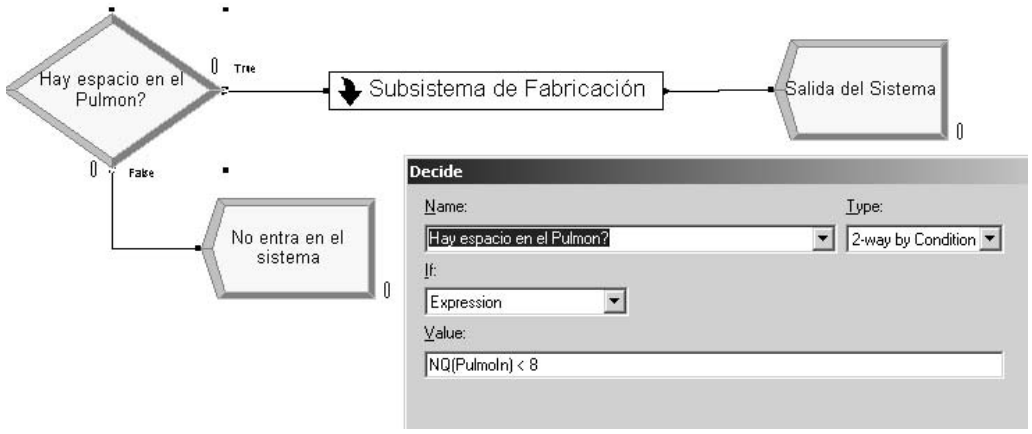


Figura B.28 Comprobación del espacio en el pulmón

Una vez ha entrado una pieza en el subsistema de fabricación se comprueba si puede pasarse a mecanizar directamente o si se tendrá que esperar en el pulmón hasta que la máquina correspondiente quede libre y el robot ya haya descargado la pieza anterior. Para implementar este mecanismo utilizamos un bloque *DECIDE* para la comprobación, seis bloques *PROCESS* para implementar los

© Los autores, 2002; © Edicions UPC, 2002.

procesos de carga/descarga y mecanizado y un bloque *Hold* para la espera en el pulmón. En la figura B.29 se puede observar el diagrama de esta parte de la solución.

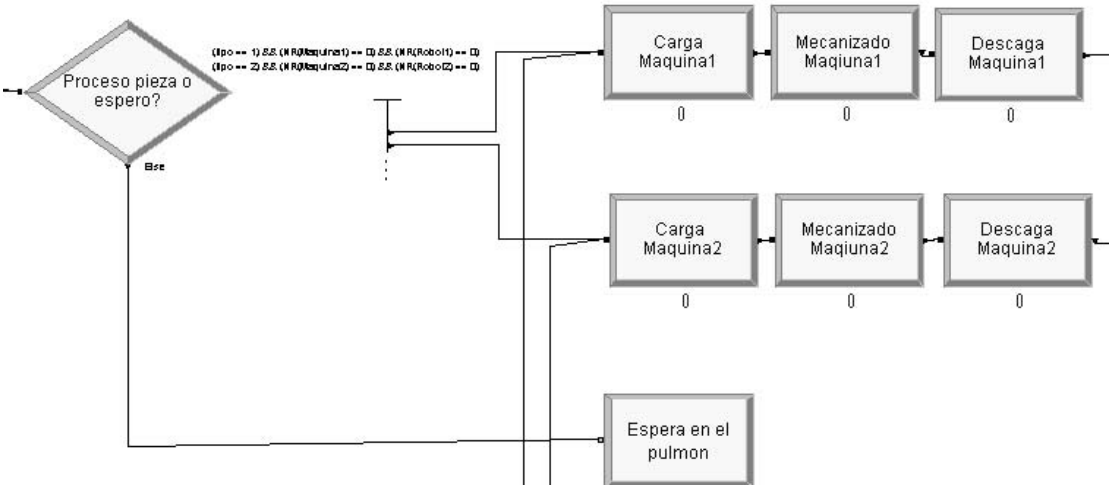


Figura B.29 Mecanizado y espera de las piezas

Una vez se ha terminado el proceso de una pieza se pasa a buscar en el pulmón si hay otra pieza para ser procesada, mediante la búsqueda en la cola del pulmón con un bloque *SEARCH*; si existe, se extrae del pulmón mediante un bloque *REMOVE*. En la figura B.30 se muestra la implementación de esta parte del modelo.

