

TESIS

33

54

03 FEB. 1998

P.D. #12m

PLANIFICACIÓN DE LA PRODUCCIÓN EN SISTEMAS DE FABRICACIÓN FLEXIBLE. MODELOS, APLICACIONES Y MÉTODOS DE RESOLUCIÓN

Autor: Fernando Guerrero López
Ingeniero Industrial por la E. S. I. de Sevilla

T.162

presentada en la ESCUELA SUPERIOR DE INGENIEROS de la
UNIVERSIDAD DE SEVILLA

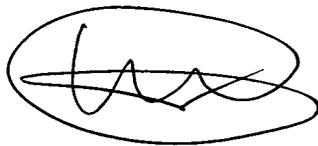
para la obtención del Grado de Doctor Ingeniero Industrial

Director: Prof. Sebastián Lozano Segura

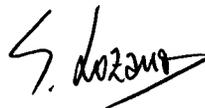
Sevilla, Febrero de 1998

TESIS

**PLANIFICACIÓN DE LA PRODUCCIÓN EN
SISTEMAS DE FABRICACIÓN FLEXIBLE.
MODELOS, APLICACIONES Y MÉTODOS DE
RESOLUCIÓN**



Doctorando: Fernando Guerrero López



Director: Prof. Sebastián Lozano Segura

T. 162

ESCUELA SUPERIOR DE INGENIEROS
DE SEVILLA BIBLIOTECA

A mis padres

ÍNDICE

Agradecimientos	v
Resumen	vi
Lista De Figuras	viii
Lista De Tablas	ix

CAPÍTULO 1. PLANIFICACIÓN DE LA PRODUCCIÓN EN LOS SFF

1.1. Sistemas de fabricación flexible	1
1.2. Diseño, planificación y gestión de los SFF	3
1.3. Aproximaciones existentes para el problema de carga	5
1.3.1. Modelos de carga	5
1.3.2. Modelos de carga y agrupación de máquinas	6
1.3.3. Modelos de carga y selección de ítems	6
1.3.4. Modelos de carga y secuenciación	7
1.3.5. Modelos de asignación de herramientas y operaciones	7
1.3.6. Resumen	8
1.4. Motivación y planteamiento de la investigación	11

CAPÍTULO 2. PROBLEMAS DE CARGA Y SELECCIÓN DE ÍTEMS

2.1. Introducción	12
2.2. Formulación del problema de carga	12
2.2.1. Suposiciones	12
2.2.2. Notación	13
2.2.3. Función objetivo y restricciones	13
2.3. Formulación del problema de selección de ítems	15
2.4. Ilustración	16
2.4.1. Datos de entrada	17
2.4.2. Carga de máquinas	18
2.4.3. Selección de ítems	19
2.5. Resumen	20

CAPÍTULO 3. SISTEMA DE COSTES PARA SISTEMAS DE FABRICACIÓN FLEXIBLE

3.1. Introducción	21
3.2. Motivación	21

3.3. Costes basados en actividades	22
3.4. Diseño del sistema de costes para SFF	22
3.4.1. Descripción del escenario	22
3.4.2. Modelo de planificación de la producción	23
3.4.3. Modelo de simulación	25
3.4.4. Sistema de costes propuesto	26
3.5. Ilustración	29
3.5.1. Datos de entrada	29
3.5.2. Estimación de costes.....	30
3.6. Experimentos adicionales	32
3.6.1. Análisis de sensibilidad	32
3.6.2. Discusión	35
3.7. Resumen	35
CAPÍTULO 4. SOLUCIONES HEURÍSTICAS PARA EL PROBLEMA DE CARGA	
4.1. Introducción	36
4.2. Soluciones heurísticas	36
4.2.1. Pasos comunes	36
4.2.2. Heurística 1	39
4.2.3. Heurística 2	40
4.3. Experimentación	40
4.3.1. Primera fase	41
4.3.1.1. Resultados	42
4.3.1.2. Discusión	46
4.3.2. Segunda fase	46
4.3.1.1. Resultados	46
4.3.1.2. Discusión	50
4.4. Resumen	51
CAPÍTULO 5. ESTRATEGIAS DE BÚSQUEDA EN LA VECINDAD	
5.1. Introducción	52
5.2. Búsqueda en la vecindad.....	52
5.2.1. Codificación de soluciones.....	52
5.2.2. Estructura de la vecindad	53

5.2.3. Evaluación de soluciones	54
5.2.4. Solución inicial	57
5.2.4.1. Búsqueda aleatoria sesgada	57
5.3. Búsqueda Tabú.....	59
5.3.1. Diseño del algoritmo	59
5.3.2. Experimentación.....	61
5.3.2.1. Parámetros del algoritmo	61
5.3.2.2. Resultados	61
5.3.2.3. Discusión	65
5.4. Algoritmo mixto	66
5.4.1. Diseño del algoritmo	67
5.4.2. Experimentación.....	67
5.4.2.1. Parámetros del algoritmo	67
5.4.2.2. Resultados	68
5.4.2.3. Discusión	70
5.5. Resumen	70

CAPÍTULO 6. DISEÑO Y CARGA DE CÉLULAS DE FABRICACIÓN FLEXIBLE

6.1. Introducción	71
6.2. Tecnología de Grupos Generalizada	71
6.3. Metodología propuesta	74
6.3.1. Notación.....	74
6.3.2. Planes de proceso como partes independientes.....	75
6.3.2.1. Red neuronal auto-organizativa	75
6.3.2.2. Algoritmo de enfriamiento basado en los campos medios.....	78
6.3.3. Formación de células basada en técnicas difusas	80
6.3.3.1. Algoritmo basado en las pertenencias de máquinas a células.....	80
6.3.3.2. Algoritmo basado en las pertenencias de partes a células	81
6.3.4. Carga de las células	82
6.4. Ilustración	83
6.4.1. Diseño de células	83
6.4.1.1. Datos de entrada.....	83
6.4.1.2. Planes de proceso como partes independientes	84

6.4.1.3. Planes de proceso difusos.....	88
6.4.2. Carga de células.....	89
6.4.2.1. Configuración de ZODIAC	90
6.4.2.2. Configuración de la RNA y los algoritmos difusos	91
6.5. Experimentos adicionales	91
6.5.1. Análisis de sensibilidad en el modelo de carga	92
6.5.1.1. Función objetivo.....	92
6.5.1.2. Mix de rutas	93
6.5.2. Comparación de la RNA con otras heurísticas	94
6.6. Resumen	96
CAPÍTULO 7. CONCLUSIONES Y FUTURAS LÍNEAS DE INVESTIGACIÓN	
7.1. Conclusiones y aportaciones.....	98
7.1.1. Sistemas de Fabricación Flexible	98
7.1.2. Problemas de carga y selección de ítems.....	98
7.1.3. Sistema de costes para Sistemas de Fabricación Flexible	99
7.1.4. Soluciones heurísticas para el problema de carga.....	99
7.1.5. Estrategias de búsqueda en la vecindad.....	99
7.1.6. Diseño y carga de células de fabricación flexible	100
7.2. Posibles extensiones	100
ANEXO I. GENERACIÓN DE PROBLEMAS DE CARGA	
I.1. Introducción	102
I.2. Primer conjunto de problemas	102
I.3. Segundo conjunto de problemas	103
I.4. Modelado del equilibrado de las cargas	105
I.5. Evaluación de las funciones objetivo	106
ANEXO II. IMPLEMENTACIÓN DEL SISTEMA DE COSTES	
II.1. Introducción.....	110
II.2. Implementación.....	110
REFERENCIAS	114

AGRADECIMIENTOS.

En primer lugar deseo expresar mi gratitud al Prof. D. Sebastián Lozano Segura, director de esta Tesis, por su confianza, constante seguimiento, ideas, exigencia y minuciosidad. Los posibles resultados de esta Tesis son en gran medida debidos a él.

Mi agradecimiento igualmente a D. Ignacio Eguía Salinas y D. Jesús Racero Moreno. Al primero por facilitarme el código de los algoritmos ZODIAC y MST usados en el capítulo 6 de este trabajo. Al segundo por su ayuda en la implementación de varios de los algoritmos diseñados a lo largo de la Tesis.

Por último, agradezco a los Profesores Canca, Onieva y Larrañeta su ánimo y apoyo constantes.

RESUMEN.

Esta tesis gira en torno a la problemática de carga de máquinas en los sistemas de fabricación flexible. En primer lugar, se modela el problema mediante programación lineal entera mixta. Las variables de decisión son las referidas al mix agregado de rutas para cada uno de los ítems a procesar y las correspondientes a la asignación de herramientas a máquinas. En cuanto a las restricciones, se tienen en cuenta las de capacidad de los almacenes de herramientas de las máquinas, limitación del número de copias disponibles y del tiempo de vida de cada tipo de herramienta. El criterio usado es el de equilibrar las cargas de trabajo de las máquinas. El modelo de carga es extendido para englobar, además de las anteriores, las decisiones referidas a los ítems a procesar conjuntamente. En esta segunda formulación, se usa como criterio la maximización de la suma de prioridades de los ítems seleccionados. El equilibrado de las cargas es modelado mediante restricciones en los desequilibrios de las cargas de las máquinas.

La funcionalidad de las formulaciones propuestas es ilustrada no sólo en el contexto de la planificación de la producción, sino que se enmarcan en un contexto más amplio. Así, se diseña un sistema ABC para la estimación de los costes totales de producción en los sistemas de fabricación flexible, donde los costes indirectos se imputan a los distintos ítems de acuerdo al consumo de los recursos de producción por parte de las actividades requeridas para el procesado de los mismos. El sistema de costes adapta de una forma dinámica los costes, a las distintas formas de operación, a través de su enlace con la fase de planificación de la producción en la que se usa una modificación de la formulación propuesta para el problema de selección de ítems.

Posteriormente, se aborda la resolución eficiente del modelo de carga propuesto. Debido a la integridad de las variables que modelan la asignación de herramientas a máquinas, así como a las múltiples restricciones de la mochila presentes en la formulación matemática, en el peor de los casos, los métodos estándares de ramificación y acotación requieren un elevado tiempo de computación. Como consecuencia, es necesario abordar su resolución mediante técnicas heurísticas. Inicialmente, se proponen dos procedimientos basados en relajaciones continuas del problema original. En ambos métodos la problemática global es descompuesta en dos etapas. En la primera etapa se fija una asignación de herramientas a máquinas, y en la segunda, se determina un mix agregado de rutas compatible con la misma. Los resultados muestran que, aún sin garantizar admisibilidad, los dos procedimientos ofrecen soluciones para un gran número de problemas de diversa dificultad, con muy bajos requerimientos computacionales.

La persistencia de problemas de elevada dificultad, para los que los procedimientos heurísticos anteriores no ofrecen solución admisible, plantea la necesidad de recurrir a técnicas metaheurísticas. En esta línea, se diseñan los elementos necesarios para estrategias de búsqueda en la vecindad. A partir de una codificación entera de soluciones, y de una estrategia de oscilación en torno a la frontera de admisibilidad, introducida mediante una modificación en la evaluación de soluciones, se exploran distintas asignaciones de herramientas a máquinas. A continuación, los elementos anteriores se introducen en esquemas de Búsqueda Tabú para controlar la evolución de la búsqueda. Se diseña un algoritmo Tabú que divide la búsqueda en dos fases. La primera fase consiste en buscar la primera solución admisible para el problema original. En ella se hacen uso de un mecanismo de diversificación. En la segunda fase, se hace una búsqueda intensiva para mejorar la calidad de las soluciones. El algoritmo diseñado se evalúa para un conjunto de problemas, mostrando los resultados que el método diseñado ofrece buenas soluciones consistentemente para todos los problemas considerados. Además, se presenta un análisis de las soluciones ofrecidas por el algoritmo al variar la

solución inicial. También se presenta una modificación sobre el algoritmo Tabú donde, la primera fase es implementada mediante un esquema basado en técnicas de Recocido Simulado con la idea de acelerar la obtención de la primera solución admisible. No obstante, los resultados muestran que esto se consigue a costa de alargar la segunda fase.

Finalmente, se aborda el diseño y la carga de células de fabricación flexible. Se propone una metodología que descompone la problemática en dos etapas. En la primera etapa, se diseñan las células de producción. En la segunda etapa, se cargan las células y las máquinas que las componen. Para la etapa de diseño de las células se proponen dos aproximaciones. La primera está basada en considerar cada uno de los planes de proceso de las partes a procesar como una parte (virtual) independiente. Se diseña una Red Neuronal Auto-organizativa (RNA) para resolver el problema de clusterización planteado, comparando los resultados ofrecidos por la misma, con los de otros procedimientos heurísticos. Los resultados muestran que la RNA ofrece las mejores soluciones en unos tiempos de computación razonables. La segunda aproximación propuesta para la fase de diseño se basa en la construcción de un único plan de proceso difuso para cada parte, a partir de los planes alternativos de la misma. Se diseñan dos procedimientos de clusterización basado en el algoritmo de las C-medias. Para la etapa de carga de las células se propone un modelo de programación lineal continua que es una modificación de la formulación general utilizada.

LISTA DE FIGURAS

1.1.	Componentes básicos de los SFF	1
1.2.	Estación de trabajo	2
1.3.	Clasificación según volumen de producción y flexibilidad	2
1.4.	Estructura Jerárquica de Planificación de Van Looveren et al (1986).....	4
1.5.	Selección de ítems a procesar y carga de máquinas.....	4
3.1.	Diagrama de Ciclos de Actividades del sistema	25
3.2.	Sistema de costes para Sistemas de Fabricación Flexible	26
4.1.	Procedimiento PROC_ADM	38
4.2.	Pseudocódigo para el cálculo del mix de rutas en heurística 2	40
5.1.	Codificación de soluciones.....	53
5.2.	Jerarquía entre tipos de soluciones.....	54
5.3.	Evaluación de soluciones según el tipo	57
5.4.	Búsqueda aleatoria sesgada.....	58
5.5.	Evolución del algoritmo Tabú para H1 y H2. Problema 0121	65
5.6.	Evolución del algoritmo Tabú para RC y BAS. Problema 0121	65
5.7.	Evolución del algoritmo mixto. Problema 0121	68
6.1.	Arquitectura de la Red Neuronal Auto-organizativa	76
6.2.	Pseudocódigo del algoritmo de Enfriamiento basado en los Campos Medios.....	79
6.3.	Evolución de la función objetivo (6.21)	88
6.4.	Evolución de la función objetivo (6.27)	89
6.5.	Resultados de sensibilidad.....	92
I.1.	Variación del NPE con los parámetros fundamentales.....	107
I.2.	Variación del CVR de M3 con los parámetros fundamentales	108
I.3.	Variación del tiempo de CPU de M3 con el número de variables binarias	109
II.1.	Diagrama de contexto de la aplicación.....	110
II.2.	Menú principal de la aplicación.....	111
II.3.	Salida del optimizador: pedidos seleccionados y mix de rutas.....	112
II.4.	Salida del optimizador: carga de máquinas.....	112
II.5.	Salida del simulador	113

LISTA DE TABLAS

1.1.	Clasificación de las referencias según problemática abordada y criterio usado ...	9
1.2.	Clasificación de las referencias según las características del modelo	10
2.1.	Notación usada en los modelos	13
2.2.	Datos de las estaciones.....	17
2.3.	Datos de los ítems.....	17
2.4.	Datos de las herramientas	17
2.5.	Rutas de los ítems.....	18
2.6.	Asignación de herramientas y carga de máquinas.....	18
2.7.	Ítems con requerimientos de producción	19
2.8.	Ítems seleccionados y mix de rutas	19
2.9.	Asignación de las herramientas y carga de máquinas (selección de ítems)	20
3.1.	Datos de los pedidos	29
3.2.	Rutas alternativas de los ítems.....	30
3.3.	Pedidos seleccionados y mix de rutas.....	31
3.4.	Asignación de herramientas y utilización de máquinas.....	31
3.5.	Bases y tasas de asignación de los costes indirectos.....	31
3.6.	Costes unitarios de producción	32
3.7.	Definición de escenarios.....	32
3.8.	Costes unitarios de producción para los nuevos escenarios.....	33
3.9.	Cambios en el tiempo total de producción (makespan).....	34
3.10.	Cambios en las bases y tasas de asignación.....	34
4.1.	Notación en heurísticas.....	37
4.2.	Resultados de experimentación. Fase 1	43
4.3.	Resultados de experimentación. Fase 2.....	47
4.4.	Resumen de resultados. Fase 2.....	50
4.5.	Clasificación de problemas. Fase 2.....	51
5.1.	Resultados del algoritmo de BT. Sensibilidad respecto a la solución inicial	63
5.2.	Tiempos de computación de BT. Sensibilidad respecto a la solución inicial	64
5.3.	Resultados del algoritmo mixto.....	69
6.1.	Descripción de planes de proceso	84
6.2.	Matriz de incidencias máquina-plan de proceso.....	85
6.3.	Resultados de ZODIAC: células y familias	86
6.4.	Resultados de la RNA ($p = 0.1$): células y familias.....	87
6.5.	Pertenencias de máquinas a células (χ_{ks}).....	88

6.6.	Centroides (ψ_{ks})	89
6.7.	Requerimientos de producción para el período actual.....	89
6.8.	Rutas de procesado para el período actual ($\alpha=0.25, \beta=0.15$) (I).....	90
6.9.	Cargas de trabajo para el período actual ($\alpha=0.25, \beta=0.15$) (I)	90
6.10.	Rutas de procesado para el período actual ($\alpha=0.25, \beta=0.15$) (II).....	91
6.11.	Cargas de trabajo para el período actual ($\alpha=0.25, \beta=0.15$) (II)	91
6.12.	Requerimientos de producción para el próximo período	93
6.13.	Rutas de procesado para el próximo período ($\alpha=0.25, \beta=0.15$) (I)	93
6.14.	Cargas de trabajo para el próximo período ($\alpha=0.25, \beta=0.15$) (II)	94
6.15.	Problemas de la literatura.....	94
6.16.	Resultados de la RNA, MST y BT ($p = 0.1$)	96
I.1.	Parámetros para la generación de problemas (fase 1)	102
I.2.	Parámetros para la generación de problemas (fase 2)	103
I.3.	Modelado del objetivo de equilibrar la carga de trabajo de las máquinas.....	106
I.4.	Número de problemas equilibrado con cada función objetivo	107
I.5.	Coficiente de variación medio con cada función objetivo	108
I.6.	CVR medio con cada FO y número de veces que el CV es mínimo.....	108
I.7.	IRM y NM en las ocho funciones objetivo	109

1.1. SISTEMAS DE FABRICACIÓN FLEXIBLE.

Un Sistema de Fabricación Flexible (SFF) puede ser definido como un sistema de producción controlado por computador capaz de producir eficientemente una variedad de tipos de piezas demandadas en cantidades variables. Los componentes básicos de los SFF son:

- Estaciones de trabajo (ET1-ET8).
- Estaciones de carga/descarga (EC/D).
- Sistemas de transportes para piezas y herramientas (SMM).
- Sistema de control por computador (SCC).

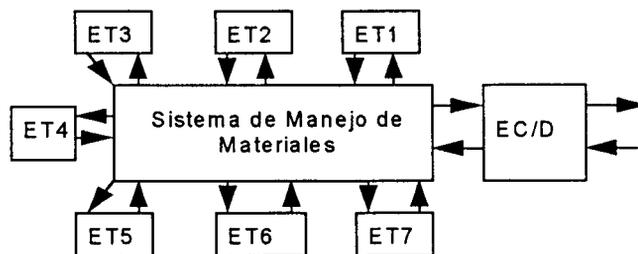


Figura 1.1. Componentes básicos de los SFF

En la mayoría de los SFF, una pieza a ser procesada se fija inicialmente en un pallet en la estación de carga/descarga. A continuación, y por medio del sistema de manejo de materiales, ésta es trasladada a las colas de las estaciones de trabajo donde se realizarán las operaciones de procesado. Si la estación de trabajo requerida está vacía, el pallet se carga directamente en la misma. En caso contrario, el pallet se almacena en un dispositivo de almacenamiento interno. Cuando se termina el procesado, la pieza abandona la estación de trabajo quedando ésta disponible para la siguiente pieza que la requiera. Si la pieza sobre la que se ha terminado la operación de procesado requiere otras operaciones, el sistema de manejo de materiales traslada el pallet hasta la siguiente estación de trabajo. En caso de que una pieza no requiera más operaciones, ésta abandona el sistema a través de la estación de carga/descarga. El flujo de las piezas a través del sistema, realizado por el sistema de manejo de materiales, está dirigido y controlado por el sistema de control por computador. Además de estos componentes básicos puede haber otros equipos auxiliares tales como estaciones de inspección, equipos de lavado y secado. En Maleki (1991) puede encontrarse una completa descripción de la tecnología asociada a los SFF.

Básicamente, las estaciones de trabajo (véase esquema en la Figura 1.2) son máquinas herramientas controladas numéricamente, las cuales disponen de un almacén de herramientas y de un dispositivo de intercambio automático de las mismas. Las herramientas requeridas por todas las operaciones realizadas en una máquina dada son almacenadas en su almacén de herramientas el cual tiene capacidad limitada. El paso de una herramienta situada en el almacén a la máquina es realizado automáticamente en segundos por medio del dispositivo de intercambio de herramientas. La rapidez de estos intercambios permite a la máquina realizar operaciones consecutivas sin casi tiempos de puesta a punto entre operaciones, lo cual implica bajos costes de puesta a punto del SFF, el cual permite menos inventario en proceso que un sistema de producción tradicional debido a la posibilidad de trabajar en los SFF con tamaños de lotes tan pequeños como se quiera.

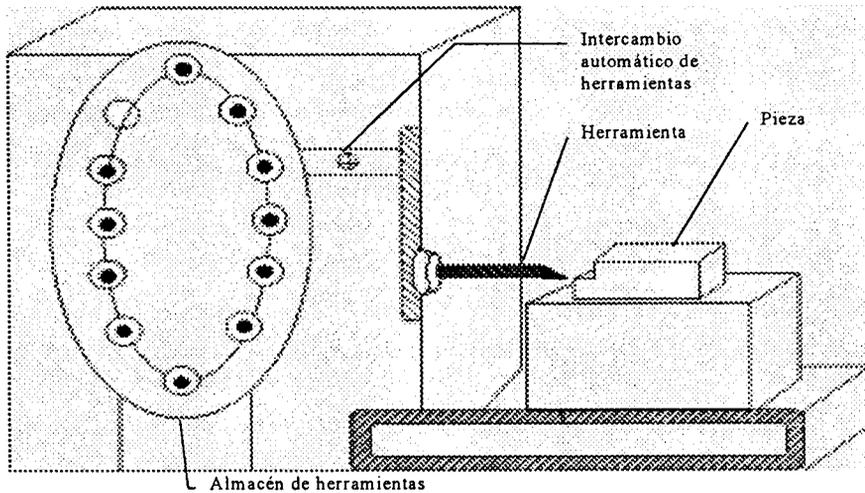


Figura 1.2. Estaciones de trabajo

La característica más importante en los SFF no es su alto grado de automoción (el cual está también presente en otros sistemas de producción) sino su flexibilidad, esto es, la capacidad de producir eficientemente una gran variedad de tipos de piezas en cantidades variables. En Browne et al (1984) puede encontrarse una exhaustiva división de la flexibilidad de los SFF en distintos tipos, así como los orígenes de los mismos.

Los SFF engloban características de las líneas "Transfer" y de los sistemas tipo "Taller". Las líneas "Transfer" están indicadas para volúmenes de producción altos, alta utilización de las máquinas y pequeños tiempos de proceso. De otro lado, los tradicionales sistemas tipo "Taller" se diseñan para producir pequeñas series de una variedad de productos. Esta flexibilidad se paga normalmente con tasas de utilización de máquinas bajas, altos tiempos de proceso y niveles de inventario elevados.

En la Figura 1.3 se presenta un esquema de diversos sistemas de producción clasificados atendiendo al volumen de producción y la flexibilidad de la que disponen. En un extremo están las líneas "Transfer" (altos volúmenes de producción de un producto). En el extremo opuesto se encuentran los sistemas tipo "Taller" (bajo volumen de producción de diversos productos). Los SFF se sitúan en la parte central al combinar la eficiencia de las líneas "Transfer" y la flexibilidad de los sistemas tipo "Taller".

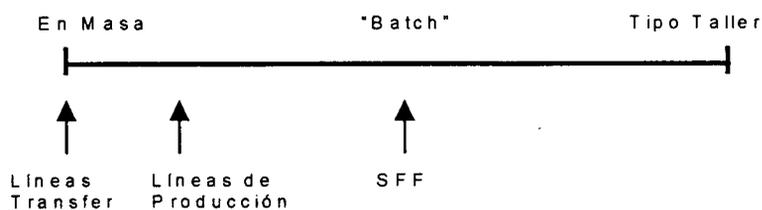


Figura 1.3. Clasificación según volumen de producción y flexibilidad

Aunque los SFF pueden ofrecer grandes beneficios como mayor utilización de las máquinas, menores costes unitarios, menores tiempos de proceso, mayor calidad y mayor rapidez de respuesta a los cambios del mercado, estas potenciales ventajas no son fáciles de obtener. De hecho, un SFF es un sistema complejo, compuesto por gran cantidad de

componentes de hardware y software interconectados entre sí, y con recursos muy limitados. Esta complejidad hace que el éxito en la implementación y gestión de tales sistemas sea difícil.

En la implementación de un SFF se presentan tanto problemas técnicos como organizativos. Dentro de estos últimos nos fijaremos en los relacionados con la gestión de la producción. Revisiones de las problemáticas y metodologías se pueden encontrar en Buzacott y Yao (1986), Kouvelis (1992), Gunasekaran et al (1993). Algunas razones que ilustran la complejidad de la gestión de la producción en los SFF frente a la de las líneas "Transfer" y los sistemas tipo "Taller" son:

- cada máquina es bastante versátil y capaz de realizar distintas operaciones.
- el SFF puede procesar varios ítems simultáneamente.
- cada ítem puede tener rutas alternativas.
- hay poca holgura en el sistema debido a la interrelación de los componentes y a las restricciones de operar en tiempo real.

En resumen, en la configuración de un SFF hay que enfrentarse por un lado con mayores potencialidades (que trae consigo un número mayor de variables de decisión) y de otro lado con restricciones adicionales. Esto crea la necesidad de desarrollar nuevos y apropiados procedimientos de planificación y control de la producción para extraer las potencialidades del sistema para altas tasas de producción.

1.2. DISEÑO, PLANIFICACIÓN Y GESTIÓN DE LOS SFF.

Hay varios artículos en la literatura discutiendo la naturaleza de los problemas de planificación en los SFF. Uno de los más completos es el de Van Looveren et al (1986) el cual identifica seis problemas y tres niveles de planificación (véase Figura 1.4):

- 1er Nivel: Planificación Estratégica.

Análisis de alternativas: No es más que una evaluación económica preliminar de las distintas alternativas (selección de equipos, ítems, etc.) para eliminar los diseños ineficientes.

Selección de alternativa: Identificación de la alternativa de menor coste, considerando factores técnicos y económicos.

- 2º Nivel: Planificación Táctica.

Selección de ítems: División de los órdenes de producción en lotes, de forma que sean procesadas a tiempo con los recursos disponibles (véase Figura 1.5).

Carga de máquinas: Asignación de operaciones y herramientas a máquinas dadas las órdenes a procesar (ver Figura 2).

- 3er Nivel: Planificación Operacional.

Entrada de piezas: Controlar el flujo de trabajo en el sistema, decidiendo qué ítems entran en el mismo, teniendo en cuenta la asignación de recursos a los ítems y el estado actual (máquinas, transportes, etc.) del sistema.

Selección de rutas: Elección de rutas para los ítems de las posibles alternativas.

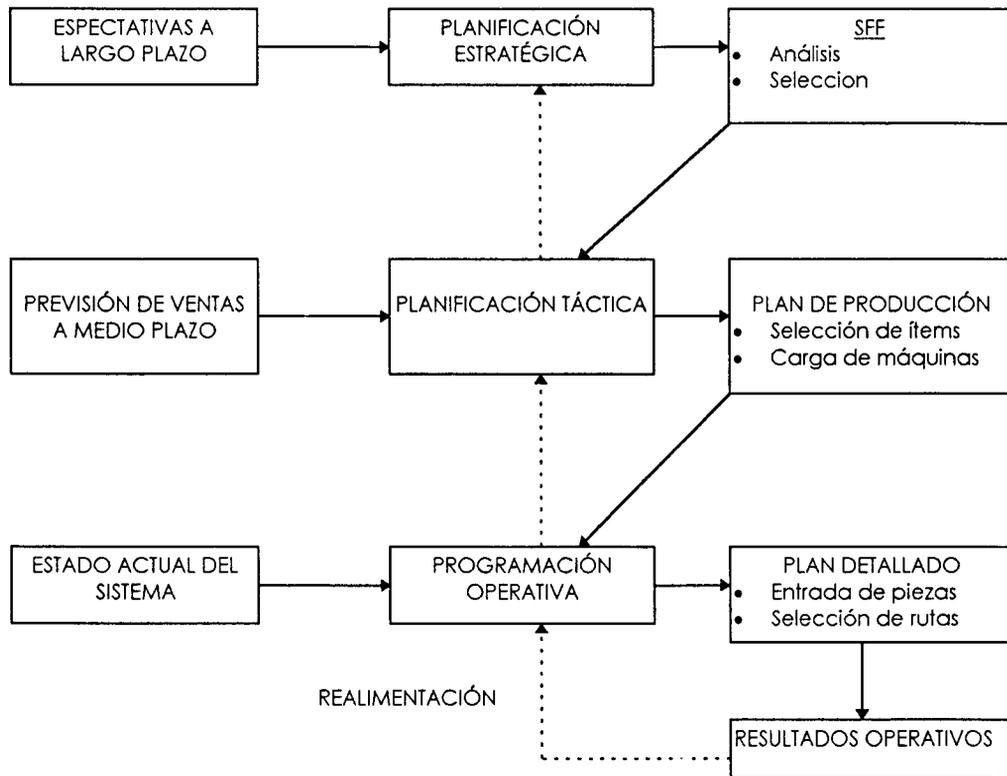


Figura 1.4. Estructura Jerárquica de Planificación de Van Looveren et al (1986)

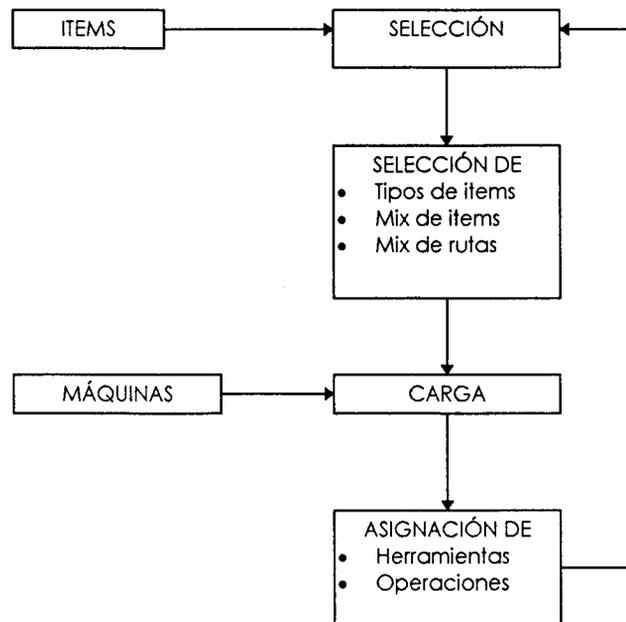


Figura 1.5. Selección de ítems a procesar y carga de máquinas

El nivel estratégico está relacionado con las decisiones a largo plazo, que influirán en la flexibilidad básica del SFF. En el táctico, las decisiones son las referentes a la planificación "off-line" del sistema de producción. En el nivel operacional las decisiones son las referidas

al funcionamiento del sistema en tiempo real.

Hay varias contribuciones en la literatura referidas a la planificación táctica de los SFF. Stecke (1983) da una jerarquización de cinco problemas que descomponen con más detalle la problemática de la planificación táctica de Van Looveren et al (1986):

- Selección de ítems: Determinación de un subconjunto (extraído del conjunto de ítems que tienen requerimientos de producción) para su inmediato y simultáneo procesamiento.
- Agrupación de máquinas: Particionar las máquinas en grupos de máquinas, de forma que cada máquina de un grupo dado sea capaz de realizar el mismo conjunto de operaciones.
- Ratios de producción: Determinación de los ratios relativos con los que serán producidos los ítems seleccionados.
- Asignación de recursos: Asignar el número limitado de recursos (pallets y fijaciones) entre los ítems seleccionados.
- Carga de máquinas: Asignar las operaciones y herramientas requeridas por los ítems seleccionados entre los grupos de máquinas, sujeto a las restricciones tecnológicas y de capacidad de los SFF.

Dentro de este entorno, se han propuesto distintos enfoques jerárquicos para la planificación y control de la producción (Jaikumar y Van Wassenhove, 1989; Sodhi et al, 1994-b y Chen y Chun, 1996). Básicamente, se usan dos formas en la planificación de la producción en los SFF: flexible y por lotes. En la flexible (Stecke y Kim, 1991) se opera de forma continua, haciendo los cambios necesarios en la configuración del sistema cuando sean necesarios. Por lotes (Hwang y Shogan, 1989), se seleccionan los ítems a ser procesados y se configura el sistema antes de que el procesado de los mismo comience.

En cuanto al nivel operacional, Stecke (1983) habla de problemas de secuenciación y de control. Los problemas de secuenciación se refieren al flujo de ítems a través del sistema, una vez que éste ha sido organizado (al nivel táctico). Los problemas de control están asociados con la monitorización del sistema y el seguimiento de la producción para asegurar requerimientos y plazos de entrega. Una revisión y clasificación de los problemas de secuenciación que surgen en los SFF puede encontrarse en Liu y MacCarthy (1996).

1.3. APROXIMACIONES EXISTENTES PARA EL PROBLEMA DE CARGA.

De acuerdo a su importancia en la planificación de la producción de los SFF, el problema de carga ha sido intensamente estudiado. En los distintos enfoques, éste ha sido abordado de forma aislada, así como conjuntamente con otros problemas relacionados como agrupación de máquinas, selección de ítems y secuenciación de operaciones. Por otro lado, algunos autores sólo consideran una de las dos problemáticas involucradas en el problema de carga: asignación de herramientas o asignación de operaciones. A continuación se analizan las aproximaciones existentes agrupadas según la problemática abordada.

1.3.1. MODELOS DE CARGA.

Chakravarty y Shtub (1984) abordan el problema de asignar herramientas y cargar las máquinas mediante un modelo de programación entera mixta con el objetivo de minimizar el tiempo total de procesado. Berrada y Stecke (1986) diseñan un algoritmo de ramificación y acotación para la resolución de un modelo de programación no lineal del problema de carga con el objetivo de equilibrar las cargas de trabajo de las máquinas.

Wilson (1992) diseña otro procedimiento de resolución para el mismo modelo. Lashkari et al (1987) presentan un modelo de programación entera no lineal con dos objetivos alternativos: minimizar la carga de los transportes o minimizar las actividades de fijación de las piezas. Sarin y Chen (1987) presentan un modelo de programación lineal entera para la asignación de operaciones y herramientas con el criterio de minimizar costes. Shanker y Rajamarthandan (1989) formulan la problemática mediante programación binaria no lineal con el objetivo de minimizar el número de movimientos de las piezas. Como método de resolución diseñan un algoritmo de ramificación y acotación. Ram et al (1990) modelan el problema de carga como un problema de flujo en redes generalizado de mínimo coste. A partir de la especial estructura de las restricciones diseñan un algoritmo de ramificación y acotación como procedimiento de resolución. Basnet (1996) diseña una heurística basada en técnicas de Búsqueda Tabú para la misma formulación.

De Werra y Widmer (1990) usan programación lineal entera con objetivos tales como minimizar el tiempo total de procesado y minimizar del número de intercambio de herramientas. Kouvelis y Lee (1991) modelan el problema de carga mediante programación lineal entera con el criterio de minimizar el coste total de operación. Kim y Yano (1993) analizan el problema como uno de secuenciación con restricciones adicionales con la minimización del tiempo de finalización del procesado como criterio. Se diseñan procedimientos heurísticos que usan la asignación de cargas de trabajo como un objetivo indirecto de maximizar la tasa de producción. Kim y Yano (1994) diseñan un procedimiento distinto de resolución basado en técnicas de ramificación y acotación. Kirkavak and Dinçer (1993) proponen un modelo de programación lineal entera mixta además de diversas reglas heurísticas para el problema de carga con el criterio de equilibrar cargas de trabajo.

1.3.2. MODELOS DE CARGA Y AGRUPACIÓN DE MÁQUINAS.

Stecke (1983, 1986) formula los problemas de carga y agrupación de máquinas usando programación entera no lineal con objetivos como equilibrar cargas y minimizar los movimientos de piezas. Stecke y Raman (1994) primero solucionan el problema de agrupamiento y luego la carga de las máquinas con distintos objetivos de carga.

1.3.3. MODELOS DE CARGA Y SELECCIÓN DE ÍTEMS.

Bastos (1988) formula mediante programación lineal entera mixta la problemática de selección de ítems con el objetivo de maximizar la suma de las prioridades de los ítems seleccionados. Para el problema de carga propone un procedimiento iterativo basado en programas lineales con el doble objetivo de minimizar por un lado la carga de trabajo de la máquina "cuello de botella", y por otro, el desequilibrio de de cargas de trabajo del resto de las máquinas. Hwan y Shogan (1989) presentan una formulación lineal entera para la selección de ítems y carga de máquinas con el objetivo de maximizar la suma de prioridades. El modelo es resuelto como un problema de flujo en redes con restricciones. Shanker y Srinivasulu (1989) formulan la problemática mediante programación lineal entera con el objetivo de maximizar la carga total de trabajo. Además, diseñan tres procedimientos heurísticos con el doble objetivo de minimizar el desequilibrio de cargas y maximizar la tasa de producción de los recursos críticos.

Stecke y Kim (1991) presentan un modelo de programación lineal para simultáneamente seleccionar ítems y determinar los ratios de producción, con el criterio de equilibrar cargas. Liang y Dutta (1993) usan programación lineal entera mixta para simultáneamente abordar la selección de ítems y carga de máquinas con el objetivo inicial de maximizar la suma de prioridades y el posterior de minimizar costes o tiempo de finalización del procesado. Moreno y Ding (1993) presentan un modelo de programación lineal entera mixta y dos

heurísticas: una para equilibrar cargas y la otra inicialmente para minimizar el número de trabajos retrasados, para a continuación equilibrar cargas. Srivastava y Chen (1993) modelan mediante programación lineal entera los problemas de selección de ítems y asignación a las máquinas de las herramientas requeridas con el criterio de maximizar la suma de prioridades de los ítems seleccionados. Los autores diseñan una heurística basada en técnicas de Búsqueda Tabú para resolver el modelo propuesto. Mohamed (1996) presenta un modelo de programación lineal entera para la selección de ítems y carga de máquinas en un horizonte discretizado en períodos. De nuevo el objetivo está basado en la suma de prioridades. La salida del modelo es enlazada con un algoritmo para el diseño de células y familias el cual equilibra la carga de trabajo de las células.

1.3.4. MODELOS DE CARGA Y SECUENCIACIÓN.

Shanker y Tzen (1985) formulan los problemas de carga y secuenciación mediante programación lineal entera mixta con el objetivo de minimizar, o un desequilibrio de las cargas de trabajo, o una mezcla del desequilibrio de las cargas de trabajo y el retraso de los ítems. Green y Sadowski (1986) presentan una formulación de programación lineal entera mixta para simultáneamente abordar los problemas de carga y secuenciación con diferentes objetivos interrelacionados entre sí (minimizar tiempo de finalización del procesado, tiempo de flujo medio y retraso medio). Sawik (1990) presenta tres modelos distintos para los problemas de selección de ítems, carga de máquinas y secuenciación de operaciones. Los objetivos usados son respectivamente: minimizar el número de intercambios de herramientas, minimizar el desequilibrio de cargas y minimizar el máximo retraso de un ítem. Sherali et al (1990) presenta un procedimiento exacto y otro heurístico para la resolución de un modelo de programación lineal entera mixta que aborda simultáneamente la selección de ítems, asignación y secuenciación de operaciones. El objetivo usado es el de minimizar el tiempo de finalización del procesado. Chen y Chun (1996) presentan una formulación de programación lineal entera mixta para a la vez seleccionar ítems, asignar herramientas y operaciones, con el objetivo de minimizar el tiempo de finalización del procesado. Además, la secuenciación de las piezas la obtienen mediante diversas reglas.

1.3.5. MODELOS DE ASIGNACIÓN DE HERRAMIENTAS Y OPERACIONES.

Arbib et al (1990) solucionan el problema de asignación de operaciones suponiendo que las herramientas se han asignado previamente. Proponen dos objetivos, uno principal y otro secundario. El principal es el del equilibrado de cargas de trabajo. El secundario, minimizar el número de movimientos de las piezas entre estaciones de trabajo. Liang y Dutta (1990) presentan un modelo de programación lineal entera mixta para la asignación de operaciones con el objetivo de minimizar los costes totales, incluyendo en éstos un coste de penalización por el excesivo tiempo de finalización del procesado. Chen y Chun (1991) primero solucionan conjuntamente la asignación de herramientas y operaciones con el objetivo de maximizar el número de operaciones asignadas. A continuación resuelven el problema de asignación de operaciones con el criterio de minimizar el tiempo de finalización del procesado. Sodhi et al (1994b) proponen un modelo de programación lineal para la selección de ítems, uno de programación lineal entera mixta para la carga de máquinas y otro de programación lineal para la asignación de operaciones. En todas las formulaciones usan como objetivo la minimización de costes. D'Alfonso and Ventura (1995) presentan un modelo de programación lineal entera y una heurística Lagrangiana para la asignación de herramientas con el objetivo de minimizar los movimientos de las piezas entre máquinas. Sodhi et al (1994a) inicialmente solucionan la asignación de herramientas de forma que se puedan producir todos los tipos de ítems, y posteriormente asignan la carga de trabajo de las piezas de forma lo más equilibrada posible. Atan y

Pandit (1996) inicialmente abordan las asignación de operaciones, suponiendo que no hay restricciones de herramientas, para a continuación usar un modelo de programación lineal entera para asignar las herramientas requeridas. Usan como objetivo la minimización del número total de herramientas en el sistema como una forma indirecta de minimizar el coste de las herramientas y maximizar su utilización.