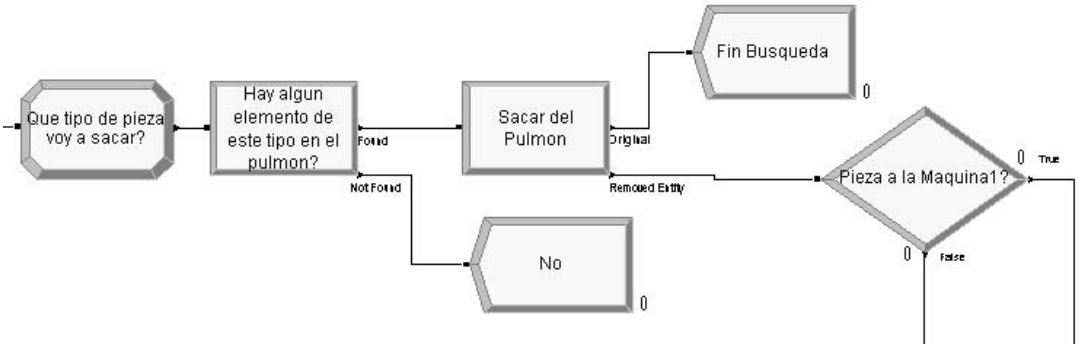


Figura B.30 Búsqueda de la siguiente pieza que se quiere mecanizar

Así pues, una vez se ha finalizado la programación del modelo se pasa al análisis de sus resultados. Sorprendentemente, al analizar el pulmón se observa que el número máximo de piezas que han estado esperando en él es de nueve (figura B.31). Para comprobar qué ha sucedido se ha insertado una gráfica de la evolución del tamaño del pulmón en el modelo, para así poder comprobar qué es lo que pasa realmente.

© Los autores, 2002; © Edicions UPC, 2002.

Pulmoln

Time	Average	Half Width	Minimum	Maximum
Waiting Time	14.2286	(Insufficient)	0.5808	31.3131

Other	Average	Half Width	Minimum	Maximum
Number Waiting	7.2312	(Insufficient)	0	9.0000

Figura B.31 Análisis de la ocupación del pulmón

Analizando la grafica del modelo se observa que este caso se repite frecuentemente. Estudiando el modelo en detalle se encuentra un error. En el sistema siempre entran piezas mientras hay espacio en el pulmón; así, si el robot1 descarga una pieza cuando la ocupación del pulmón está al límite, el número de piezas será de 9 durante un pequeño instante si hay piezas de tipo1 para procesar (figura B.32 a) y puede ser más tiempo si no hay piezas de tipo1 para procesar (se tendrá que esperar a que acabe una de tipo2 –figura B.32 b-).

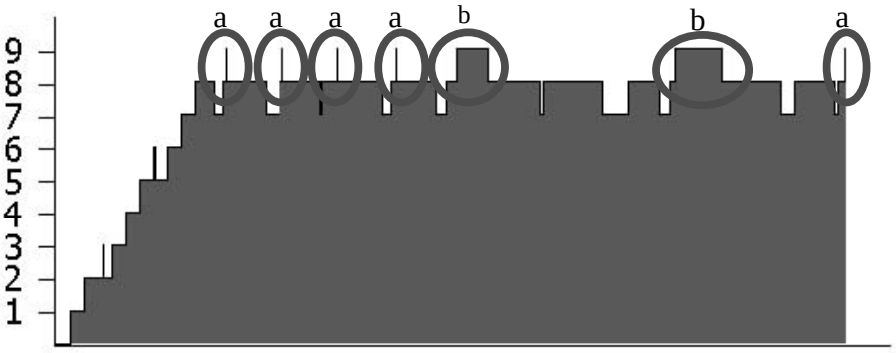


Figura B.32 Ocupación del pulmón en función del tiempo

La solución a este problema es fácil: sólo se permité descargar al robot1 si hay espacio para la pieza. La opción de limitar a 7 el pulmón no es válida ya que se malgastaría una posición de éste. Para incluir esta modificación en el modelo se inserta un bloque *SCAN* para bloquear la operación de descarga hasta que haya espacio libre en el pulmón. Con esta solución se garantiza que el robot1 nunca deje una pieza si en el pulmón hay ya 8 piezas esperando. En la figura B.33 se observa la solución a este problema.