1.3.6. RESUMEN.

En la Tabla 1.1 se muestra un resumen de las distintas aproximaciones, las cuales se clasifican según la problemática abordada y la función objetivo utilizada. Téngase en cuenta que debido a que algunos autores usan más de una función objetivo, la misma referencia puede aparecer en distintas posiciones. En la Tabla 1.2 se muestran las mismas referencias, clasificadas de acuerdo a la herramienta de modelada usada. Una misma referencia puede aparecer en más de una misma columna si por ejemplo se proponen un modelo de programación lineal entera mixta y un procedimiento heurístico para su resolución.

A partir de la Tabla 1.1 puede comentarse que:

- La función objetivo más usada es la de equilibrar las cargas de las máquinas seguida por la más tradicional de minimizar costes.
- Las problemáticas más abordadas han sido, por este orden, las de carga, carga con selección de ítems y carga con secuenciación. Para cada una de ellas los criterios usados son respectivamente los de equilibrar cargas, maximizar la suma de prioridades de los ítems seleccionados y esquemas de secuenciación tradicionales como minimización del tiempo de finalización del procesado y minimización del retraso.

Respecto a la herramienta más usada podemos decir que ha sido la programación lineal entera/mixta. Téngase en cuenta que las variables naturales del problema incluyen variables enteras asociadas con la selección de ítems, asignación de herramientas y secuenciación. Teniendo en cuenta la alta complejidad computacional de este tipo de modelos, se han diseñado distintos procedimientos para encontrar buenas soluciones de una forma rápida.

En cuanto a las hipótesis empleadas por los distintos autores para la formulación de sus modelos, dos de ellas merecen algunos comentarios:

- Cada tipo de pieza se procesa por una sólo ruta de procesado.
Esta hipótesis es impuesta, o bien asignando cada operación a una sólo máquina, o bien asignando cada tipo de herramienta a una sólo máquina. La consecuencia de esta hipótesis está en que una de las características de la flexibilidad de los SFF, la flexibilidad de rutas, sólo se usa parcialmente.
- Asignación de una copia de cada tipo de herramienta a las máquinas donde son requeridas.

No obstante, la asignación de múltiples copias de las herramientas puede ser beneficiosa, y necesaria en algunos casos como cuando las herramientas se requieren fuertemente o éstas tienen una vida útil limitada. Por lo tanto, puede ser conveniente permitir que en la asignación de herramientas a máquinas se permita la asignación de más de una copia de cada uno de los tipos de herramientas. De esta forma, se incrementará tanto el horizonte de tiempo hasta que el sistema sea parado para una nueva configuración, como el tiempo de procesado total que puede ser compartido entre las máquinas.

Tabla 1.1. Clasificación de las referencias según problemática abordada y criterio usado

Función Objetivo	Carga	Agrupamiento y carga	Selección y carga	Carga y secuenciación	Asignación de herramientas y determinación de rutas
Equilibrado de cargas	Berrada & Steckle (1986) Wilson (1992) Kim & Yano (1993) Kirkavak & Dinçer (1993) Kim & Yano (1994) Sarin & Chen (1987) Ram et al (1990) Kouvelis & Lee (1991) Basnel (1996)	Stecke (1983) Stecke (1986) Stecke & Raman (1994)	Bastos (1988) Shanker & Srinivasulu (1989) Stecke & Kim (1991) Moreno & Ding (1993) Solomon et al (1995)	Shanker & Tzen (1985) Sawik (1990)	Arbib et al (1990) Sodhi et al (1994a)
Optimización de costes			Liang & Dutta (1993)		Liang & Dutta (1990) Sodhi et al (1994b)
Minimización de movimientos de piezas	Shanker & Rajamarthandan (1989) Wilson (1989)	Stecke (1983) Stecke (1986)			Arbib et al (1990) D'Alfonso & Ventura (1995)
Maximización de la suma de prioridades de ítems seleccionados			Bastos (1988) Hwan & Shogan (1989) Liang & Dutta (1993) Srivastava & Chen (1993) Mohamed (1996)		
Minimización de intercambios de herramientas	De Weira & Widmer (1990)			Sawik (1990)	
Minimización del tiempo máximo de producción			Chen & Chung (1996) Liang & Dutta (1993)	Greene & Sadowski (1986) Sherali et al (1990) Chen & Chung (1996)	Chen & Chung (1991) Liang & Dutta (1990)
Minimización del tiempo de finalización	Chakravarty & Shitub (1984) De Weira & Widmer (1990)				
Minimización del retraso de los ítems			Moreno & Ding (1993)	Shanker & Tzen (1985) Greene & Sadowski (1986) Sawik (1990)	
Otros	Lashkari et al (1987) De Weira & Widmer (1990)	Stecke (1983) Stecke (1986)	Shanker & Srinivasulu (1989)	Greene & Sadowski (1986)	Chen & Chung (1991) Atan & Pandit (1996)

Tabla 1.2. Clasificación de referencias según las características del modelo

Programación No Lineal	Programación Lineal Entera/Mixta	Programación Lineal	Modelo de Flujo en Redes	Procedimiento Heurístico
Stecke (1983)	Chakravarty & Shtub (1984)	Bastos (1988)	Ram et al (1990)	Shanker & Tzen (1985)
Stecke (1986)	Shanker & Tzen (1985)	Stecke & Kim (1991)	Hwan & Shogan (1989)	Shanker & Srinivasulu (1989)
Berrada & Stecke (1986)	Greene & Sadowski (1986)	Stecke & Raman (1994)		Sherali et al (1990)
Lashkari et al (1987)	Sari & Chen (1987)	Sodhi et al (1994b)		Wilson (1992)
Shanker & Rajamarthandan (1989)	Bastos (1988)			Moreno & Ding (1993)
Wilson (1992)	Shanker & Srinivasulu (1989)			Kim & Yano (1993)
	Wilson (1989)			Kirkavak & Dinçer (1993)
	Sawik (1990)			Stecke & Raman (1994)
	Liang & Dutta (1990)			Atan & Pandit (1996)
	De Werra & Widmer (1990)			Chen & Chung (1996)
	Arbib et al (1990)			
	Sherali et al (1990)			
	Kouvelis & Lee (1991)			
	Chen & Chung (1991)			
	Srivastava & Chen (1993)			
	Kirkavak & Dinçer (1993)			
	Liang & Dutta (1993)			
	Moreno & Ding (1993)			
	Kim & Yano (1994)			
	Sodhi et al (1994a)			
	Sodhi et al (1994b)			
	D'Alfonso & Ventura (1995)			
	Solomon et al (1995)			
	Chen & Chung (1996)			
	Basnet (1996)			
	Atan & Pandit (1996)			
	Mohamed (1996)			

1.4. MOTIVACIÓN Y PLANTEAMIENTO DE LA INVESTIGACIÓN.

Esta tesis gira en torno a las problemáticas asociadas a la carga de las máquinas en los SFF. La importancia de dichas problemáticas así como las hipótesis empleadas en formulaciones existentes determinan la necesidad de su estudio. El paso inicial es la reformulación del problema de carga teniendo en cuenta las limitaciones de los modelos existentes. Para ello, en el capítulo 2 se presenta un nuevo modelo de programación lineal entera mixta que engloba las decisiones referidas a la asignación de operaciones y herramientas a las máquinas. Además, el modelo se extiende para englobar las decisiones referidas a la selección de ítems. Ambos modelos están basados en la determinación del mix agregado de rutas para los ítems, en lugar de la asignación individual de operaciones.

En el capítulo 3 se enmarcan los modelos propuestos para la planificación de la producción en un contexto más amplio. Así se diseña un sistema ABC de estimación de costes de operación en los SFF, donde la característica principal es su enlace con la fase de planificación de la producción. Los costes indirectos son imputados a los distintos ítems en base al consumo directo de recursos de producción.

En el capítulo 4 se presentan dos procedimientos heurísticos diseñados para abordar la resolución eficiente del modelo propuesto para el problema de carga. Ambos procedimientos están basados en relajaciones continuas del problema original. Los procedimientos se evalúan, sobre dos conjuntos de problemas generados aleatoriamente de dificultad creciente, tanto en lo referido a la calidad de las soluciones ofrecidas, así como a sus requerimientos computacionales. Los resultados muestran la existencia de algunos problemas de elevada dificultad para los que los procedimientos no son capaces de ofrecer solución admisible. Para abordar la resolución de estos problemas de alta dificultad, en el capítulo 6 se diseñan los elementos básicos de estrategias de búsqueda local. Se utiliza una codificación entera de soluciones y una evaluación de las mismas basada en una estrategia de oscilación en torno a la frontera de admisibilidad. Como soluciones iniciales se utilizan, además de las ofrecidas por las heurísticas diseñadas en el capítulo 4, otras obtenidas a partir de la relajación continua del problema original. La integración de estos elementos en estrategias mixtas de Recocido Simulado y Búsqueda Tabú permite abordar la resolución de los problemas de elevada dificultad en unos tiempos razonables.

En el capítulo 6 se presenta una nueva metodología para los problemas de diseño y carga de células de fabricación flexible. La metodología divide la problemática en dos etapas. En la primera, se define la configuración de las distintas células. En la segunda, una vez diseñadas las células, se cargan las mismas. En ambas etapas se tiene en cuenta la existencia de rutas alternativas de procesado para los distintos ítems. La fase de diseño es abordada mediante dos enfoques. El primero de ellos está basado en la consideración de cada una de las rutas de procesado (planes de proceso) de un ítem como partes (virtuales) independientes. Para el problema de clusterización planteado se propone una Red Neuronal Auto-organizativa la cual es comparada con otros algoritmos usados para el mismo problema. El segundo enfoque para la fase de diseño de células está basado en la transformación de los planes de proceso alternativos de cada una de los ítems en un único plan de proceso difuso. Se diseñan dos procedimientos de clusterización basados en el algoritmo de las C-medias. Para la fase de carga de las células flexible se propone un modelo de programación lineal continua basado en el presentado en el capítulo 2 para la carga de los SFF. Finalmente, en el capítulo 7 se hacen las consideraciones finales señalándose varios caminos por donde avanzar la investigación, mostrándose en el capítulo 8 la bibliografía directamente relacionada tanto con las problemáticas en estudio como con los métodos de resolución diseñados.

2.1 INTRODUCCIÓN.

En este capítulo se presentan dos nuevas formulaciones para los problemas de carga y selección de ítems en los SFF. Ambas formulaciones se plantean dentro de una estructura de planificación por lotes (Hwang y Shogan, 1989), esto es, una vez que comience el procesado de las piezas de un lote, la configuración del SFF no podrá ser modificada hasta que termine el procesado de todas las piezas del lote. Una vez presentadas las formulaciones matemáticas de los dos problemas, éstas son ilustradas a partir de los datos de un SFF.

2.2 FORMULACIÓN DEL PROBLEMA DE CARGA.

En este apartado se establece la formulación matemática del problema de carga, para lo cual previamente se introducen las suposiciones que lo caracterizan y la notación empleada.

2.2.1 SUPOSICIONES.

Consideremos un SFF compuesto por un grupo de estaciones. Una estación está compuesta de máquinas funcionalmente idénticas, es decir, similares físicamente y cargadas con las mismas herramientas, por lo que son capaces de realizar el mismo conjunto de operaciones en idénticos tiempos de procesado. Los ítems a ser procesados simultáneamente, así como sus cantidades, son conocidas y dados.

El elemento más importante usado en el modelado es el de ruta de procesado. Un ítem puede ser procesado de diferentes formas, cada una de las cuales constituye una ruta de procesado. Cada ruta de procesado puede ser descrita como un conjunto de operaciones cada una de las cuales requiere una estación, un tipo de herramienta y un tiempo de procesado. Para cada tipo de herramienta se conoce su vida útil, el número de copias disponibles y las posiciones que ocupa cada tipo de herramienta en los almacenes de herramientas de las máquinas de las estaciones.

El problema de carga en este contexto no es más que la determinación del mix de rutas para cada ítem, así como la asignación de herramientas a máquinas. El mix de rutas de un ítem vendrá determinado por la proporción del mismo que es procesada por cada una de sus rutas alternativas. La asignación de herramientas a máquinas no es más que determinar cuántas copias de cada tipo de herramienta se asignan a las máquinas de las estaciones.

En cuanto a las restricciones del problema de carga las podemos clasificar en dos categorías: de capacidad y tecnológicas. La primera categoría incluye las correspondientes a las capacidades limitadas de los almacenes de herramientas de las máquinas, número limitado de copias de cada tipo de herramienta y limitación del tiempo de procesado disponible en el sistema teniendo en cuenta las herramientas asignadas y sus vidas útiles. En la segunda categoría se incluyen las correspondientes a los datos de producción, esto es, ítems, rutas, estaciones, máquinas y tipos de herramientas. El criterio usado será el de equilibrar la carga de las máquinas.

Entre las suposiciones se han tenido en cuenta alguna de las cuestiones comentadas en el apartado 1.3.6 referentes a otras aproximaciones existentes para el mismo problema:

- Como objetivo se usa el de equilibrar las cargas de trabajo en las máquinas. Además de ser el más usado (véase Tabla 1.1) existen resultados (Stecke y Solberg, 1985) que muestran que este criterio es una forma indirecta de maximizar la tasa de producción y además, da lugar a formulaciones lineales.

- Para cada ítem se permite la posibilidad de usar rutas alternativas de procesado. De esta forma pretendemos hacer uso en la fase de carga de las máquinas de la flexibilidad de rutas característica de los SFF.
- En cuanto a la asignación de herramientas a máquinas, se permite usar más de una copia de cada tipo de herramienta en las máquinas.

2.2.2 NOTACIÓN.

En la Tabla 2.1 se presenta la notación usada para la formulación del problema de carga.

Tabla 2.1. Notación usada en los modelos

Subíndices y datos de entrada:	
i	ítems
r	rutas
j	estaciones
k	herramientas
M	número de estaciones
D_i	demanda de ítem i
m_j	número de máquinas en estación j
C_j	capacidad del almacén de herramientas de cada máquina de la estación j
A_k	número de copias disponibles de herramienta tipo k
L_k	vida útil de una copia de herramienta tipo k
S_{jk}	posiciones ocupadas por una copia de herramienta tipo k en máquinas de estación j
p_{irjk}	tiempo de procesado de ítem i según su ruta r en estación j y usando la herramienta tipo k
$J(k) = \left\{ j: \sum_i \sum_r p_{irjk} > 0 \right\}$ conjunto de estaciones que usa la herramienta tipo k	
$K(j) = \left\{ k: \sum_i \sum_r p_{irjk} > 0 \right\}$ conjunto de herramientas usadas en estación j	
Variables de decisión:	
θ_{ir} (continuas)	proporción de unidades de ítem i asignadas a su ruta r
u_j (continuas)	carga de trabajo en cada una de las máquinas de la estación j
u_{\min}, u_{\max} (continuas)	cargas de trabajo máxima y mínima
ξ_{jk} (enteras)	número de copias de herramienta tipo k asignadas a cada una de las máquinas de la estación j

2.2.3 FUNCIÓN OBJETIVO Y RESTRICCIONES.

Una vez optado por el objetivo de equilibrar las cargas de trabajo de las máquinas se plantea el problema de cómo modelarlo debido a las distintas propuestas aparecidas en la literatura. Así por ejemplo, Stecké (1983) y Berrada y Stecké (1986) minimizan la máxima

carga de trabajo de las máquinas mientras que Kirkavak y Dinçer (1993) maximizan la menor carga de trabajo de las máquinas y Stecke (1983) minimiza la diferencia entre la mayor y menor carga de trabajo. En definitiva, se ha decidido que se van a equilibrar las cargas de trabajo de las máquinas pero se han encontrado distintas formulaciones de un mismo objetivo surgiendo la cuestión de si son equivalentes. Gaudioso y Legato (1991) muestran que en problemas donde la carga total es constante, las tres formulaciones anteriores del equilibrado son equivalentes. No obstante, este no es el caso de la carga de máquinas en los SFF. Dado un ítem, al ser los tiempos de procesado distintos entre rutas, la carga total de trabajo es función del mix de rutas de los ítems. Como consecuencia, las anteriores formulaciones dan lugar a tres medidas del desequilibrio distintas.

Además de las anteriores, se han usado otras medidas del desequilibrio tales como una ponderación de desequilibrios sobre una carga de referencia (Shanker y Tzen, 1985) o expresiones no lineales (Stecke, 1983).

Después de modelar y evaluar ocho formas distintas de equilibrar cargas (véase Anexo I), se ha concluido que la forma más robusta es la que minimiza la diferencia entre la mayor y menor carga de trabajo.

El modelo propuesto para el problema de carga es:

$$\text{minimizar } u_{\max} - u_{\min} \quad (2.1)$$

sujeto a:

$$\sum_r \theta_{ir} = 1 \quad \forall i \quad (2.2)$$

$$\sum_i \sum_r D_i p_{irjk} \theta_{ir} \leq m_j L_k \xi_{jk} \quad \forall k, J(k) \quad (2.3)$$

$$\sum_{j \in J(k)} m_j \xi_{jk} \leq A_k \quad \forall k \quad (2.4)$$

$$\sum_{k \in K(j)} S_{jk} \xi_{jk} \leq C_j \quad \forall j \quad (2.5)$$

$$\sum_i \sum_r \sum_k D_i p_{irjk} \theta_{ir} = m_j u_j \quad \forall j \quad (2.6)$$

$$u_{\min} \leq u_j \leq u_{\max} \quad \forall j \quad (2.7)$$

$$\xi_{jk} \geq 0 \text{ enteras} \quad \forall k, J(k) \quad (2.8)$$

$$\theta_{ir} \geq 0 \quad \forall i, r; u_j \geq 0 \quad \forall j; u_{\max} \geq 0; u_{\min} \geq 0 \quad (2.9)$$

Las restricciones (2.2) imponen para cada ítem que la suma de las proporciones asignadas a cada una de sus rutas debe ser la unidad, esto es, que se procesen todas las piezas del

mismo. En (2.3) se acota el número de herramientas asignadas a máquinas según los requerimientos de producción. Las restricciones (2.4) y (2.5) son la limitación en el número de copias y la capacidad de los almacenes de herramientas de las máquinas respectivamente. En (2.6) se computa la carga de las máquinas de las distintas estaciones. En (2.7) se define u_{\max} y u_{\min} como la máxima y mínima carga de trabajo de las máquinas respectivamente. La integridad de las variables ξ_{jk} se impone en (2.8).

El modelo propuesto es diferente de otros de la literatura en los siguientes aspectos:

- Considera rutas de procesamiento enteras, en lugar de operaciones individuales.
- Referido a las herramientas, considera no sólo la capacidad limitada de los almacenes de herramientas las máquinas, sino que también tiene en cuenta que: las copias disponibles lo están en un número limitado, las vidas útiles de las herramientas son finitas y la asignación de múltiples copias de las herramientas a las máquinas es posible.
- El objetivo de equilibrar las cargas de trabajo de las máquinas ha sido modelado tras un análisis de objetivos aparentemente equivalentes.

Como fue comentado en el apartado 1.3.6, la programación lineal entera mixta es la forma natural de abordar el problema debido al carácter de las decisiones a tomar. El número de variables enteras depende del grado de "comunalidad" entre operaciones y en ningún caso superará el número de estaciones por el número de herramientas. Las restricciones (2.4) y (2.5) son múltiples restricciones del tipo de la mochila. Como consecuencia, el problema puede no ser solucionable de una forma fácil en el peor de los casos. En el capítulo 4 se diseñan diversos procedimientos heurísticos para la resolución del modelo propuesto del problema de carga.

2.3 FORMULACIÓN DEL PROBLEMA DE SELECCIÓN DE ÍTEMS.

El modelo anterior para el problema de carga puede ser extendido para incluir las decisiones referentes a la selección de ítems. Para ello, además de la notación usada para el problema de carga (apartado 2.2.2), se introducen variables binarias (Y_i) que recogerán si un ítem es seleccionado o no para ser procesado en el próximo período.