© Los autores, 2002; © Edicions UPC, 2002.

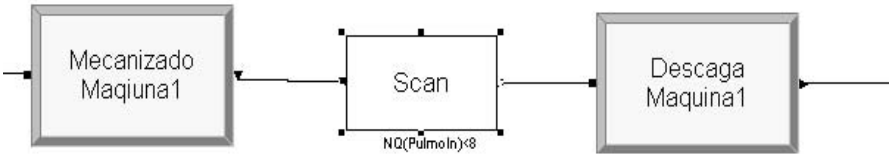


Figura B.33 Bloqueo del robot1 si hay 8 piezas en el pulmón

Una vez se ha realizado el cambio en el modelo y se estudia su comportamiento se observa de nuevo que el tamaño máximo de la cola vuelve a ser otra vez superior a 8 (figura B.34).

PulmoIn

Time	Average	Half Width	Minimum	Maximum
Waiting Time	13.7207	(Insufficient)	0.2282	32.1447
Other	**Average**	**Half Width**	**Minimum**	**Maximum**
Number Waiting	7.1332	(Insufficient)	0	9.0000

Figura B.34 Análisis de la ocupación del pulmón, bloqueando al robot1

Si se estudia la evolución del tamaño del pulmón se puede ver que se ha solucionado una parte del problema: ya no se encuentran posiciones de tipo b (figura B.32). En este caso, la ocupación del pulmón por nueve piezas es de tipo a, tal como se observa en la figura B.35. El problema surge al descargar la máquina ya que, mientras el robot comprueba el número de piezas en el pulmón antes de iniciar el proceso de descarga, las piezas que entran en el sistema no comprueban si el robot está descargando. Así, mientras el robot descarga la pieza de la máquina1 pues tienen espacio en el pulmón, puede darse el caso de que lleguen piezas al sistema que hagan sobrepasar el tamaño máximo del pulmón.

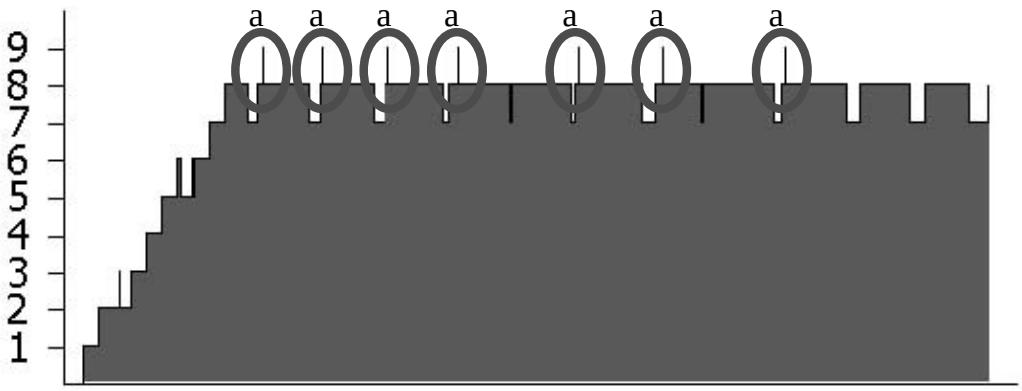


Figura B.35 Ocupación del pulmón en función del tiempo (bloqueando el robot1)

© Los autores, 2002; © Edicions UPC, 2002.

La solución viene apuntada en el párrafo anterior: si impedimos la entrada de piezas en el sistema mientras el robot está descargando y éste sólo puede descargar si tiene espacio para ello, ya se habrá eliminado el problema y no se sobrepasará nunca el tamaño máximo del pulmón. Para realizar estos cambios se ha de comprobar que el número de piezas en el pulmón sea inferior a 8 y que el robot1 no esté descargando. Este código se añade a la comprobación inicial de entrada al subsistema, tal como muestra la figura B.36.