Como se comentó en el apartado 1.3.6, los dos objetivos más comunes en los problemas de selección de ítems y carga de máquinas son el equilibrado de cargas y la maximización de prioridades de los ítems seleccionados, por lo que, en la medida de lo posible, sería interesante tenerlos en cuenta ambos. Para conseguirlo, se utiliza la carga media (u_{med}) de las estaciones y se incluyen limitaciones en los desequilibrios de las cargas de trabajo en (2.13). Además se ha considerado una prioridad (w_i) para cada ítem. En la función objetivo (2.10) se maximiza la suma de prioridades de los ítems seleccionados. El modelo completo es:

$$\text{maximizar } \sum_i w_i Y_i \quad (2.10)$$

sujeto a:

$$\sum_r \theta_{ir} = Y_i \quad \forall i \quad (2.11)$$

$$u_{med} - \frac{1}{M} \sum_j u_j = 0 \quad (2.12)$$

$$(1 - \alpha) u_{med} \leq u_j \leq (1 + \alpha) u_{med} \quad \forall j \quad (2.13)$$

(2.3), (2.4), (2.5), (2.6), y (2.8)

$$Y_i = 0,1 \quad \forall i \quad (2.14)$$

$$\theta_{ir} \geq 0 \quad \forall i, r; u_j \geq 0 \quad \forall j; u_{med} \geq 0 \quad (2.15)$$

Las restricciones (2.11) son análogas a (2.2) e imponen que, para cada ítem, la suma de proporciones asignadas a cada una de sus rutas debe ser uno o cero dependiendo de si el ítem es seleccionado o no. En (2.12) se computa la carga media de trabajo. Las restricciones (2.13) acotan las cargas de trabajo sobre la carga media usando un parámetro α que define los desequilibrios máximos y mínimos. En (2.14) se impone la integridad de las variables Y_i . El resto de las restricciones permanecen igual que en la formulación del problema de carga.

Obsérvese que este modelo busca el equilibrado de las cargas de una forma flexible. Haciendo α igual a cero se obtiene un equilibrado perfecto. No obstante, $\alpha = 0$ puede dar lugar a peores soluciones en cuanto a función objetivo e incluso a inadmisibilidades. Como consecuencia, sería recomendable usar valores pequeños de α distintos de cero.

Los coeficientes de prioridad (w_i) indican cómo de urgente es un ítem. El modelo intentará seleccionar los ítems más urgentes, quizás a costa de algo de deterioro en el desequilibrio. Los ítems no seleccionados para el actual plan de producción deberían ser considerados para el próximo con un incremento en sus prioridades de tal forma que aumente su posibilidad de entrar en el plan de producción. En esta línea, la función objetivo se puede ver como una forma indirecta de minimizar el número de trabajos retrasados.

2.4 ILUSTRACIÓN.

En este apartado se ilustra la funcionalidad de las formulaciones anteriormente propuestas a partir de los datos de un ejemplo. Inicialmente, se plantea una situación en la que, dados los requerimientos de producción sobre un conjunto de ítems, se cargan las máquinas. A continuación, los requerimientos de producción son ampliados a más ítems y, debido a que es imposible su procesamiento simultáneo, se aborda conjuntamente las decisiones de carga con las de selección de ítems.

2.4.1 DATOS DE ENTRADA.

Consideramos un SFF con 6 máquinas formando cuatro estaciones como se muestra en la Tabla 2.2 En dicha tabla también se muestra las capacidades de los almacenes de herramientas de las máquinas de las estaciones.

Tabla 2.2. Datos de las estaciones

Estaciones	No. de máquinas	Capacidad del almacén de herramientas
1	1	30
2	1	35
3	2	30
4	2	30

Hay cuatro tipos de ítems a procesar en las cantidades mostradas en la Tabla 2.3. Además, se muestra el número de rutas alternativas para cada ítem.

Tabla 2.3. Datos de los ítems

Ítem	Demanda	No. de rutas
1	25	3
2	30	3
3	32	2
4	45	2

Se consideran 20 tipos de herramientas, presentándose en la Tabla 2.4 la vida útil o duración de cada copia (en minutos) y el número de copias disponibles.

Tabla 2.4. Datos de las herramientas

Tipo	1	2	3	4	5	6	7	8	9	10
No. de copias	5	6	5	7	5	5	6	7	5	7
Vida útil	900	1200	1100	1200	900	900	1200	900	700	900

Tipo	11	12	13	14	15	16	17	18	19	20
No. de copias	6	5	5	6	7	5	6	5	4	5
Vida útil	1100	900	1100	950	900	1200	900	1100	1200	1000

Los tiempos de procesado han sido generados en el rango de 0 a 48 minutos según la siguiente fórmula pseudoaleatoria:

$$p_{irk} = [(9i + 7r + 11k) \bmod 7] [(5i + 7r + 3j) \bmod 5] S_{jk} \tag{2.16}$$

donde S_{jk} es el número de posiciones que ocupan las herramientas en los almacenes de las máquinas de las estaciones y viene dada por:

$$S_{jk} = (17j + 7k) \text{ mod } 3 \tag{2.17}$$

Cuando $S_{jk} = 0$ se considera que la herramienta k no puede ser usada en la estación j por ningún ítem a lo largo de ninguna de las rutas, esto es, $p_{irk} = 0 \forall i,r$.

2.4.2 CARGA DE MÁQUINAS.

Construyendo el modelo de carga (2.1)-(2.9) para los datos de carga presentdos en el apartado anterior, queda un modelo de programación lineal entera mixta compuesto de de 96 restricciones, 10 variables continuas y 56 variables enteras. El modelo ha sido resuelto en 44 segundos usando el optimizador XA Callable Library (1994) en un PC-Pentium a 100 Mhz bajo el sistema operativo Windows 95.

En la Tabla 2.5 se presenta para cada ítem su mix de rutas, esto es, la proporción de unidades del mismo que es procesado según cada una de sus rutas. En la Tabla 2.6 se muestran, además de los resultados referidos a asignación de herramientas a máquinas, las posiciones ocupadas en los almacenes de herramientas de las distintas máquinas de las estaciones y la carga de las máquinas.

Tabla 2.5. Rutas de los ítems

Ítem (i)	Mix de rutas		
	θ_{i1}	θ_{i2}	θ_{i3}
1	0	1	0
2	0.385	0.615	0
3	1	0	-
4	0.113	0.887	-

Tabla 2.6. Asignación de herramientas y carga de máquinas

j	Asignación de herramientas (ξ_{jk})																				Posic ocup	u_j
	Herramienta (k)																					
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20		
1	-	1	2	-	1	2	-	1	2	-	1	1	-	1	2	-	1	2	-	1	29	10164
2	3	-	1	1	-	1	2	-	1	2	-	1	1	-	1	1	-	1	1	-	28	7860
3	1	1	-	1	1	-	1	2	-	1	1	-	1	1	1	1	-	1	1	-	23	9506
4	-	1	1	-	1	1	1	1	1	-	1	1	-	1	1	1	1	-	1	-	19	7907

Algunos comentarios pueden ser hechos sobre la solución óptima: (1) no todos los ítems necesitan usar sus rutas alternativas, (2) hay poca holgura en las restricciones relacionadas con las capacidades de los almacenes de herramientas y (3) en cuanto a las restricciones que limitan el número de copias, muchas de ellas se verifican con signo de igualdad.

2.4.3 SELECCIÓN DE ÍTEMS.

Supóngase que los requerimientos de producción en lugar de ser los de la Tabla 2.3, vienen dados por la Tabla 2.7. Para cada ítem se muestra la demanda, el número de rutas alternativas y la prioridad. Obsérvese que además de los 4 primeros ítems (los de la Tabla 2.3), que son los que presentan mayor prioridad, aparecen 6 nuevos ítems con requerimientos de producción. Los tiempos de procesado de los nuevos ítems se generan usando (2.16).

Tabla 2.7. Ítems con requerimientos de producción

Ítem	1	2	3	4	5	6	7	8	9	10
Demanda	25	30	32	45	25	20	45	40	20	28
No. de rutas	3	3	2	2	2	2	3	2	3	2
Prioridad	0.9	0.8	0.8	0.7	0.6	0.6	0.6	0.5	0.5	0.4

En esta situación, los requerimientos de producción de los ítems a procesar exceden la capacidad de producción en el corto plazo (fijada por los recursos disponibles en cuanto a máquinas, herramientas, etc.). Como consecuencia, es imposible procesar simultáneamente todos los ítems con requerimientos de producción, por lo que deberemos seleccionar un subconjunto de los mismos. Además, el modelo deberá determinar el mix de rutas para cada ítem, la carga de las máquinas y la asignación de herramientas a máquinas.

Para solucionar el modelo hemos usado $\alpha = 0.1$, esto es, un desequilibrio máximo del 20%. El modelo consta de 100 restricciones, 29 variables continuas, 10 variables binarias y 57 variables enteras. El tiempo de CPU requerido para obtener la solución óptima es 4 minutos y 18 segundos. En las Tablas 2.8 y 2.9 se muestran los resultados de selección de ítems, mix de rutas, asignación de herramientas y carga de máquinas.

Tabla 2.8. Ítems seleccionados y mix de rutas

Ítem (i)	Selección Y_i	Mix de rutas		
		θ_{11}	θ_{12}	θ_{13}
1	1	0.354	0.514	0.132
2	1	0	1	0
3	1	0.8	0.2	-
4	0	0	0	-
5	1	0.235	0.765	-
6	0	0	0	-
7	0	0	0	0
8	0	0	0	-
9	1	0.350	0.650	0
10	0	0	0	-

Tabla 2.9. Asignación de herramientas y carga de máquinas (selección de ítems)

j	Asignación de herramientas (ξ_{jk})																				Posic ocup	u_j
	Herramienta (k)																					
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20		
1	-	1	1	-	1	1	-	1	2	-	1	1	-	1	1	-	1	2	-	1	23	9863
2	1	-	1	5	-	2	2	-	1	2	0	1	1	-	2	1	-	1	1	-	34	8691
3	1	1	-	1	1	-	1	2	-	1	1	-	1	1	1	1	-	1	1	-	23	8957
4	-	1	1	-	1	1	1	1	1	-	1	1	-	1	1	1	1	1	-	1	19	8355

Obsérvese que el ítem 4 no ha sido seleccionado a pesar de tener mayor prioridad que otros ítems sí incluidos en el plan de producción. La razón es que los ítems 5 y 9 juntos presentan una carga de trabajo similar a la del 4 pero la suma de sus prioridades es mayor que la de éste. En general, el modelo intentará seleccionar el mayor número de ítems (teniendo en cuenta las prioridades) que las restricciones permitan.

2.5 RESUMEN.

En este capítulo se han formulado dos problemas básicos en la planificación de la producción de los SFF: carga de máquinas y selección de ítems. En la primera formulación se abordan conjuntamente las decisiones de asignar operaciones y herramientas a las máquinas, con el objetivo de equilibrar las cargas de trabajo de las mismas. En la segunda problemática, además de las decisiones anteriores, se abordan las de seleccionar los ítems que van a ser conjuntamente procesados en el SFF. En esta formulación se usan como objetivos el del equilibrado de las cargas, incluido mediante restricciones, y el de maximizar la suma de prioridades de los ítems seleccionados. En los modelos propuestos, la asignación de operaciones a máquinas no se hace de forma individual, sino que para cada ítem se determina su mix agregado de rutas. Además ambos modelos consideran tanto la flexibilidad de rutas de los SFF, como la posibilidad de asignar más de una copia de las herramientas según los requerimientos de tiempo que exista sobre las mismas.

3.1. INTRODUCCIÓN.

En este capítulo se diseña un sistema para la estimación de los costes de producción en los SFF. El sistema de costes está basado en metodologías ABC (del inglés Activity-Based Costing). Inicialmente se presentan las características básicas de las metodologías ABC, así como las razones que inducen a aplicar éstas en los SFF. Posteriormente se diseña el sistema de costes propuesto, el cual está basado en un módulo para la planificación de la producción y otro para la simulación de la operación del taller. Una vez presentadas la funcionalidad básica de la aplicación informática que implementa el sistema de costes, se ilustra éste haciendo uso de los datos de un ejemplo. Además, se analiza la sensibilidad de los costes de producción frente a diversas modificaciones en los datos de entrada al sistema de costes.

3.2. MOTIVACIÓN.

La estimación de costes basados en actividades es uno de los últimos avances en el campo de gestión de costes (Johnson, (1991) y Kaplan, (1991)). Estos sistemas de costes están motivados por dos cambios importantes:

1. El incremento del peso de los costes indirectos en los costes de los sistemas de producción y de servicios. En los sistemas con altos costes indirectos, la forma en la que se asignan éstos puede influir significativamente en la toma de decisiones. Como consecuencia, los sistemas de asignación de costes que mejor reflejen el consumo real de recursos por parte de los productos o servicios, mejorarán de forma significativa el rendimiento del sistema.
2. El desarrollo de las tecnologías de la información permite disponer de datos del sistema de forma eficiente y en tiempo real. Por lo tanto, el consumo de recursos por parte de los productos en las distintas estaciones de trabajo puede ser medido y almacenado con precisión (Cooper, 1988).

Los SFF son un escenario adecuado para la aplicación de las metodologías ABC debido a confluir en los mismos las dos características anteriormente comentadas. Así, los SFF presentan altos costes fijos por varias razones:

- El alto valor de las instalaciones (máquinas, sistema de manejo de materiales, sistemas de control por computador, etc.) hace que la depreciación en los SFF sea bastante mayor que en otros sistemas de producción.
- La operación de forma continuada requiere de sofisticadas actividades, normalmente realizadas por equipos altamente especializados.
- Varios tipos de costes, los cuales tradicionalmente se consideran como directos, son tratados como indirectos en los SFF (por ejemplo los referidos a la mano de obra).

De otro lado, los SFF disponen de un sofisticado sistema de información, soportado por el sistema de control por computador, por lo que los datos requeridos por los sistemas ABC pueden ser fácilmente obtenidos. Además, otra importante razón que avala la aplicación de metodologías ABC en los SFF, es la peculiar forma de operación de los mismos. Así, en la ilustración del capítulo 2 (apartado 2.4), se ha mostrado que en un mismo período de planificación, las piezas de un mismo ítem pueden procesarse en distinta forma debido a las rutas alternativas, distintas configuraciones de las máquinas, etc. Por otro lado, de un período de planificación a otro, tanto el mix de ítems como la configuración del SFF son susceptibles de cambio. Estas diferentes y cambiantes formas de procesado de los ítems se traducen en distinto consumo en los recursos de producción, por lo que de alguna manera se deberían traducir en distintos costes de los mismos. Todas estas características

justifican la aplicación de ABC en los SFF, con el objetivo de evitar la distorsión de los altos costes fijos en los costes de producción.

3.3. COSTES BASADOS EN ACTIVIDADES.

La idea básica de ABC es asignar a cada producto aquella parte de los costes indirectos del sistema que esté justificada por el consumo relativo de recursos por parte de cada producto. Los costes indirectos de producción pueden ser vistos como los costes necesarios para crear y mantener los recursos disponibles del sistema de producción.

El procesado de un producto viene definido por un conjunto de actividades, cada una de las cuales requerirá diferentes recursos productivos. Lo ideal sería el conocimiento exacto del consumo de recursos por parte de cada una de las actividades de los productos. No obstante, esto redundaría en una elevada e innecesaria complicación. Para simplificar, sin perder de vista la idea básica de ABC, las actividades con similares características en cuanto a consumo de recursos son agrupadas en centros de actividad (Cooper, 1989). El coste indirecto correspondiente a cada centro de actividad se asigna a los productos en base a los llamados generadores de costes. A pesar de que se han publicado diversos artículos sobre la implementación de sistemas ABC en sistemas de producción tradicionales y de servicios (Dhavale, 1992; Ness y Cucuzza, 1995; y Thomas y Mackey, 1994) todavía no se han aplicado a los SFF.

3.4. DISEÑO DEL SISTEMA DE COSTES PARA SFF.

El problema que nos planteamos en este apartado es el diseño de un sistema para la estimación del coste total de producción en los SFF. Inicialmente, se hace una descripción del escenario en el que se va a diseñar el sistema de costes. A continuación, se presentan los modelos de planificación de la producción y simulación, los cuales permitirán calcular las tasas y bases de asignación de los centros de actividad considerados para imputar los costes indirectos.

3.4.1. DESCRIPCIÓN DEL ESCENARIO.

Consideramos un SFF compuesto por un número conocido de máquinas, las cuales serán dotadas para la realización de las operaciones de procesado de copias de los distintos tipos de herramientas disponibles. Cada ítem puede ser procesado por distintas rutas alternativas, las cuales vendrán definidas por el conjunto de operaciones a realizar sobre las piezas. Cada una de las piezas a ser procesadas entrará al sistema por la estación de carga/descarga, donde será cargada en uno de los pallets, los cuales están disponibles en un número limitado. Los transportes de las piezas entre las distintas máquinas se realizan mediante vehículos. Se supone que no existe limitación de capacidad en los dispositivos de almacenamiento de las máquinas destinados al inventario en proceso. Los requerimientos de producción vienen definidos por los datos de los pedidos de los clientes. Cada pedido vendrá determinado por tres características: ítem demandado, cantidad requerida y fecha de entrega del pedido. Si un cliente realiza peticiones que involucran varios ítems, se constituyen varios pedidos. Un pedido estará disponible para su entrega al cliente cuando se haya completado el proceso de todas las piezas del mismo.

Dados los datos de los pedidos y los del sistema de producción, el primer paso será planificar la producción. En este caso, el problema de selección de ítems puede ser visto como uno de selección de pedidos. Estos es, tendremos que decidir por un lado, qué pedidos serán procesados de forma simultánea en el sistema, y por otro, la configuración del mismo, esto es, mix de rutas para cada pedido y asignación de herramientas a máquinas. Todas estas decisiones estarán obviamente restringidas por las limitaciones de

capacidad del sistema de producción. A continuación, una vez planificada la producción, se iniciaría la operación del taller, la cual finalizará una vez que se procesen todas las piezas de los pedidos seleccionados.

Dentro de este entorno, el sistema de costes debe ser tal que se calculen los costes unitarios totales (suma de los directos más indirectos) de cada uno de los pedidos procesados en el período actual. Ante un nuevo período, se repetiría la metodología: planificación de la producción, operación del taller y estimación de costes de producción.

3.4.2. MODELO DE PLANIFICACIÓN DE LA PRODUCCIÓN.

Para determinar el plan de producción así como la configuración del SFF, vamos a utilizar una modificación del modelo descrito en el capítulo 2 para la selección de ítems. El modelo utilizado se planteará la cuestión de decidir qué pedidos procesar simultáneamente además de las referidas al mix de rutas (en este caso de cada pedido) y asignación de herramientas a máquinas.

Los datos de entrada al modelo de planificación serán:

- M: número de máquinas.
- w_q : prioridad del pedido q (calculada como el inverso de los días que le quedan al pedido hasta su fecha de entrega).
- D_q : unidades demandadas en el pedido q .
- p_{qijk} : tiempo de procesado para cada pieza del pedido q cuando siguen la ruta r en la máquina j y usando la herramienta k .
- L_k : vida útil de la herramienta k .
- A_k : copias disponibles de la herramienta k .
- C_j : capacidad del almacén de herramientas de la máquina j .
- S_{jk} : posiciones ocupadas por una copia de la herramienta k en la máquina j .

Las variables de decisión son:

- y_q : variable 0-1 para indicar que el pedido ha sido seleccionado o no.
- θ_{qr} : proporción de piezas del pedido q procesadas por la ruta r .
- ξ_{jk} : número de herramientas del tipo k asignadas a la máquina j .
- u_j : carga de la máquina j .
- u_{med} : carga media de las máquinas.

Parámetros:

- α : desequilibrio de las cargas de trabajo (medido como fracción de la carga media).

El modelo completo viene dado por:

$$\text{maximizar } \sum_q w_q y_q \quad (3.1)$$

sujeto a:

$$\sum_q \theta_{qr} = y_q \quad \forall i \quad (3.2)$$

$$\sum_q \sum_r D_q p_{qrjk} \theta_{qr} \leq L_k \xi_{jk} \quad \forall j,k \quad (3.3)$$

$$\sum_j \xi_{jk} \leq A_k \quad \forall k \quad (3.4)$$

$$\sum_k s_{jk} \xi_{jk} \leq C_j \quad \forall j \quad (3.5)$$

$$\sum_q \sum_r \sum_k D_q p_{qrjk} \theta_{qr} = u_j \quad \forall j \quad (3.6)$$

$$\sum_j u_j = M u_{med} \quad (3.7)$$

$$(1 - \alpha) u_{med} \leq u_j \leq (1 + \alpha) u_{med} \quad \forall j \quad (3.8)$$

$$\xi_{jk} \geq 0 \text{ enteras} \quad \forall j,k; y_q \in \{0,1\} \quad (3.9)$$

$$\theta_{qr} \geq 0 \quad \forall q,r; u_j \geq 0 \quad \forall j; u_{med} \geq 0 \quad (3.10)$$

Los pedidos con fechas de entrega más próximas presentarán mayor prioridad, como consecuencia, en la función objetivo (3.1) se prioriza la selección de los pedidos más urgentes. En (3.2) se impone para cada pedido que la suma de las proporciones procesadas por sus rutas suman la unidad para los pedidos seleccionados y cero para los no seleccionados. Las restricciones (3.3) imponen que el número de copias de herramientas de cada tipo sean suficientes para cubrir los requerimientos de las mismas. Las restricciones (3.4) y (3.5) son la limitación de copias de herramientas y capacidades de los almacenes de las máquinas respectivamente. En (3.6) se computa la carga de las máquinas, mientras que en (3.7) se fija la carga media de las máquinas. En (3.8) se impone que la carga de cada una de las máquinas esté comprendida en la banda de amplitud $2\alpha u_{med}$ alrededor de u_{med} .