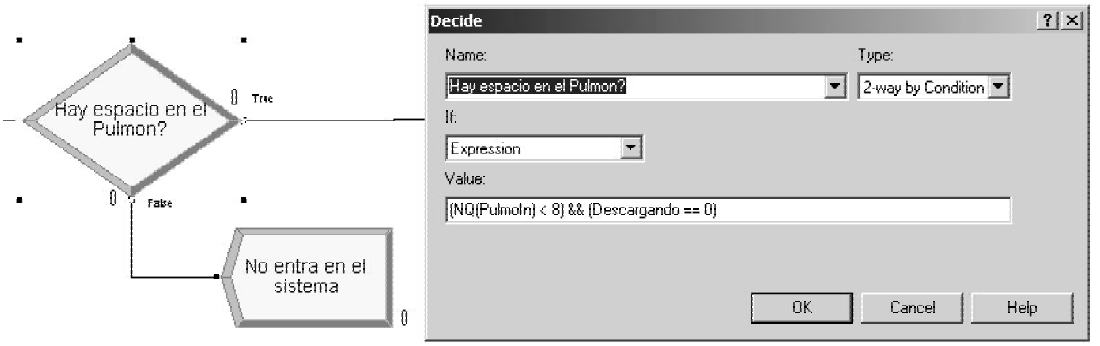


Figura B.36 Comprobación de descarga del robot1

Si analizamos los resultados del sistema se observa que ahora ya no se supera nunca el tamaño máximo del pulmón, figura B.37.

PulmoIn

Time	Average	Half Width	Minimum	Maximum
Waiting Time	9.5009	(Insufficient)	2.9832	25.5244

Other	Average	Half Width	Minimum	Maximum
Number Waiting	7.1835	(Insufficient)	0	8.0000

Figura B.37 Análisis de la ocupación del pulmón, bloqueando al robot1 y comprobando su estado

Pero el sistema tiene ahora un problema aún mucho más grave que todos los anteriores. Tal como muestra la gráfica de evolución del pulmón (figura B.38), el sistema se bloquea. Supongamos que en el pulmón hay 7 piezas de tipo1 mientras la máquina está procesando otra, y llega la octava pieza que también es de tipo1. Cuando la máquina termine de procesar la pieza el robot la querrá descargar pero no podrá, ya que no tendrá espacio en el pulmón, y se quedará bloqueado, esperando que se libere un espacio. Este espacio no se liberará nunca, debido a que todas las piezas son de tipo1. Además no podrán entrar piezas nuevas en el sistema.

© Los autores, 2002; © Edicions UPC, 2002.

Figura B.38 Ocupación del pulmón, bloqueando el robot1 y comprobando su estado

En este pequeño ejemplo se puede identificar cuáles son los problemas que van surgiendo, al seguir un método de programación *a destajo* se producen efectos no deseados, como puede ser la entrada en una espiral de errores (el ejemplo que nos ocupa), cuando al intentar solucionar un problema se genera otro. También se puede acabar programando por el método de prueba y error, con lo que al final el programador ya no sabe si el modelo cumple con las especificaciones del sistema, puesto que al iterar el proceso de programación se descuida la funcionalidad. Y el peor de los problemas que puede ocasionar este tipo de programación es el llegar a la inabarcabilidad del problema debido al gran número de modificaciones que se han realizado sobre la marcha, con lo que además se suman problemas de falta de mantenibilidad, imposibilidad de reutilización del código, etc. Imagínese el lector un sistema de fabricación con decenas de recursos a utilizar y cientos de eventos. Todos estos posibles problemas realzan la necesidad de utilización de formalismos para la modelización de los sistemas y para su programación en un simulador. Este mismo ejemplo, formalizado mediante RdPC, evita la aparición de problemas añadidos debidos a una mala estructuración de la programación. Se propone como ejercicio al lector que modelize el problema mediante una RdPC y la implemente en un simulador.

© Los autores, 2002; © Edicions UPC, 2002.