3.4.3. MODELO DE SIMULACIÓN.

Una vez determinado el plan de producción y la configuración del SFF, la operación del taller se simula haciendo uso de un modelo de simulación descrito a partir del Diagrama de Ciclos de Actividades (DCA) del mismo. Las entidades definidas son:

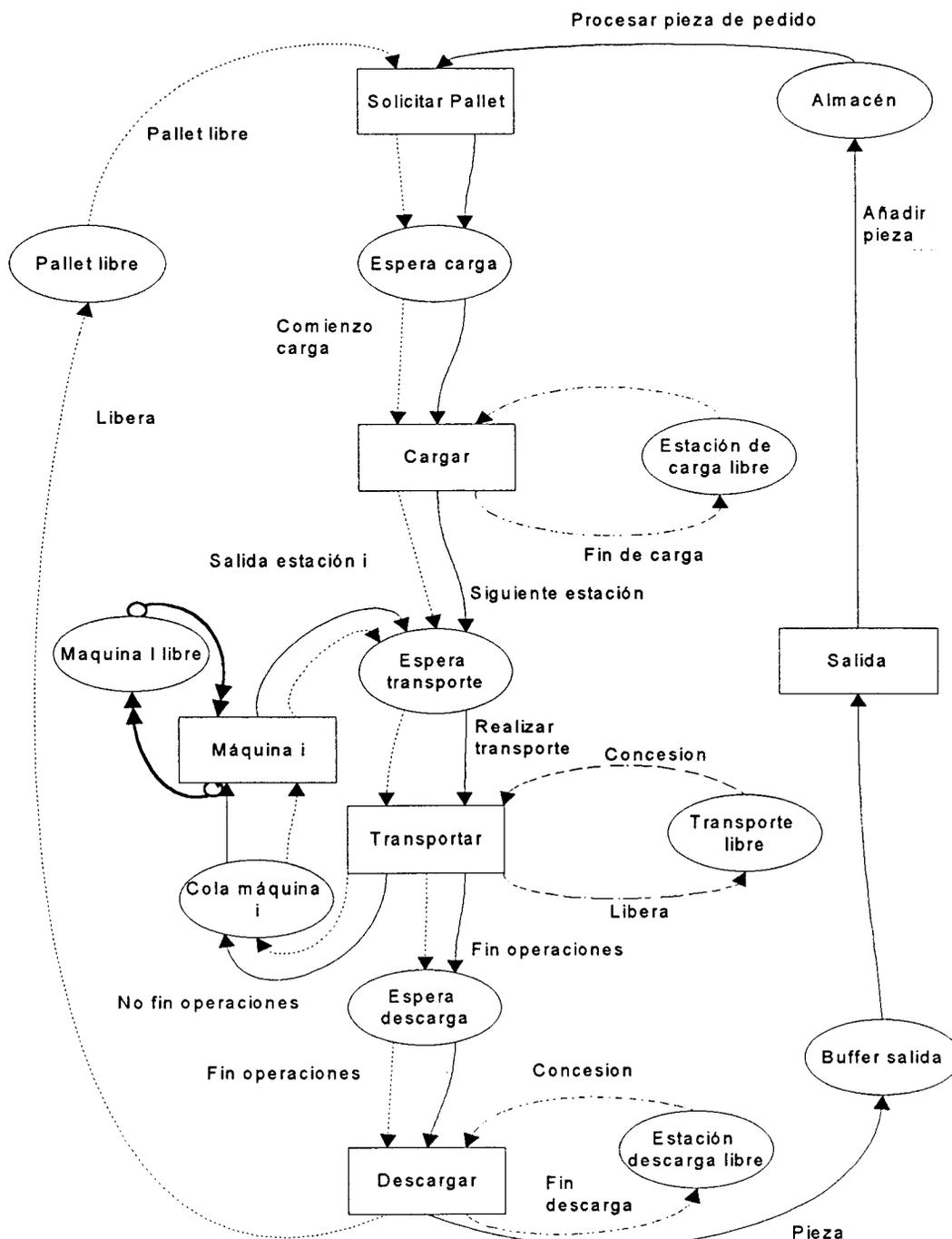
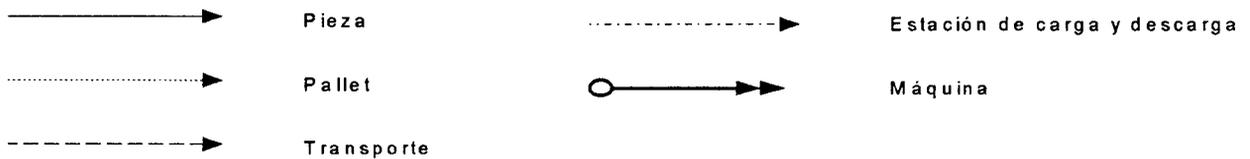


Figura 3.1. Diagrama de Ciclos de Actividades del sistema

Para cada tipo de entidad se representa la sucesión de colas y actividades que conforman su ciclo. Los distintos tipos de entidades se relacionan entre sí a través de las actividades comunes a las mismas. Las piezas entran en el sistema procedentes del almacén en el que se encuentran las piezas que serán procesadas. Para la entrada de una pieza en el taller requieren de un pallet libre. Una vez en el taller, la pieza y el pallet circulan solidariamente hasta su salida del mismo. En la estación de carga/descarga se realiza la actividad de cargar la pieza en el pallet. Posteriormente, un transporte realizará el traslado de la pieza (y el pallet) a la máquina donde se realizará su primera operación. Una vez en la máquina, la pieza espera en la cola correspondiente a que se quede la máquina libre. A partir de este instante se realiza la operación sobre la pieza. Terminada la operación, si la pieza requiere otras operaciones, pasa de nuevo a la cola de espera de transporte. Cuando la pieza finaliza su procesado, pasa a la estación de carga/descarga donde se efectúa la operación de despalletización. Una vez finalizada ésta, la pieza sale del sistema y el pallet queda libre, pudiendo ser usado por otras piezas que van a entrar al sistema.

3.4.4. SISTEMA DE COSTES PROPUESTO.

En la Figura 3.2 se muestra un esquema con el sistema de costes diseñado. El sistema de costes está basado en los dos módulos descritos anteriormente: modelo de planificación de la producción y modelo de simulación de la operación del taller.

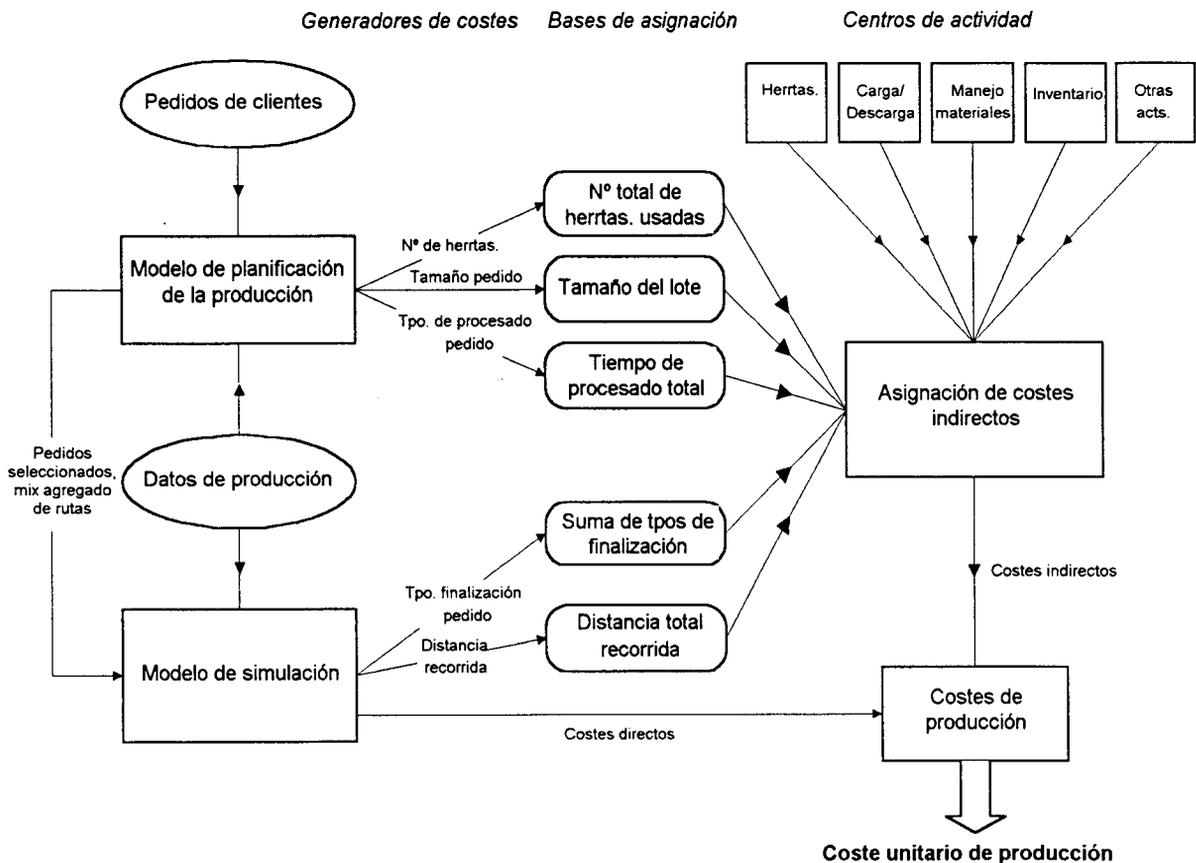


Figura 3.2. Sistema de costes para Sistemas de Fabricación Flexible

Para la imputación de los costes indirectos se han definido cinco centros de actividad:

- Herramientas: Las herramientas son uno de los elementos fundamentales en los SFF. Su elevado coste además de las muchas actividades relacionadas con ellas (calibración, mantenimiento, etc.) hace necesario la definición para las mismas de un centro de actividad.
- Carga/descarga: Las piezas para entrar en el SFF necesitan ser cargadas en pallets. Todas las actividades relacionadas con estas cargas y descargas (fijaciones, etc.) se consideran en este centro de actividad.
- Manejo de materiales: Comprende todo lo referido al transporte de piezas, equipos de control del mismo, etc.
- Inventario: Comprende todas aquellas actividades referidas al inventario de las piezas tales como ubicación, mantenimiento en inventario, etc.
- Otras actividades: En estas actividades se engloban entre otras las correspondientes a control de calidad, mantenimiento, sistema de información, depreciación, etc.

En cuanto a los generadores de costes de cada uno de los centros de actividad considerados se han utilizado los siguientes:

1. Número de herramientas: Para imputar el coste indirecto referido a las herramientas entre los distintos pedidos procesados se calcula el número de herramientas usadas en el procesado del pedido.
2. Tamaño del pedido: Los costes referidos a todas las actividades de carga/descarga se asignan a los distintos pedidos seleccionados en función del tamaño del mismo.
3. Distancia recorrida: Los costes relativos al sistema de manejo de materiales se desglosan entre pedidos de acuerdo a la distancia recorrida por los transportes para cada uno de los pedidos.
4. Inventario: Se imputa a los pedidos de acuerdo al tiempo de terminación de los mismos. Un pedido estará totalmente terminado cuando se procesa la última pieza del mismo.
5. Tiempo de procesado: Los costes englobados en "otras actividades" se van a asignar a los pedidos de acuerdo a la utilización de las máquinas por parte de cada uno de los pedidos.

Dados los datos referidos al sistema de producción, así como las órdenes de los clientes, del modelo de planificación obtenemos qué pedidos son seleccionados, mix de rutas para cada uno de ellos, asignación de herramientas a máquinas y utilización de las máquinas. Las tasas de asignación correspondientes a las herramientas, carga/descarga y otras actividades se calculan a partir del modelo de planificación, como el cociente entre el coste indirecto del centro de actividad, y la correspondiente base:

- Tasa de asignación del coste de "herramientas":

$$T.H. = \frac{\text{Coste Indirecto de "herramientas"}}{\sum_k \frac{1}{L_k} \sum_q \sum_j \sum_r D_q p_{qjrk} \theta_{qr}} \quad (3.11)$$

- Tasa de asignación del coste de "Carga /descarga":

$$T.C/D.= \frac{\text{Coste Indirecto de "carga / descarga"}}{\sum_q D_q Y_q} \quad (3.12)$$

- Tasa de asignación del coste de "otras actividades":

$$T.O.A.= \frac{\text{Coste Indirecto de "otras actividades"}}{\sum_q \sum_k \sum_j \sum_r D_q p_{qjk} \theta_{qr}} \quad (3.13)$$

Para calcular el tiempo de terminación de cada uno de los pedidos así como la distancia recorrida por los transportes, hacemos uso del modelo de simulación. Como entradas a la simulación, además de todos los datos relativos al sistema de producción (máquinas, distancia entre ellas, número de transportes, etc.) necesitamos saber del modelo de planificación los pedidos a procesar y el mix de rutas para cada pedido. Para realizar la simulación necesitamos fijar la secuencia de entrada de las piezas en el sistema así como la ruta concreta que sigue cada pieza. Estas dos variables se han fijado de la siguiente forma: la secuencia de introducción de las piezas en el sistema es aleatoria y las rutas de cada pieza se toman de forma aleatoria pero siguiendo las salidas de la planificación referidas al mix de rutas. El cálculo de las tasas correspondientes tanto al inventario como al sistema de manejo de materiales se obtienen, a partir del modelo de simulación, como cociente entre los correspondientes costes indirectos y sus bases de asignación:

- Tasa de asignación del coste de "inventario":

$$T.I.= \frac{\text{Coste Indirecto de "inventario"}}{\sum_q \text{Tiempo de terminación}_q} \quad (3.14)$$

- Tasa de asignación del coste "manejo de materiales":

$$T.M.M.= \frac{\text{Coste Indirecto de "manejo de materiales"}}{\sum_q \text{Distancias recorridas}_q} \quad (3.15)$$

Una vez determinadas las tasas, el coste indirecto imputado a cada pedido de cada uno de los centros de actividad se calcula multiplicando la correspondiente tasa por el consumo (de la actividad del centro) por parte del pedido. Los costes indirectos unitarios para cada pedido se obtienen dividiendo entre el tamaño del pedido.

Para el cálculo del coste total de cada producto quedan por determinar los costes directos. Para los costes directos se han considerado dos términos. El primero, un pedido está referido a un ítem dado cuya materia prima tiene un determinado coste. La parte del coste directo de un pedido referida a la materia prima no es más que la multiplicación de las unidades que componen el pedido por el coste de la materia prima de cada unidad. El segundo término es el coste de las operaciones de las rutas. Asociando a cada operación un coste unitario, la parte del coste directo de un pedido referida a las rutas se calculará multiplicando las piezas del pedido que siguen cada ruta por el coste de la ruta (suma de los costes de las operaciones de la ruta). Para calcular este segundo término

necesitamos tomar de la simulación el número de piezas de cada pedido que sigue cada una de las rutas.

El coste total de producción de cada pedido será la suma de los indirectos más los directos. A su vez los costes indirectos son suma de los términos correspondientes a cada una de los cinco centros de actividad. El coste unitario de producción de un pedido es el cociente entre el coste total de producción y el tamaño del pedido.

En el anexo II se presentan algunas de las características de la aplicación informática que implementa el sistema de costes diseñado.

3.5. ILUSTRACIÓN.

En este apartado se ilustra la funcionalidad del sistema de costes propuesto. Para ello, inicialmente se presentan los datos de entrada de un ejemplo. A continuación, se realiza la estimación de costes a partir de los modelos de planificación y simulación.

3.5.1. DATOS DE ENTRADA.

Consideramos un sistema compuesto por 2 máquinas ($M=2$) y una estación de carga/descarga. Para la fijación y transporte de las piezas se usan cuatro pallets y 4 vehículos respectivamente. Se disponen de 6 tipos de herramientas. Para cada tipo se tienen 3 copias ($A_k=3$). Los almacenes de herramientas de las máquinas tienen una capacidad de 10 posiciones ($C_j=10$). El número de posiciones que cada tipo de herramienta ocupa en las máquinas es igual a uno ($S_{jk}=1$). Las vidas útiles de cada tipo de herramienta son iguales a 125 minutos ($L_k=125$).

En la Tabla 3.1 se muestra los pedidos recibidos por los clientes, los cuales involucran cinco tipos de piezas. Para cada pedido se conoce el tipo de las piezas requeridas, cantidad demandada (D_q) y prioridad (W_q). La prioridad ha sido calculada como la inversa de los días que quedan hasta la fecha del pedido. De esta forma pedidos más urgentes tendrán mayor prioridad.

Tabla 3.1. Datos de los pedidos

Pedido (q)	Ítem	Demanda (D_q)	Prioridad (w_q)
1	1	20	0.10
2	1	16	0.17
3	2	12	0.17
4	2	20	1.00
5	3	14	0.10
6	3	20	0.20
7	4	16	0.20
8	4	14	0.50
9	5	20	0.50
10	5	20	0.10

En la Tabla 3.2 se especifican las rutas alternativas para cada uno de los cinco tipos de piezas. Cada ruta viene especificada por las operaciones a realizar. Cada operación viene descrita por tres números que denotan: máquina en la que se realiza, tipo de herramienta usada y tiempo de operación. Así por ejemplo, la primera ruta del ítem 1

comprende 3 operaciones que se realizarán todas ellas en la primera máquina. La primera operación requiere la herramienta tipo 2 y dura 10 minutos, la segunda operación requiere la herramienta tipo 3 y requiere 12 minutos y la tercera operación requiere la herramienta tipo 5 durando 15 minutos.

Tabla 3.2. Rutas Alternativas de los Ítems

Ítem	Ruta	Datos de las Rutas (maquina, herramienta, tiempo de operación en minutos)			
		1ª operación	2ª operación	3ª operación	4ª operación
1	1	(1,2,10)	(1,3,12)	(1,5,15)	
	2	(2,1,10)	(2,4,15)	(2,6,12)	
	3	(2,1,2)	(1,1,5)	(2,2,5)	(1,5,7)
2	1	(1,2,7)	(1,3,10)	(1,5,10)	
	2	(2,1,10)	(2,4,10)	(2,6,7)	
	3	(1,3,2)	(2,5,15)	(1,4,5)	
3	1	(1,1,7)	(1,4,10)		
	2	(2,2,7)	(2,4,5)	(2,6,5)	
	3	(2,6,7)	(1,1,5)		
4	1	(1,1,12)	(1,4,15)		
	2	(2,2,5)	(2,4,2)	(2,6,7)	
	3	(1,4,10)	(2,2,10)	(1,1,7)	
5	1	(1,2,10)	(1,4,5)	(1,5,5)	
	2	(2,1,5)	(2,3,10)	(2,5,5)	
	3	(2,4,5)	(1,6,5)	(2,5,10)	

3.5.2. ESTIMACIÓN DE COSTES.

A partir de los datos de entrada anteriormente descritos, el primer paso será planificar la producción. Para ello, se resuelve el modelo de programación lineal entera mixta (3.1)-(3.10) con el desequilibrio máximo definido por $\alpha = 0.1$. En las Tablas 3.3 y 3.4 se presentan los resultados ofrecidos por el optimizador para el modelo de planificación de la producción. En la Tabla 3.4 se presenta la información referida a los pedidos seleccionados ($Y_q=1$) y el mix de rutas para cada uno de ellos (θ_{qr}). Así por ejemplo, el pedido 3 es seleccionado y el 62% de las piezas del mismo siguen la ruta 2 y el resto la 3. La Tabla 3.4 contiene la asignación de herramientas a máquinas (ξ_{jk}) y la carga de las máquinas (u_j). Obsérvese que todas las copias de todos los tipos de herramientas se usan. La carga de trabajo de las máquinas presenta un desequilibrio del 2.5%.

Tabla 3.3. Pedidos seleccionados y mix de rutas

Pedido (q)	(Y _q)	Mix de rutas		
		θ_{q1}	θ_{q2}	θ_{q3}
1	0	0	0	0
2	0	0	0	0
3	1	0	0.62	0.38
4	1	1	0	0
5	0	0	0	0
6	1	0	0	1
7	1	0.39	0.61	0
8	1	0	1	0
9	1	0.5	0.5	0
10	0	0	0	0

Tabla 3.4. Asignación de herramientas y utilización de máquinas.

Máquina(j)	Asignación de Herramientas (ξ_{jk})						Número de posiciones usadas	Utilización de Máquinas
	Tipo de Herramienta (k)							
	1	2	3	4	5	6		
1	2	2	2	2	2	0	10	1041
2	1	1	1	1	1	3	8	942
Total	3	3	3	3	3	3		1983

Para calcular la distancia recorrida por los transportes para el procesado de las piezas correspondientes a los pedidos seleccionados, así como los tiempos de finalización de los mismos, se ha usado el modelo de simulación descrito anteriormente. Para la ejecución de la simulación se ha especificado: secuencia de entrada de las piezas (aleatoria), distancia entre máquinas y velocidad de los vehículos, etc.

En la Tabla 3.5 se presentan los datos de los costes asociados a los centros de actividad. Para cada centro de actividad se ha considerado un coste horario de la misma. El coste de la actividad será la multiplicación de este coste horario por las horas de funcionamiento del taller (obtenidas de la simulación). Además se muestran las bases y tasas de asignación de cada centro, calculadas de la forma descrita en el apartado 3.4.4.

Tabla 3.5. Bases y tasas de asignación de costes indirectos

Centro de Actividad	Coste horario	Coste indirecto	Base	Tasa
Herramientas	55	1069	15.9	67.23
Carga /Descarga	40	777	102	7.62
Manejo de materiales.	50	971	966	1.01
Inventario	35	608	6203	0.109
Otras Actividades.	100	1943	1983	0.98

Los costes unitarios de producción de cada uno de los pedidos seleccionados en el ejemplo se muestran en la Tabla 3.6. Para cada pedido seleccionado se presentan cada uno de los costes indirectos unitarios además de los directos. Obsérvese que aproximadamente el 70% del coste total de un producto es indirecto.

Tabla 3.6. Costes unitarios de producción

Pedido	Coste indirecto unitario					Coste directo unitario	Coste unitario de Producción
	Herrtas.	Carga/ Descarga	Manejo Materiales	Inventario	Otras Actividades		
1	-	-	-	-	-	-	-
2	-	-	-	-	-	-	-
3	13.54	7.62	12.15	9.43	24.62	30.23	97.59
4	14.55	7.62	12.07	6.39	26.45	31.10	98.18
5	-	-	-	-	-	-	-
6	6.47	7.62	7.29	6.34	11.76	20.60	60.08
7	10.30	7.62	9.87	7.75	18.72	28.66	82.92
8	7.54	7.62	7.40	4.48	13.72	27.20	67.96
9	10.78	7.62	8.85	6.28	19.59	23.00	76.12
10	-	-	-	-	-	-	-

3.6. EXPERIMENTOS ADICIONALES.

En este apartado se va a analizar la sensibilidad de los costes de producción frente a variaciones tanto en el tamaño y prioridad de los pedidos, así como ante la inclusión de nuevos pedidos urgentes (alta prioridad). Para ello se han diseñado 6 nuevos escenarios a partir de los datos del ejemplo descrito en el apartado 3.6. En la Tabla 3.7 se muestran las variaciones que generan cada uno de los nuevos escenarios. Así por ejemplo, en el escenario I sólo se modifica el tamaño del pedido 9, el cual se incrementa de 20 a 25. Los dos últimos escenarios corresponden a la consideración de un nuevo pedido de alta prioridad.

Tabla 3.7. Definición de escenarios

	Escenario I	Escenario II	Escenario III	Escenario IV	Escenario V	Escenario VI
Pedido(q)	9	9	5	9	11	11
Tamaño (D _a)	20 ⇒ 25	20 ⇒ 15	igual	igual	20	10
Prioridad (w _a)	igual	igual	0.1 ⇒ 0.2	0.5 ⇒ 0.1	1.0	1.0
Ítem	igual	igual	igual	igual	1	1

3.6.1. ANÁLISIS DE SENSIBILIDAD.

Recalculando para cada escenario los costes (para cada escenario se vuelve a planificar y simular) obtenemos los mostrados en la Tabla 3.8.

Tabla 3.8. Coste unitarios de producción para los nuevos escenarios.

Pedido	Ejemplo	Escenario I	Escenario II	Escenario III	Escenario IV	Escenario V	Escenario VI
1	-	-	-	-	-	-	-
2	-	-	81.12	-	90.91	-	-
3	97.59	-	-	-	-	-	-
4	98.18	95.18	98.31	88.36	96.54	103.72	99.71
5	-	61.89	-	62.75	73.96	-	-
6	60.08	63.33	75.52	55.93	62.65	-	70.21
7	82.92	77.67	71.91	74.29	69.94	-	92.94
8	67.96	68.73	77.30	66.64	73.16	79.44	71.35
9	76.12	74.63	79.71	72.08	-	86.16	80.00
10	-	-	-	-	-	-	-
11						94.88	81.96

En las Tablas 3.9 y 3.10 se presentan las variaciones en porcentaje del tiempo total de producción así como de las bases y tasas de asignación de costes.

Tabla 3.9. Cambios en el tiempo total de producción (makespan)

Ejemplo	Cambios expresados en tanto por ciento						
	Escenario I	Escenario II	Escenario III	Escenario IV	Escenario V	Escenario VI	
Makespan (en horas)	19.43	-1.29	3.34	-13.29	-3.17	-11.23	4.80

Tabla 3.10. Cambios en las bases y tasas de asignación.

Centro Actividad	Ejemplo		Cambios en tanto por ciento											
	Base	Tasa	Escenario I		Escenario II		Escenario III		Escenario IV		Escenario V		Escenario VI	
			Base	Tasa	Base	Tasa	Base	Tasa	Base	Tasa	Base	Tasa	Base	Tasa
Herramientas	15.9	67.23	-1.2	-0.12	-3.77	7.60	-8.80	-5.15	-9.43	4.77	-16.35	6.43	0	4.76
Carga/Descarga	102	7.62	6.86	-7.61	-0.98	4.33	1.96	-14.96	-1.96	-1.18	-27.45	22.44	-1.96	6.96
Manejo Material	966	1.01	5.48	-6.93	4.24	-0.99	2.17	-15.84	-0.10	-2.97	-24.12	16.83	7.56	2.97
Inventario	6203	0.109	-11.11	11.13	-1.35	4.84	-14.67	1.64	-11.22	9.12	-44.20	59.12	-4.95	10.31
Otras Actividades	1983	0.98	-0.96	0	-3.78	7.14	-8.37	-5.1	-7.36	4.08	-16.39	6.12	0.30	4.08

3.6.2. DISCUSIÓN.

Pueden hacerse algunos comentarios respecto a los resultados de los costes en cada uno de los seis escenarios. Así, en el Escenario I se incrementa el tamaño del pedido 9. Debido a su alta prioridad este pedido permanece como seleccionado, no obstante, el pedido 3 se cae del lote mientras que el 5 entra en el mismo. Estos cambios dan lugar a un incremento del tamaño del lote de pedidos procesados simultáneamente, y un ligero descenso del tiempo total de producción (véase Tabla 3.9). Todo esto sugiere un ligero incremento en la eficiencia que se traduce en promedio en menores tasas de asignación.

En el Escenario II disminuye el tamaño del pedido 9. La capacidad de producción liberada permite la sustitución del pedido 3 por el 2 el cual tiene la misma prioridad pero mayor tamaño. Aunque el tamaño del lote se decremента en una unidad, el tiempo total de producción se incrementa. Este descenso en la eficiencia da lugar en promedio a tasas de asignación mayores.

En el Escenario III se incrementa la prioridad del pedido 5 que no se encontraba incluido en el lote. Como resultado el lote se reorganiza entrando el pedido 5 y saliendo el 3. Aunque el tamaño del lote se incrementa, el tiempo total de producción baja significativamente. Este aumento en la eficiencia provoca de medio menores tasas de asignación.

En el Escenario IV el descenso en la prioridad en el pedido 9 provoca su salida del lote de pedidos seleccionados. De nuevo la capacidad liberada permite la sustitución del pedido 3 por el 2 que tiene la misma prioridad pero mayor tamaño. El tamaño del lote baja en dos unidades y el tiempo total de producción también baja ligeramente. La eficiencia se mantiene y los cambios en las tasas de asignación son menos significativos.

En el Escenario V se considera un nuevo pedido (el 11) de gran prioridad y tamaño elevado. Esta orden urgente provoca la caída del lote de varios pedidos de prioridad baja (pedidos 3, 6 y 7). El tamaño del lote baja considerablemente mientras que el tiempo total de producción baja ligeramente. Esta menor eficiencia da lugar a unas mayores tasas de asignación. El coste unitario de producción de todos los pedidos que permanecen se incrementa debido a que tienen que soportar todos los costes indirectos.

En el Escenario VI la orden urgente tiene un menor tamaño. Únicamente el pedido 3 es desplazado. El tamaño del lote baja en dos unidades mientras que el tiempo total de producción se incrementa. La menor eficiencia provoca mayores tasas de asignación aunque menores que las del Escenario V. Por esta razón, el coste unitario de producción de la orden urgente de menor tamaño (81.96) es menor que el coste de la de mayor tamaño (94.88). El coste unitario de producción del resto de los pedidos se incrementa pero no tanto como en el Escenario V.

3.7. RESUMEN.

En este capítulo se ha diseñado un sistema de costes para SFF, donde la característica básica es la adaptabilidad de la estimación de costes a las peculiaridades características de operación de los SFF. Ésta se consigue enlazando el sistema de costes con la fase de planificación de la producción. Toda la metodología diseñada ha sido ilustrada a partir de los datos de un ejemplo. Mediante distintos experimentos se ha mostrado la sensibilidad de los costes de producción frente a pequeñas variaciones en los datos de entrada.

4.1 INTRODUCCIÓN.

En este capítulo nos planteamos ofrecer soluciones para la formulación del modelo del problema de carga, (2.1)-(2.9), propuesta en el apartado 2.2.3. Como se comentó en dicho apartado, la integridad de las variables que modelan la asignación de herramientas a máquinas, así como la estructura de las restricciones (2.4) y (2.5), recuérdese que son múltiples restricciones del tipo de la mochila, hacen que el modelo de programación lineal entera mixta no sea fácilmente resoluble en el peor de los casos.

Para la obtención de soluciones del problema original se diseñan dos procedimientos heurísticos basados en relajaciones continuas de la formulación de programación lineal entera mixta. La idea básica en la que se apoyan ambos procedimientos es la de fijar el valor de la asignación de las herramientas a máquinas (variables enteras) y reducir el problema a uno de programación lineal continua en el que se determina el mix agregado de rutas para cada uno de los ítems. Para la evaluación de ambos procedimientos se han generado aleatoriamente dos conjuntos de problemas de dificultad creciente. La calidad ofrecida por las heurísticas así como el tiempo de computación se comparan con procedimientos estándares de ramificación y acotación. Aunque no se garantiza la obtención de admisibilidad en los procedimientos diseñados, las experiencias computacionales demuestran que, no sólo ésta se consigue un elevado número de veces, sino que además el esfuerzo computacional requerido es muy bajo.

4.2 SOLUCIONES HEURÍSTICAS.

Los dos procedimientos heurísticos diseñados son estructuralmente similares. Inicialmente, se fija el valor de las variables continuas que representan el mix agregado de rutas para cada ítem (θ_{ir}^*). A continuación, se calcula la asignación de herramientas a máquinas (ξ_{jk}) requerida por el mix de rutas fijado. A partir de esta asignación de herramientas a máquinas, se hace una búsqueda exhaustiva de admisibilidad en las múltiples restricciones de la mochila, (2.4)-(2.5). Por último, se determina el mix agregado de rutas, compatible con la nueva asignación de herramientas a máquinas (admisibles en las múltiples restricciones de la mochila (2.4)-(2.5)), que da lugar a un menor desequilibrio. Este último paso se implementa mediante la resolución de un problema de programación lineal continua donde las variables de decisión son el mix agregado de rutas para cada uno de los ítems. A continuación, se describen tanto los pasos comunes a ambos procedimientos heurísticos como los específicos de cada uno de ellos.

4.2.1 PASOS COMUNES.

Suponiendo fijado el mix agregado de rutas para cada uno de los ítems θ_{ir}^* , el número de herramientas a asignar a las máquinas de las estaciones se calcula según:

$$\xi_{jk}^* = \left\lceil \frac{\sum_i \sum_r D_i p_{irjk} \theta_{ir}^*}{m_j L_k} \right\rceil \quad \forall k, J(k) \quad (4.1)$$

Esto es, el cociente calcularía el número de herramientas sin tener en cuenta que tiene que ser un número entero. Para fijar el número de herramientas a asignar se toma el entero inmediatamente superior.

El siguiente paso será asegurar que la asignación de herramientas a máquinas (4.1)

cumple las restricciones de la mochila (2.4)-(2.5). Para ello, dada la asignación ξ_{jk}^* se calculan las holguras de las restricciones tipo de la mochila (véase Tabla 4.1). Posteriormente, y junto con la asignación ξ_{jk}^* , se construyen los conjuntos representados en la Tabla 4.1.

Tabla 4.1. Notación en heurísticas

Holguras:	
$g_k^* = \sum_{j \in J(k)} m_j \xi_{jk}^* - A_k \quad \forall k$	holguras de las restricciones (2.4)
$h_j^* = \sum_{k \in K(j)} s_{jk} \xi_{jk}^* - C_j \quad \forall j$	holguras de las restricciones (2.5)
Conjuntos:	
$K^* = \{k: g_k^* > 0\}$	herramientas cuya restricción tipo (2.4) es violada
$J^* = \{j: h_j^* > 0\}$	estaciones cuya restricción tipo (2.5) es violada
$JK^* = \{jk: \xi_{jk}^* \geq 1\}$	combinaciones estación-herramienta con asignación positiva

Si los conjuntos J^* y K^* están vacíos, tendríamos una solución original para el problema de carga. En caso contrario (ξ_{jk}^* es inadmisibles en alguna de las restricciones (2.4) y (2.5)), la asignación de herramientas a máquinas deberá ser modificada para conseguir esta admisibilidad en (2.4) y (2.5). Para conseguir que ξ_{jk}^* sea admisible en las restricciones tipo de la mochila (2.4) y (2.5), se ha diseñado un procedimiento (PROC_ADM) en el que de forma exhaustiva y ordenada se decrementan variables ξ_{jk}^* que provocan inadmisibilidad en (2.4) y (2.5).

En la Figura 4.1 se presenta un pseudocódigo del procedimiento PROC_ADM. La entrada algoritmo es una asignación de herramientas a máquinas (ξ_{jk}^*). La salida del procedimiento es una asignación de herramientas a máquinas que admisible en cuanto a las restricciones de la mochila (2.4)-(2.5). En el paso 1 se construyen los conjuntos (K^* y J^*) que recogen los índices de las restricciones de la mochila violadas. Además, en el conjunto JK^* se recogen los índices de las correspondientes asignaciones estrictamente positivas, esto es, las que son candidatas a ser decrementadas. En el caso de que la asignación de herramientas a máquinas sea inicialmente admisible en (2.4)-(2.5), se termina el procedimiento (paso 2.A). En caso contrario, se decrementan de una en una asignaciones de herramientas a máquinas involucradas en restricciones violadas. Con el objetivo de realizar estas operaciones en el menor número de pasos posibles (modificando el menor número de asignaciones), el orden en el decremento en las asignaciones se hace teniendo en cuenta su impacto en la inadmisibilidad de las restricciones (2.4)-(2.5). Esto es, la elección de la asignaciones a decrementar se hace en orden decreciente de su impacto en la inadmisibilidad.

Paso 1.	Dada ξ_{jk}^* contruir los conjuntos K^* , J^* y JK^*
Paso 2.	
A)	Si $K^* = \phi$, $J^* = \phi \Rightarrow$ Fin
B)	En caso contrario:
B.1)	Si $J^* \times K^* \cap JK^* \neq \phi \Rightarrow j'k' = \left\{ jk : \max_{jk \in J^* \times K^* \cap JK^*} \{g_k^* h_j^*\} \right\}$ $\xi_{j'k'}^* \leftarrow \xi_{j'k'}^* - 1$ e ir a Paso 1
B.2)	Si $J^* \times K^* \cap JK^* = \phi$ entonces:
B.2.1)	$J^* = \phi$, $K^* \neq \phi \Rightarrow k' = \left\{ k : \max_{k \in K^*} \{g_k^*\} \right\}$, $j' = \left\{ j \in J(k') : \max_{jk' \in JK^*} \{m_j\} \right\}$ $\xi_{j'k'}^* \leftarrow \xi_{j'k'}^* - 1$ e ir a Paso 1
B.2.2)	$J^* \neq \phi$, $K^* = \phi \Rightarrow j' = \left\{ j : \max_{j \in J^*} \{h_j^*\} \right\}$, $k' = \left\{ k \in K(j') : \max_{jk' \in JK^*} \{S_{jk}\} \right\}$ $\xi_{j'k'}^* \leftarrow \xi_{j'k'}^* - 1$ e ir a Paso 1
B.2.3)	$J^* \neq \phi$, $K^* \neq \phi \Rightarrow$ seleccionar $j'k'$ igual que en B.2.1. y B.2.2. $\xi_{j'k'}^* \leftarrow \xi_{j'k'}^* - 1$ e ir a Paso 1

Figura 4.1. Procedimiento PROC_ADM

Una vez terminado PROC_ADM, la asignación de herramientas definida por ξ_{jk}^* será admisible en (2.4) y (2.5). El siguiente paso será determinar un mix agregado de rutas para cada uno de los ítems compatible con la nueva asignación de herramientas a máquinas. Téngase en cuenta que puede que no exista mix agregado de rutas para los ítems compatible con la nueva asignación. Además, en caso de que exista más de uno, nos interesa aquel que da lugar a un menor desequilibrio según (2.1). Esta problemática se modela con el siguiente problema de programación lineal continua (Q):

$$\begin{aligned} & \text{minimizar } u_{\max} - u_{\min} \\ & \text{sujeto a:} \\ & \sum_i \sum_r D_i p_{irjk} \theta_{ir} \leq m_j L_k \xi_{jk}^* \quad \forall k, J(k) \\ & \sum_r \theta_{ir} = 1 \quad \forall i \\ & \sum_i \sum_r \sum_k D_i p_{irjk} \theta_{ir} = m_j u_j \quad \forall j \\ & u_{\min} \leq u_j \leq u_{\max} \quad \forall j \end{aligned} \tag{4.2}$$

$$\theta_{ir} \geq 0 \quad \forall i,r; \quad u_j \geq 0 \quad \forall j; \quad u_{\max} \geq 0; \quad u_{\min} \geq 0$$

donde en las restricciones (4.2) se imponen la compatibilidad con la asignación de herramientas a máquinas fijada. Obsérvese que en el problema (Q) no aparecen las restricciones (2.4) y (2.5), las cuales se verifican una vez que se aplicado anteriormente el procedimiento PROC_ADM. Por otro lado, nótese que el problema (Q) es un problema lineal con todas las variables continuas, por lo que será posible resolverlo sin excesivos requerimientos de tiempo. En caso de que (Q) presente solución admisible, la solución admisible para el problema de carga original será la formada por el mix agregado de rutas obtenido de (Q), θ_{ir}^* , más la asignación de herramientas a máquinas obtenida del procedimiento PROC_ADM, ξ_{jk}^* .

4.2.2 HEURÍSTICA 1.

En la heurística 1 se evalúa la asignación de herramientas a máquinas (ξ_{jk}) como una combinación lineal de las variables θ_{ir} de la forma:

$$\xi_{jk} = \sum_i \sum_r \left[\frac{D_i p_{irjk}}{m_j L_k} \right] \theta_{ir} \quad \forall k, J(k) \quad (4.3)$$

Sustituyendo en el problema original (2.1)-(2.9) las variables enteras (ξ_{jk}) por la expresión (4.3), resulta el siguiente problema lineal continuo (R):

$$\text{minimizar } u_{\max} - u_{\min}$$

sujeto a:

$$\sum_{j \in J(k)} m_j \left(\sum_i \sum_r \left[\frac{D_i p_{irjk}}{m_j L_k} \right] \theta_{ir} \right) \leq A_k \quad \forall k \quad (4.4)$$

$$\sum_{k \in K(j)} S_{jk} \left(\sum_i \sum_r \left[\frac{D_i p_{irjk}}{m_j L_k} \right] \theta_{ir} \right) \leq C_j \quad \forall j \quad (4.5)$$

$$\sum_r \theta_{ir} = 1 \quad \forall i$$

$$\sum_i \sum_r \sum_k D_i p_{irjk} \theta_{ir} = m_j u_j \quad \forall j$$

$$u_{\min} \leq u_j \leq u_{\max} \quad \forall j$$

$$\theta_{ir} \geq 0 \quad \forall i,r; \quad u_j \geq 0 \quad \forall j; \quad u_{\max} \geq 0; \quad u_{\min} \geq 0$$

En la heurística 1 se fija el mix agregado de rutas para cada ítem ($\theta_{ir}^* \quad \forall i,r$) a la solución óptima del problema (R). Téngase en cuenta que en (Q), las restricciones (2.3) se imponen implícitamente ya que (4.2) es más restrictiva que las restricciones (2.3) del problema original.

Si el problema (R) es admisible, se entra en los pasos comunes anteriormente descritos. No obstante, debido a que (4.3) es más restrictiva que (2.3), puede ocurrir que aún siendo el problema original admisible, el problema (R) sea inadmissible. Si esto ocurre, el mix agregado de rutas para cada ítem se fija a partir de la solución óptima de la relajación continua del problema original. A continuación se entraría de igual forma en los pasos comunes descritos en el apartado 4.2.1.

4.2.3 HEURÍSTICA 2.

El objetivo de la heurística 2 es la de fijar el mix agregado de rutas para los distintos ítems mediante una búsqueda más exhaustiva de admisibilidad en las restricciones (2.4)-(2.5). Para ello, las variables θ_{ir} son fijadas procurando alejarnos lo máximo posible de la inadmisibilidad de (2.4) y (2.5). En la Figura 4.2 se presenta un pseudocódigo del procedimiento seguido para fijar el mix de rutas para cada ítem.

Paso 1: Ordenación de ítems en orden decreciente de su demanda.