C Tablas

C.1 Función de distribución normal estándar

	0	**0.01**	**0.02**	**0.03**	**0.04**	**0.05**	**0.06**	**0.07**	**0.08**	**0.09**
0	0.5	0.504	0.508	0.512	0.516	0.5199	0.5239	0.5279	0.5319	0.5359
0.1	0.5398	0.5438	0.5478	0.5517	0.5557	0.5596	0.5636	0.5675	0.5714	0.5753
0.2	0.5793	0.5832	0.5871	0.591	0.5948	0.5987	0.6026	0.6064	0.6103	0.6141
0.3	0.6179	0.6217	0.6255	0.6293	0.6331	0.6368	0.6406	0.6443	0.648	0.6517
0.4	0.6554	0.6591	0.6628	0.6664	0.67	0.6736	0.6772	0.6808	0.6844	0.6879
0.5	0.6915	0.695	0.6985	0.7019	0.7054	0.7088	0.7123	0.7157	0.719	0.7224
0.6	0.7257	0.7291	0.7324	0.7357	0.7389	0.7422	0.7454	0.7486	0.7517	0.7549
0.7	0.758	0.7611	0.7642	0.7673	0.7704	0.7734	0.7764	0.7794	0.7823	0.7852
0.8	0.7881	0.791	0.7939	0.7967	0.7995	0.8023	0.8051	0.8078	0.8106	0.8133
0.9	0.8159	0.8186	0.8212	0.8238	0.8264	0.8289	0.8315	0.834	0.8365	0.8389
1	0.8413	0.8438	0.8461	0.8485	0.8508	0.8531	0.8554	0.8577	0.8599	0.8621
1.1	0.8643	0.8665	0.8686	0.8708	0.8729	0.8749	0.877	0.879	0.881	0.883
1.2	0.8643	0.8665	0.8686	0.8708	0.8729	0.8749	0.877	0.879	0.881	0.883
1.3	0.9032	0.9049	0.9066	0.9082	0.9099	0.9115	0.9131	0.9147	0.9162	0.9177
1.4	0.9192	0.9207	0.9222	0.9236	0.9251	0.9265	0.9279	0.9292	0.9306	0.9319
1.5	0.9332	0.9345	0.9357	0.937	0.9382	0.9394	0.9406	0.9418	0.9429	0.9441
1.6	0.9452	0.9463	0.9474	0.9484	0.9495	0.9505	0.9515	0.9525	0.9535	0.9545
1.7	0.9554	0.9564	0.9573	0.9582	0.9591	0.9599	0.9608	0.9616	0.9625	0.9633
1.8	0.9641	0.9649	0.9656	0.9664	0.9671	0.9678	0.9686	0.9693	0.9699	0.9706
1.9	0.9713	0.9719	0.9726	0.9732	0.9738	0.9744	0.975	0.9756	0.9761	0.9767
2	0.9772	0.9778	0.9783	0.9788	0.9793	0.9798	0.9803	0.9808	0.9812	0.9817
2.1	0.9821	0.9826	0.983	0.9834	0.9838	0.9842	0.9846	0.985	0.9854	0.9857
2.2	0.9861	0.9864	0.9868	0.9871	0.9875	0.9878	0.9881	0.9884	0.9887	0.989
2.3	0.9893	0.9896	0.9898	0.9901	0.9904	0.9906	0.9909	0.9911	0.9913	0.9916
2.4	0.9918	0.992	0.9922	0.9925	0.9927	0.9929	0.9931	0.9932	0.9934	0.9936
2.5	0.9938	0.994	0.9941	0.9943	0.9945	0.9946	0.9948	0.9949	0.9951	0.9952
2.6	0.9953	0.9955	0.9956	0.9957	0.9959	0.996	0.9961	0.9962	0.9963	0.9964
2.7	0.9965	0.9966	0.9967	0.9968	0.9969	0.997	0.9971	0.9972	0.9973	0.9974
2.8	0.9974	0.9975	0.9976	0.9977	0.9977	0.9978	0.9979	0.9979	0.998	0.9981
2.9	0.9981	0.9982	0.9982	0.9983	0.9984	0.9984	0.9985	0.9985	0.9986	0.9986
3	0.9987	0.9987	0.9987	0.9988	0.9988	0.9989	0.9989	0.9989	0.999	0.999
3.1	0.999	0.9991	0.9991	0.9991	0.9992	0.9992	0.9992	0.9992	0.9993	0.9993
3.2	0.9993	0.9993	0.9994	0.9994	0.9994	0.9994	0.9994	0.9995	0.9995	0.9995
3.3	0.9995	0.9995	0.9995	0.9996	0.9996	0.9996	0.9996	0.9996	0.9996	0.9997
3.4	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	0.9998
3.5	0.9998	0.9998	0.9998	0.9998	0.9998	0.9998	0.9998	0.9998	0.9998	0.9998
3.6	0.9998	0.9998	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999
3.7	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999
3.8	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999

© Los autores, 2002; © Edicions UPC, 2002.

C.2 Puntos críticos de la distribución de Chi-cuadrado

gl	0.005	0.01	0.025	0.05	0.1
1	7.879	6.635	5.024	3.841	2.706
2	10.597	9.21	7.378	5.991	4.605
3	12.838	11.345	9.348	7.815	6.251
4	14.86	13.277	11.143	9.488	7.779
5	16.75	15.086	12.833	11.07	9.236
6	18.548	16.812	14.449	12.592	10.645
7	20.278	18.475	16.013	14.067	12.017
8	21.955	20.09	17.535	15.507	13.362
9	23.589	21.666	19.023	16.919	14.684
10	25.188	23.209	20.483	18.307	15.987
11	26.757	24.725	21.92	19.675	17.275
12	28.3	26.217	23.337	21.026	18.549
13	29.819	27.688	24.736	22.362	19.812
14	31.319	29.141	26.119	23.685	21.064
15	32.801	30.578	27.488	24.996	22.307
16	34.267	32	28.845	26.296	23.542
17	35.718	33.409	30.191	27.587	24.769
18	37.156	34.805	31.526	28.869	25.989
19	38.582	36.191	32.852	30.144	27.204
20	39.997	37.566	34.17	31.41	28.412
21	41.401	38.932	35.479	32.671	29.615
22	42.796	40.289	36.781	33.924	30.813
23	44.181	41.638	38.076	35.172	32.007
24	45.559	42.98	39.364	36.415	33.196
25	46.928	44.314	40.646	37.652	34.382
26	48.29	45.642	41.923	38.885	35.563
27	49.645	46.963	43.195	40.113	36.741
28	50.993	48.278	44.461	41.337	37.916
29	52.336	49.588	45.722	42.557	39.087
30	53.672	50.892	46.979	43.773	40.256
40	66.766	63.691	59.342	55.758	51.805
50	79.49	76.154	71.42	67.505	63.167
60	91.952	88.379	83.298	79.082	74.397
70	104.215	100.425	95.023	90.531	85.527
80	116.321	112.329	106.629	101.879	96.578
90	128.299	124.116	118.136	113.145	107.565
100	140.169	135.807	129.561	124.342	118.498

© Los autores, 2002; © Edicions UPC, 2002.

C.3 Puntos críticos de la distribución t de Student

$\upsilon \backslash \alpha$	**0.10**	**0.05**	**0.025**	**0.01**	**0.001**
1	3.078	6.314	12.706	31.821	318.313
2	1.886	2.920	4.303	6.965	22.327
3	1.638	2.353	3.182	4.541	10.215
4	1.533	2.132	2.776	3.747	7.173
5	1.476	2.015	2.571	3.365	5.893
6	1.440	1.943	2.447	3.143	5.208
7	1.415	1.895	2.365	2.998	4.782
8	1.397	1.860	2.306	2.896	4.499
9	1.383	1.833	2.262	2.821	4.296
10	1.372	1.812	2.228	2.764	4.143
11	1.363	1.796	2.201	2.718	4.024
12	1.356	1.782	2.179	2.681	3.929
13	1.350	1.771	2.160	2.650	3.852
14	1.345	1.761	2.145	2.624	3.787
15	1.341	1.753	2.131	2.602	3.733
16	1.337	1.746	2.120	2.583	3.686
17	1.333	1.740	2.110	2.567	3.646
18	1.330	1.734	2.101	2.552	3.610
19	1.328	1.729	2.093	2.539	3.579
20	1.325	1.725	2.086	2.528	3.552
21	1.323	1.721	2.080	2.518	3.527
22	1.321	1.717	2.074	2.508	3.505
23	1.319	1.714	2.069	2.500	3.485
24	1.318	1.711	2.064	2.492	3.467
25	1.316	1.708	2.060	2.485	3.450
26	1.315	1.706	2.056	2.479	3.435
27	1.314	1.703	2.052	2.473	3.421
28	1.313	1.701	2.048	2.467	3.408
29	1.311	1.699	2.045	2.462	3.396
30	1.310	1.697	2.042	2.457	3.385
40	1.303	1.684	2.021	2.423	3.307
50	1.299	1.676	2.009	2.403	3.261
60	1.296	1.671	2.000	2.390	3.232
70	1.294	1.667	1.994	2.381	3.211
80	1.292	1.664	1.990	2.374	3.195
90	1.291	1.662	1.987	2.368	3.183
99	1.290	1.660	1.984	2.365	3.175
100	1.290	1.660	1.984	2.364	3.174
∞	1.282	1.645	1.960	2.326	3.090

© Los autores, 2002; © Edicions UPC, 2002.