Paso 2: Para cada ítem i (partiendo del de mayor demanda y yendo en orden decreciente de la misma):

Paso 2.1: Para cada ruta r :

Hacer $\theta_{ir} = 1$

Evaluar ξ_{ijk}^* según (40)

Evaluar g_k^* y h_j^*

Evaluar $\max_r = \max(g_k^*, h_j^*)$

Paso 2.2: $r^* = r: \min \{ \max_r \}$

Paso 2.3: $\theta_{ir}^* = \begin{cases} 1 & r = r^* \\ 0 & \text{resto} \end{cases}$

Figura 4.2. Pseudocódigo para el cálculo del mix de rutas en heurística 2

Obsévese que el mix agregado de rutas se fija de una forma secuencial. Para ello, inicialmente se ordenan los ítems en orden decreciente de su demanda (paso 1). A continuación, tomando los ítems de uno en uno según la ordenación del paso 1, se determina para cada ítem aquella ruta que causa menor deterioro en la inadmisibilidad de la múltiples restricciones de la mochila (2.4)-(2.5). Para el cálculo de estas inadmisibilidades se tienen en cuenta tanto los ítems para los que se ha fijado su ruta de procesado como el que estamos analizando. La ordenación de los ítems en orden decreciente de su demanda se realiza con la idea de fijar primero las rutas de los ítems más demandados, esto es, los que requerirán mayor número de herramientas en las máquinas. Una vez terminado este proceso se tiene fijado un mix agregado de rutas para los ítems (θ_{ir}^*), con el que se entraría en los pasos comunes descritos en el apartado 4.2.1.

4.3 EXPERIMENTACIÓN.

En este apartado nos planteamos evaluar los dos procedimientos heurísticos diseñados atendiendo a dos factores: calidad de las soluciones ofrecidas y eficiencia computacional. Para ello, las soluciones heurísticas se comparan con las ofrecidas por

métodos estándares de ramificación y acotación. Además, se pretende caracterizar los problemas resolubles por cada uno de las heurísticas, atendiendo a alguna medida de la dificultad de los mismos.

La experimentación se ha dividido en dos fases. En cada una de las fases se aborda la resolución de 81 problemas de carga generados aleatoriamente. La diferencia entre los problemas de la primera fase y los de la segunda es su dificultad. Así, mientras que en la primera fase se aborda la resolución de 81 problemas de baja dificultad, en la segunda se abordará un conjunto de problemas de mayor dificultad. Básicamente, la dificultad de los problemas de ambas fases se ha conseguido a partir de una mayor restricción de los recursos de producción (copias de herramientas, vidas útiles, etc.). En los Anexos I-1 y I-2 puede encontrarse una completa descripción del diseño de la generación de problemas de ambas fases.

4.3.1 PRIMERA FASE.

En primer lugar, cada uno de los 81 problemas de la primera fase ha sido resuelto óptimamente usando el optimizador XA Callable Library (1994). A continuación, los dos procedimientos heurísticos se aplican a los 81 problemas. Los experimentos han sido realizados en un PC-Pentium a 100 Mhz. bajo Windows 95. La información recogida para cada uno de los problemas es la siguiente:

Referido al problema:

PROBLEMA: Identificación del problema.

VC-VE-RES: Números de variables continuas, enteras y restricciones del problema.

Referido a la solución óptima:

LP: Función objetivo óptima de la relajación continua del problema.

MILP: Función objetivo óptima del problema original.

GAP (%): Gap (expresado en porcentaje) entre las "MILP" y "LP" medido sobre "LP".

CPU MILP: Tiempo de computación (horas: minutos: segundos) para la obtención del óptimo entero.

ITER: Número de iteraciones del optimizador XA para encontrar el óptimo.

Referido a la heurística 1:

- H1:** Función objetivo para la solución ofrecida por la heurística 1 ("NA" cuando el procedimiento no ofrece solución admisible).
- DIF H1 (%):** Desviación (expresada en porcentaje) de "H1" respecto a "MILP" medido respecto a "MILP".
- CPU H1:** Tiempo de computación (expresado en segundos) requerido por la heurística 1.

Referido a la heurística 2:

- H2:** Función objetivo para la solución ofrecida por la heurística 2 ("NA" cuando el procedimiento no ofrece solución admisible).
- DIF H2 (%):** Desviación (expresada en porcentaje) de "H2" respecto a "MILP" medido respecto a "MILP".
- CPU H2:** Tiempo de computación (expresado en segundos) requerido por la heurística 2.

4.3.1.1 RESULTADOS.

En la Tabla 4.2 se muestran los resultados correspondientes a cada uno de los 81 problemas de la fase 1.

Tabla 4.2. Resultados de experimentación. Fase 1

PROBLEMA		SOLUCIÓN ÓPTIMA					HEURÍSTICA 1			HEURÍSTICA 2		
PROBLEMA	VC-VE-RES	LP	MILP	GAP[%]	CPU MILP	ITER	HI	DIF H1 [%]	CPU H1	H2	DIF H2 [%]	CPU H2
0000	127-48-124	144.543	144.543	0.00	0:01:10	23023	144.543	0.00	0.27	144.543	0.00	0.6
0001	85-45-101	0	0	0.00	0:01:02	21008	0	0.00	0.17	263.024	263.02	0.17
0002	168-53-137	0	0	0.00	0:04:11	67585	0	0.00	0.4	26.411	26.41	1.48
0010	132-48-124	0	0	0.00	0:00:48	14279	0	0.00	0.33	0	0.00	0.72
0011	138-60-136	0	0	0.00	0:02:00	37371	0	0.00	0.28	0	0.00	0.72
0012	142-35-109	0	0	0.00	0:00:14	4621	0	0.00	0.22	0	0.00	0.78
0020	104-24-72	0	0	0.00	0:00:31	10837	0	0.00	0.1	0	0.00	0.33
0021	102-30-78	0	0	0.00	0:00:03	1945	0	0.00	0.12	0	0.00	0.32
0022	89-53-109	2487.667	3169.792	26.36	0:00:10	3550	NA	-	-	NA	-	-
0100	108-36-110	466.099	466.099	0.00	0:00:21	7785	466.099	0.00	0.11	466.099	0.00	0.34
0101	104-60-134	2223	2223	0.00	0:02:25	44929	2223	0.00	0.17	2223	0.00	0.27
0102	192-71-165	2916.513	2916.513	0.00	0:16:15	235791	2916.513	0.00	0.5	2916.513	0.00	1.97
0110	145-25-99	1275.888	1275.888	0.00	0:00:14	5392	1275.888	0.00	0.28	1275.888	0.00	0.78
0111	109-30-86	1479.61	1479.61	0.00	0:00:07	2595	1479.61	0.00	0.22	1479.61	0.00	0.45
0112	95-53-101	0	0	0.00	0:00:03	1210	0	0.00	0.11	0	0.00	0.17
0120	79-36-92	833.408	833.408	0.00	0:00:11	5570	833.408	0.00	0.05	833.941	0.06	0.16
0121	159-45-129	930.783	930.783	0.00	0:00:51	15204	930.783	0.00	0.5	930.783	0.00	1.21
0122	168-71-157	0	0	0.00	0:00:59	15630	0	0.00	0.22	0	0.00	1.16
0200	134-48-116	0	0	0.00	0:00:06	1969	0	0.00	0.12	0	0.00	0.7
0201	111-60-134	55.513	55.513	0.00	0:07:29	143819	55.513	0.00	0.18	55.513	0.00	0.33
0202	74-35-73	1519	1519	0.00	0:00:17	8249	1519	0.00	0.06	1695.72	10.92	0.1
0210	146-36-112	512.965	512.965	0.00	0:00:21	7448	512.965	0.00	0.38	512.965	0.00	0.88
0211	103-60-126	2734.442	2734.442	0.00	0:03:47	83867	2734.442	0.00	0.16	2734.442	0.00	0.23
0212	88-53-109	161.611	161.611	0.00	0:01:44	35974	161.611	0.00	0.17	280.042	45.27	0.11
0220	86-48-114	0	0	0.00	0:00:37	12893	0	0.00	0.1	0	0.00	0.11
0221	89-45-101	701.064	701.064	0.00	0:02:41	63147	701.064	0.00	0.1	1959.83	157.14	0.22

PROBLEMA	SOLUCIÓN ÓPTIMA						HEURÍSTICA 1			HEURÍSTICA 2			
	PROBLEMA	VC-VE-RES	LP	MILP	GAP[%]	CPU MILP	ITER	HI	DIF HI [%]	CPU HI	H2	DIF H2 [%]	CPU H2
0222	119-35-91	0	0	0.00	0.00	0:00:27	10298	0	0.00	0.16	0	0.00	0.54
1000	226-36-120	0	0	0.00	0.00	0:00:58	16271	0	0.00	0.43	0	0.00	2.31
1001	236-60-146	0	0	0.00	0.00	0:00:24	5449	0	0.00	0.15	0	0.00	3.3
1002	207-53-111	0	0	0.00	0.00	0:00:12	3573	0	0.00	0.21	0	0.00	1.98
1010	224-36-104	1970.5	1970.5	0.00	0.00	0:01:35	32073	1970.5	0.00	0.38	1970.5	0.00	2.63
1011	176-60-136	0	0	0.00	0.00	0:04:15	80040	0	0.00	0.37	0	0.00	1.48
1012	124-53-101	0	0	0.00	0.00	0:00:07	2488	0	0.00	0.17	0	0.00	0.49
1020	148-24-72	0	0	0.00	0.00	0:00:04	1738	0	0.00	0.17	0	0.00	0.71
1021	128-60-134	0	0	0.00	0.00	0:10:27	190035	0	0.00	0.27	0	0.00	0.6
1022	177-35-91	0	0	0.00	0.00	0:00:05	1664	0	0.00	0.44	0	0.00	1.48
1100	164-48-124	209.922	209.922	0.00	0.00	0:04:53	84989	209.922	0.00	0.43	216.862	2.24	1.43
1101	129-60-118	0	0	0.00	0.00	0:00:22	7785	0	0.00	0.22	0	0.00	0.49
1102	117-53-117	967.135	967.135	0.00	0.00	0:04:13	84740	967.135	0.00	0.17	967.135	0.00	0.33
1110	144-25-89	0	0	0.00	0.00	0:00:11	3366	0	0.00	0.28	0	0.00	0.87
1111	173-30-86	0	0	0.00	0.00	0:00:32	10304	0	0.00	0.28	0	0.00	1.05
1112	128-71-129	0	0	0.00	0.00	0:03:34	76762	0	0.00	0.17	0	0.00	0.39
1120	209-24-82	0	0	0.00	0.00	0:00:37	10182	0	0.00	0.39	0	0.00	1.81
1121	236-45-129	3770.147	3770.147	0.00	0.00	0:03:34	56440	3770.15	0.00	0.44	3809.3	1.01	3.18
1122	247-71-157	0	0	0.00	0.00	0:01:42	24398	0	0.00	0.18	0	0.00	3.69
1200	86-25-79	4082.5	4082.5	0.00	0.00	0:00:09	4366	4082.5	0.00	0.1	4082.5	0.00	0.16
1201	115-45-101	1515.012	1515.012	0.00	0.00	0:04:44	98252	1515.012	0.00	0.18	1575.86	3.77	0.5
1202	158-53-127	864.321	864.321	0.00	0.00	0:03:57	67506	864.321	0.00	0.37	864.321	0.00	1.09
1210	190-48-124	0	0	0.00	0.00	0:00:32	8983	0	0.00	0.38	0	0.00	1.65
1211	253-60-146	288.282	288.282	0.00	0.00	0:14:23	219626	288.282	0.00	0.14	288.282	0.00	4.23
1212	229-53-129	0	0	0.00	0.00	0:00:57	14919	0	0.00	0.39	0	0.00	3.13
1220	79-26-80	329.868	329.868	0.00	0.00	0:00:10	4391	329.868	0.00	0.12	329.868	0.00	0.16
1221	170-45-119	2155.342	2155.342	0.00	0.00	0:06:00	101340	2155.342	0.00	0.33	2155.342	0.00	1.31

PROBLEMA		SOLUCIÓN ÓPTIMA						HEURÍSTICA 1			HEURÍSTICA 2		
PROBLEMA	VC-VE-RES	LP	MILP	GAPI(%)	CPU MILP	ITER	H1	DIF H1 (%)	CPU H1	H2	DIF H2 (%)	CPU H2	
1222	128-71-137	0	0	0.00	0:04:59	96434	0	0.00	0.16	0	0.00	0.49	
2000	146-48-106	0	0	0.00	0:01:36	35975	0	0.00	0.22	0	0.00	0.83	
2001	228-60-128	1064	1064	0.00	0:05:43	115409	1064	0.00	0.5	1064	0.00	2.79	
2002	185-35-83	0	0	0.00	0:00:04	1832	0	0.00	0.22	0	0.00	1.42	
2010	192-24-72	0	0	0.00	0:00:02	692	0	0.00	0.23	0	0.00	1.58	
2011	283-60-138	0	0	0.00	0:00:24	5501	0	0.00	0.12	0	0.00	5.38	
2012	224-35-83	0	0	0.00	0:00:08	3045	0	0.00	0.27	0	0.00	2.26	
2020	123-24-62	0	0	0.00	0:00:02	975	0	0.00	0.16	0	0.00	0.45	
2021	132-30-68	0	0	0.00	0:00:02	860	0	0.00	0.12	0	0.00	0.54	
2022	295-71-165	0	0	0.00	0:10:17	118806	0	0.00	0.19	0	0.00	7.25	
2100	135-48-122	99.6	99.6	0.00	0:02:51	59257	99.6	0.00	0.29	99.6	0.00	0.55	
2101	164-60-134	0	0	0.00	0:15:45	274249	0	0.00	0.32	0	0.00	1.27	
2102	252-71-155	0	0	0.00	0:08:43	109949	0	0.00	0.17	0	0.00	3.68	
2110	179-36-102	0	0	0.00	0:00:26	8637	0	0.00	0.33	0	0.00	1.55	
2111	103-30-68	0	0	0.00	0:00:02	838	0	0.00	0.1	0	0.00	0.28	
2112	268-71-139	0	0	0.00	0:03:20	57454	0	0.00	0.2	0	0.00	4.39	
2120	309-36-104	0	0	0.00	0:00:09	2711	0	0.00	0.1	0	0.00	4.56	
2121	305-60-138	0	0	0.00	0:01:02	17169	0	0.00	0.11	0	0.00	6.75	
2122	151-71-129	5901.333	5901.333	0.00	0:08:21	199029	5901.333	0.00	0.16	6245	5.73	0.72	
2200	115-25-79	0	0	0.00	0:00:12	4734	0	0.00	0.21	0	0.00	0.44	
2201	303-45-121	542.364	542.364	0.00	0:04:24	66863	542.364	0.00	0.13	542.364	0.00	5.06	
2202	275-53-121	0	0	0.00	0:00:15	3908	0	0.00	0.43	0	0.00	4.67	
2210	181-24-72	0	0	0.00	0:00:03	1081	0	0.00	0.28	0	0.00	1.09	
2211	322-60-138	0	0	0.00	0:00:20	4345	0	0.00	0.12	0	0.00	7.36	
2212	131-53-101	0	0	0.00	0:00:40	15510	0	0.00	0.16	0	0.00	0.56	
2220	137-48-122	0	0	0.00	0:04:05	73786	0	0.00	0.34	0	0.00	0.77	
2221	205-30-78	0	0	0.00	0:00:18	6498	0	0.00	0.32	0	0.00	1.76	
2222	131-35-73	0	0	0.00	0:00:22	9603	0	0.00	0.16	0	0.00	0.61	

4.3.1.2 DISCUSIÓN.

A partir de los resultados mostrados en la Tabla 4.2 se pueden hacer los siguientes comentarios:

1. Todos los problemas son resueltos por ambos procedimientos excepto el "0022". Obsérvese que este problema es el único que presenta GAP estrictamente positivo.
2. La heurística 1 encuentra la solución óptima en los 80 problemas resueltos.
3. La heurística 2 da peores resultados que la heurística 1 en cuanto a calidad. La desviación promedio de las soluciones ofrecidas por la heurística 2 respecto a la heurística 1 es igual al 6.44%.
4. En cuanto a tiempos de computación, se puede decir que los requeridos por ambos procedimientos heurísticos son despreciables frente a los necesarios para encontrar los óptimos enteros. Además, en promedio la heurística 2 es más lenta que la heurística 1.

4.3.2 SEGUNDA FASE.

El proceso seguido en esta segunda fase ha sido paralelo al de la primera salvo en lo referido a la obtención de las soluciones óptimas. Debido al elevado tiempo de computación requerido en la resolución óptima de los problemas, la estrategia por la que se ha optado es la de la resolución ϵ -óptima (Garfinkel, 1979). El parámetro ϵ se ha fijado a $\epsilon=1\%$. Como consecuencia, la solución ϵ -óptima se aleja a lo sumo un 1% del óptimo absoluto. Para cada uno de los 81 problemas se ha recogido la misma información que en el caso de la primera fase salvo en lo referido a la solución óptima. En este caso los resultados son ϵ -óptimos.

4.3.2.1 RESULTADOS.

En la Tabla 4.3 se muestran los resultados correspondientes a cada uno de los 81 problemas de la fase 2.

Tabla 4.3. Resultados de experimentación. Fase 2

PROBLEMA		SOLUCIÓN ε-ÓPTIMA (ε=1%)					HEURÍSTICA 1			HEURÍSTICA 2		
PROBLEMA	VC-VE-RES	LP	MILP	GAP [%]	CPU MILP	ITER	H1	DIF H1 [%]	CPU H1	H2	DIF H2 [%]	CPU H2
0000	35-150-209	0	0	0.00	1:02:37	490616	0	0	0.17	2421.036	2421.04	0.28
0001	52-180-249	4728.567	4749.861	0.44	0:23:26	167903	4728.5667	-0.44	0.27	6660.764	39.40	1.04
0002	68-80-136	1140.068	1152.579	1.01	0:07:56	87643	1140.0683	-1.00	0.22	1453.276	24.01	0.94
0010	49-180-249	6348.052	6396.268	0.75	0:34:18	255768	6348.052	-0.74	0.27	9245.34	43.86	1.04
0011	34-60-108	1817	2004.49	9.78	0:00:44	12164	NA	-	-	NA	-	-
0012	49-180-249	8196.105	8263.516	0.81	0:37:40	242042	8196.105	-0.81	0.22	10474.719	26.44	1.25
0020	40-50-98	797.833	886.992	9.93	0:01:18	22139	NA	-	-	1547.947	66.97	0.22
0021	40-50-98	876.833	880.342	0.36	0:01:33	16367	876.833	-0.36	0.1	1271.442	39.89	0.17
0022	48-60-113	0	0	0.00	0:00:14	1685	0	0	0.11	250.752	250.75	0.38
0100	49-80-131	0	0	0.00	0:10:16	126483	NA	-	-	0	0	0.33
0101	68-80-136	0	0	0.00	0:25:25	203619	0	0	0.44	1856.126	1856.13	0.82
0102	49-80-131	0	0	0.00	0:15:00	153461	NA	-	-	963.531	963.53	0.44
0110	63-60-118	0	0	0.00	0:00:21	4679	0	0	0.33	0	0	0.6
0111	49-100-156	0	0	0.00	0:08:10	78554	NA	-	-	1026.014	1026.01	0.61
0112	49-100-156	0	0	0.00	0:12:07	96641	NA	-	-	1198.671	1198.67	0.49
0120	60-60118	0	0	0.00	0:00:47	7445	0	0	0.38	0	0	0.6
0121	47-60-113	3363.5	4037.672	19.47	0:47:10	640931	NA	-	-	NA	-	-
0122	33-40-78	0	0	0.00	0:01:59	32043	NA	-	-	217.602	217.6	0.17
0200	52-50-98	1024.273	3645.205	233.12	0:03:48	44334	NA	-	-	NA	-	-
0201	50-60-113	0	0	0.00	0:00:10	1067	NA	-	-	NA	-	-
0202	52-50-98	3873.376	6945.357	77.31	0:20:52	250312	NA	-	-	NA	-	-
0210	66-40-88	0	0	0.00	0:00:11	2340	NA	-	-	NA	-	-
0211	66-40-88	0	0	0.00	0:00:06	831	NA	-	-	NA	-	-
0212	32-40-78	0	525.476	525.48	0:00:49	10086	NA	-	-	NA	-	-
0220	73-40-88	0	0	0.00	0:00:05	450	0	0	0.38	1920.562	1920.56	0.55
0221	62-50-103	0	0	0.00	0:00:22	3633	NA	-	-	NA	-	-

PROBLEMA	SOLUCIÓN ϵ -ÓPTIMA ($\epsilon=1\%$)						HEURÍSTICA 1			HEURÍSTICA 2		
	PROBLEMA	VC-VE-RES	LP	MILP	GAP [%]	CPU MILP	ITER	H1	DIF H1 [%]	CPU H1	H2	DIF H2 [%]
0222	46-50-98	8252.771	9330.796	12.91	0:01:18	15013	NA	-	-	NA	-	-
1000	55-120-176	1023.5	1069.839	4.12	1:00:57	683362	1383.77	26.84	0.38	2427.337	126.04	0.76
1001	79-50-98	0	0	0.00	0:00:25	5095	0	0	0.38	0	0	0.67
1002	61-150-209	2254.849	2260.202	0.23	0:43:54	350201	2254.849	-0.23	0.28	2932.24	28.47	0.93
1010	107-180-254	3328.607	3339	0.30	2:05:28	719530	3328.607	-0.3	0.77	6957.89	105.23	4.66
1011	100-150-219	4436.055	4465.881	0.66	0:59:25	358941	4436.055	-0.65	0.54	6636.7	47.54	4.01
1012	51-120-174	3866.851	3897.773	0.78	0:15:26	132443	3866.851	-0.77	0.16	6720.069	70.60	0.61
1020	88-40-83	0	0	0.00	0:00:23	7739	0	0	0.33	0	0	0.6
1021	89-50-103	0	0	0.00	0:01:15	12869	0	0	0.17	0	0	0.94
1022	100-60-118	0	0	0.00	0:00:26	3034	0	0	0.66	0	0	1.15
1100	40-40-78	0	0	0.00	0:00:19	5660	447.671	447.67	0.23	223.503	223.50	0.17
1101	78-80-131	3983.283	4032.411	1.20	0:09:23	95533	4007.072	-0.61	0.71	7716.222	89.14	1.11
1102	83-120-179	9180.222	9253.516	0.79	0:13:47	92992	9180.222	-0.78	0.44	13610.488	46.58	1.92
1110	78-100-156	4152.762	4176.688	0.56	0:24:53	230578	4638.895	10.81	1.1	7199.764	70.69	1.37
1111	91-100-156	0	0	0.00	0:17:25	145804	NA	-	-	0	0	1.76
1112	131-80-136	3675.346	3687.672	0.33	0:15:40	109249	3675.346	-0.33	0.55	6933.141	85.69	3.13
1120	96-40-88	0	0	0.00	0:00:04	358	0	0	0.61	0	0	0.77
1121	91-150-219	0	0	0.00	5:15:08	1580230	NA	-	-	NA	-	-
1122	108-80-136	8697.61	8737.807	0.46	0:06:52	58116	8697.67	-0.45	1.21	10448.746	19.36	1.97
1200	83-40-83	0	0	0.00	0:00:05	369	0	0	0.5	0	0	0.66
1201	114-60-118	0	0	0.00	0:00:21	875	NA	-	-	NA	-	-
1202	78-50-103	0	0	0.00	0:00:34	3373	NA	-	-	54.483	54.48	0.88
1210	96-40-88	0	0	0.00	0:00:06	727	NA	-	-	NA	-	-
1211	120-40-88	0	0	0.00	0:00:06	767	NA	-	-	NA	-	-
1212	60-50-93	0	0	0.00	0:00:34	5070	NA	-	-	0	0	0.5
1220	60-50-93	0	0	0.00	0:00:07	953	NA	-	-	NA	-	-
1221	82-50-98	0	0	0.00	0:00:08	648	NA	-	-	NA	-	-

PROBLEMA	SOLUCIÓN ε-ÓPTIMA (ε=1%)						HEURÍSTICA 1			HEURÍSTICA 2			
	PROBLEMA	VC-VE-RES	LP	MILP	GAP (%)	CPU MILP	ITER	H1	DIF H1 (%)	CPU H1	H2	DIF H2 (%)	CPU H2
1222	56-40-78		1331.194	1973.954	44.91	0:00:29	7033	NA	-	-	NA	-	-
2000	149-150-219		0	0	0.00	3:05:35	992702	0	0	1.32	712.029	712.03	8.45
2001	98-120-181		0	0	0.00	0:02:07	18343	0	0	0.49	0	0	1.77
2002	123-80-131		0	0	0.00	0:42:42	321857	0	0	0.49	0	0	2.42
2010	111-40-83		0	0	0.00	0:00:03	878	0	0	0.16	0	0	0.77
2011	114-60-113		0	0	0.00	0:00:21	2528	0	0	0.44	0	0	1.86
2012	110-60-113		0	0	0.00	0:00:13	2761	0	0	0.5	0	0	1.22
2020	148-50-103		728.5	731.833	0.40	0:04:30	56316	728.5	-0.4	0.5	1829.546	131.96	2.19
2021	121-40-88		0	0	0.00	0:00:04	360	0	0	0.28	0	0	1.2
2022	94-40-83		1651	1654.114	0.19	0:02:13	29876	1651	-0.18	0.22	2715.658	60.52	0.87
2100	67-50-93		0	0	0.00	0:00:30	7044	NA	-	-	NA	-	-
2101	108-50-98		0	0	0.00	0:00:07	734	0	0	0.65	0	0	1.11
2102	81-80-126		0	0	0.00	0:08:48	63827	NA	-	-	419.419	419.42	1.59
2110	118-50-98		0	0	0.00	0:00:09	960	0	0	0.87	0	0	1.38
2111	118-50-98		743	1098.677	42.19	0:13:32	180577	1185.952	7.28	0.65	2531.255	119.51	1.21
2112	96-50-98		0	0	0.00	0:00:08	551	0	0	0.66	1441.782	1441.78	1.21
2120	117-150-219		0	0	0.00	7:25:06	2406928	NA	-	-	1695.709	1695.71	4.79
2121	132-50-103		5521.241	5613.926	1.65	0:02:11	27749	5632.363	0.32	1.1	8520.17	50.86	1.64
2122	102-100-156		3530.027	3563.964	0.93	0:33:24	275458	4548.821	26.88	1.49	NA	-	-
2200	154-60-118		859.319	2236.481	143.56	0:16:49	148400	NA	-	-	NA	-	-
2201	168-50-103		0	0	0.00	0:00:11	549	0	0	1.32	55.867	55.87	3.23
2202	93-50-98		3944.776	5193.304	30.87	0:05:20	61532	NA	-	-	NA	-	-
2210	127-40-88		0	0	0.00	0:00:06	483	NA	-	-	0	0	1.27
2211	128-60-118		0	82.96	82.96	0:01:19	12518	NA	-	-	NA	-	-
2212	129-60-118		10843.328	12881.035	18.62	0:18:50	172226	NA	-	-	NA	-	-
2220	102-60-113		0	0	0.00	0:00:21	2959	NA	-	-	NA	-	-
2221	92-40-83		2390.884	3219.76	33.28	0:00:15	3471	NA	-	-	NA	-	-
2222	91-100-156		0	0	0.00	0:54:47	339865	NA	-	-	NA	-	-

4.3.2.2 DISCUSIÓN.

Para analizar los resultados correspondientes a la fase 2, en la Tabla 4.4 se agrupan los 81 problemas según sean resueltos o no por cada uno de los procedimientos heurísticos. Para cada uno de los grupos se obtiene el GAP promedio. En el caso de los problemas resolubles se presenta la desviación promedio (DIF) respecto a las soluciones ϵ -óptimas.

Tabla 4.4. Resumen de resultados. Fase 2

	Resueltos			No Resueltos	
	Nº	GAP(%)	DIF (%)	Nº	GAP (%)
H1	44	1.32	11.63	37	33.57
H2	55	1.22	286.18	26	47.43

Obsérvese que la heurística 1 resuelve 44 problemas de los 81. El GAP promedio de éstos problemas es de 1.32%. En promedio las soluciones ofrecidas por la heurística 1 se alejan un 11.63% de las ϵ -óptimas. Los 37 problemas para los que la heurística 1 no ofrece solución admisible presentan un GAP promedio de 33.57%. En cuanto a la heurística 2 podemos decir que resuelve 55 de los 81 problemas los cuales presentan un GAP promedio de 1.22%. Las soluciones ofrecidas se alejan un 286.18% de las ϵ -óptimas. Por el contrario, la heurística 2 no ofrece solución para 26 problemas. El GAP promedio de estos problemas no resolubles es del 47.43%.

Tres comentarios básicos se pueden hacer sobre los resultados:

1. La heurística 2 es más efectiva en encontrar solución admisible.
2. La heurística 1 ofrece mayor calidad de soluciones que la heurística 2.
3. La heurística 2 resuelve problemas más difíciles (medida la dificultad según el GAP) que la 1. Obsérvese que el promedio de GAP de los problemas no resueltos por la heurística 2 es el 47.43% frente al 33.57% de los no resueltos por la heurística 1.

Definiendo cuatro nuevos grupos de problemas según:

- H1 & H2: problemas para los que ambos procedimientos heurísticos ofrecen solución admisible.
- H1 & NA H2: problemas para los que la primera heurística da solución y la segunda no.
- NA H1 & H2: problemas para los que sólo la heurística 2 da solución admisible.
- NA H1 & NA H2: problemas para los que ni la heurística 1 y 2 ofrecen solución admisible.

los 81 problemas pueden ser agrupados según se muestra en la Tabla 4.5. En la Tabla 4.5 se muestra el número de problemas que presenta cada uno de los grupos anteriores. Para cada grupo se muestra el GAP (en %) medio. Para los problemas en los que ambas heurísticas ofrecen solución admisible se calcula la desviación de la solución ofrecida por la heurística 2 respecto a la 1 medida respecto a la solución ofrecida por la 1.

Tabla 4.5. Clasificación de problemas. Fase 2

	Nº de problemas	GAP (%) medio	DIF H2-H1 (%)
H1 & H2	43	1.33	227.16
H1 & NA H2	1	0.93	-
NA H1 & H2	12	0.83	-
NA H1 & NA H2	25	49.29	-

De la Tabla 4.5 se puede comentar lo siguiente:

1. Para los problemas que son resueltos por ambos procedimientos heurísticos, la heurística 1 da una mayor calidad en cuanto a cercanía al ϵ -óptimo.
2. Por lo que respecta a los problemas para los que ninguna de las dos heurísticas proporciona solución admisible, obsérvese que presentan de media un GAP (en %) bastante superior al del resto de los grupos. Consecuentemente, podemos decir que el GAP es una buena medida tanto de la dificultad de los problemas como de la posibilidad de que las heurísticas ofrezcan solución admisible.
3. Comparando las desviaciones entre las soluciones ofrecidas por ambas heurísticas al pasar de la 1ª a la 2ª fase de la experimentación, podemos decir que se incrementan fuertemente, de un 6.44 % en la 1ª fase a un 227.16% en la 2ª fase. De esto se puede inducir que la calidad de los resultados ofrecidos por la heurística 2 se ve deteriorada más que los de la 1 al incrementar la dificultad de los problemas.

4.4 RESUMEN.

En este capítulo se han diseñado dos procedimientos heurísticos para el problema de carga modelado en el capítulo 2. Ambos procedimientos involucran la resolución de programas lineales continuos. Las heurísticas han sido evaluadas con dos baterías de problemas de distinta dificultad. Una característica común a ambas heurísticas es el poco tiempo de computación que requieren. La heurística 1 se caracteriza frente a la 2 en la mayor calidad de resultados que consigue. Por el contrario, la 2 se caracteriza por conseguir admisibilidad en más casos que la 1. Los resultados de ambos procedimientos se deterioran al aumentar el GAP del problema a resolver. Además, ambos procedimientos no ofrecen solución para problemas de GAP elevado.

5.1 INTRODUCCIÓN.

En el capítulo 4 se presentaron dos procedimientos heurísticos para el problema de carga. En ambos procedimientos se buscaba un mix agregado de rutas para los ítems, compatible con una asignación de herramientas a máquinas fijada previamente. Tras la experimentación se observó que los dos procedimientos eran incapaces de ofrecer solución para alguno de los problemas de mayor dificultad. La idea que nos planteamos es la del diseño de procedimientos de búsqueda en la vecindad. Esto es, en lugar de fijar sólo una asignación de herramientas a máquinas y buscar un mix agregado de rutas compatible, nos planteamos ir variando paso a paso la asignación de herramientas y, para cada uno de los pasos, buscar admisibilidad en cuanto el mix agregado de rutas. Debido a que no se dispone de método que garantice admisibilidad, la estrategia a diseñar deberá moverse por soluciones inadmisibles.

En este capítulo se diseñan los elementos necesarios para un algoritmo de búsqueda en la vecindad adaptados para la formulación del problema de carga presentada en el capítulo 2. Así, a partir de una codificación entera de soluciones, la evaluación de las mismas se diseña manejando tanto soluciones admisibles como inadmisibles, y está basada en el concepto de oscilación estratégica. Esto es, mediante una modificación en la evaluación de soluciones se introducen mecanismos de oscilación en torno a las fronteras de admisibilidad. Se proponen cuatro métodos para fijar la solución inicial. Dos de ellos corresponden a las ofrecidas por los procedimientos heurísticos diseñados en el capítulo 4. El tercero, se obtiene a partir de la relajación continua del problema original. El cuarto se corresponde con un procedimiento de búsqueda a partir de la solución óptima de la relajación continua. Se definen dos movimientos para movernos entre soluciones. A partir de estos elementos, se diseña un algoritmo basado en conceptos de Búsqueda Tabú (BT) el cual está dotado con mecanismos de diversificación e intensificación. El algoritmo diseñado es evaluado en los 26 problemas. Los resultados inducen a incorporar en la fase de búsqueda de la primera solución admisible de más mecanismo de diversificación. Para ello se diseña una estrategia mixta. En una primera fase se diversifica la búsqueda de una primera solución admisible mediante un algoritmo basado en técnicas de Recocido Simulado (RS). En una segunda fase se mejora la misma a partir del algoritmo de BT.

5.2 BÚSQUEDA EN LA VECINDAD.

Los distintos procedimientos de búsqueda en la vecindad parten de una solución inicial, moviéndose en cada paso de una solución a otra en la vecindad, de acuerdo a unas reglas definidas. El diseño de un procedimiento de búsqueda en la vecindad requiere la definición de: codificación de soluciones, estructura de la vecindad y evaluación de soluciones. Además de estos elementos comunes el procedimiento de búsqueda en la vecindad debe estar dotado las reglas para pasar de una solución a otra. La estrategia más común es la del algoritmo de descenso paso a paso: la nueva solución es la mejor de las vecinas, si ésta mejora la actual. En caso contrario (la mejor de las vecinas es peor que la actual), la búsqueda se pararía, quedándose atrapada en un mínimo local. Otras reglas más evolucionadas (por ejemplo, las basadas en BT y RS) deben permitir empeoramientos momentáneos en las soluciones para así tener la posibilidad de escapar de mínimos locales. Inicialmente se diseñan los elementos básicos de búsqueda en la vecindad. A continuación, éstos son integrados en un algoritmo basado en técnicas de BT.

5.2.1 CODIFICACIÓN DE LAS SOLUCIONES.

Para representar las soluciones del modelo del problema de carga descrito en el capítulo 2, básicamente disponemos de dos opciones: (1) fijar el mix agregado de rutas para cada

ítem $(\theta_{i,r} \forall i,r)$ ó (2) fijar la asignación de herramientas a máquinas $(\xi_{jk} \forall k, J(k))$.

En los procedimientos heurísticos presentados a continuación se ha optado por el segundo tipo de codificación. En esta representación, la estructura de la solución es un vector n-dimensional de enteros. La dimensión del vector (n) viene definida por el número posible de asignaciones, o lo que es lo mismo, la suma de cardinalidades de los conjuntos $J(k)$ (ó $K(j)$) (véase Tabla 2.1):

$$n = \sum_k |J(k)| = \sum_j |K(j)| \tag{5.1}$$

Cada coordenada representará el número de herramientas de un determinado tipo asignado a las máquinas de una estación. El valor de cada coordenada estará en el conjunto: $\{0, \dots, m_{jk}\}$ donde la cota superior para cada variable, m_{jk} , viene determinada por las restricciones (2.4) y (2.5) según:

$$m_{jk} = \min \left\{ \left\lfloor \frac{A_k}{m_j} \right\rfloor, \left\lfloor \frac{C_j}{S_{jk}} \right\rfloor \right\} \tag{5.2}$$

En la Figura 5.1 se representa esquemáticamente el vector correspondiente a una solución (ξ) . Cada variable entera está asociada a cada una de las asignaciones de herramientas a estaciones. El valor de cada posible asignación está acotado superiormente según se ha descrito anteriormente.

Estación-Herramienta	1-1	1-4	3-1	4-4	5-5	...	J-K-1	J-K
Variable	1	2	3	4	5	...	n-1	n
ξ	2	1	0	m_{44}	m_{55}	...	3	2

Figura 5.1. Codificación de soluciones

5.2.2 ESTRUCTURA DE LA VECINDAD.

Dada una solución definida por el vector ξ , las soluciones vecinas de ésta se definen como aquellas a las que se puede llegar tras la realización de un movimiento. Dos tipos de movimientos se han definido:

1. Subir: Seleccionar una coordenada del vector ξ e incrementarla en una unidad.

$$\xi_{jk} \leftarrow \xi_{jk} + 1$$

2. Bajar: Seleccionar una coordenada del vector ξ y decrementarla en una unidad.

$$\xi_{jk} \leftarrow \xi_{jk} - 1$$

Obviamente, el movimiento de bajar no puede ser aplicado a coordenadas que sean iguales a cero ya que darían lugar a una asignación negativa. Por otro lado, el movimiento de subir no puede ser aplicado a variables que están a su cota superior m_{jk} definida según (5.2). Además, téngase en cuenta que la vecindad de una solución dada ξ , no se restringe a soluciones admisibles, esto es, las soluciones vecinas serán a las que se puede llegar tras la realización de un movimiento de los definidos, independientemente del carácter admisible. Un orden de magnitud del tamaño de la vecindad será $2 \cdot n$.

5.2.3 EVALUACIÓN DE SOLUCIONES.

El problema planteado en este apartado es el de asociar a cada solución, definida por un vector ξ , una medida de su bondad. Debido a que la estructura de vecindad de las soluciones no se ha restringido únicamente a soluciones admisibles, tendremos coexistencia de distintos tipos de soluciones (admisibles e inadmisibles). Además, dentro de las soluciones inadmisibles se pueden distinguir dos tipos:

1. Inadmisibles en restricciones tipo de la mochila ((2.4) y (2.5)).
2. Admisibles en restricciones tipo de la mochila, pero inadmisibles para el problema original. Esto es, dada una asignación de herramientas a máquinas que verifica las restricciones (2.4) y (2.5), no existe un mix agregado de rutas para cada ítem compatible con la misma.

La evaluación de las soluciones diseñada está basada en el concepto de oscilación estratégica. Así, la evaluación de soluciones se modifica para introducir mecanismos de oscilación en torno a la frontera de admisibilidad. En la Figura 5.2 se representa un esquema con los distintos tipos de soluciones:

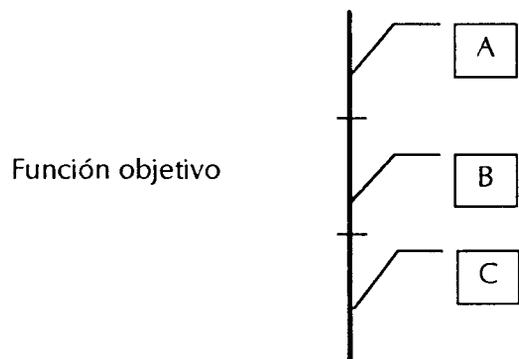


Figura 5.2. Jerarquía entre tipos de soluciones

En la Figura 5.2 se representa en la vertical la función objetivo a asociar a los distintos tipos de soluciones. Moviéndonos de peor a mejor (en la vertical de arriba a abajo) tenemos los siguientes tipos de soluciones:

- A) Inadmisibles en restricciones tipo de la mochila.
- B) Admisibles en restricciones tipo de la mochila pero inadmisibles en problema original.
- C) Admisibles en problema original.

En definitiva, en la evaluación de soluciones a diseñar, todas las soluciones admisibles (C) tendrán menor función objetivo que las inadmisibles (B y A). Además, consideramos que todas las admisibles en restricciones de la mochila pero inadmisibles en el problema original (B), tendrán menor función objetivo que las inadmisibles en las restricciones tipo de la mochila.

Una vez jerarquizados los distintos tipos de soluciones, una cuestión queda por resolver: especificar los valores de cada solución según el tipo al que pertenezca.

Tipo A. Las soluciones inadmisibles en las restricciones tipo de la mochila se valorarán atendiendo a la violación de las mismas. Dada una solución definida por su vector ξ , para cada una de las restricciones de la mochila (2.4) y (2.5) se calcula un índice que mide la violación de la misma:

$$ic_k = \max \left(\frac{\sum_{j \in J(k)} m_j \xi_{jk} - A_k}{\sum_k A_k}, 0 \right) \quad \forall k \tag{5.3}$$

$$ic_j = \max \left(\frac{\sum_{k \in K(j)} s_{jk} \xi_{jk} - C_j}{\sum_j C_j}, 0 \right) \quad \forall j \tag{5.4}$$

Obsérvese que si la restricción es admisible, el índice vale cero. Para las restricciones inadmisibles, el índice es el cociente entre la violación de la restricción y la suma de todos los recursos disponibles de ese tipo.

Llamando z a la función objetivo de una solución definida por ξ , si es inadmisibles en las restricciones tipo de la mochila la calculamos según:

$$z = \left(\sum_k ic_k + \sum_j ic_j \right) + 2 \tag{5.5}$$

Tipo B. Dada una solución ξ , para caracterizar su pertenencia a este tipo (admisibles en restricciones tipo de la mochila pero inadmisibles en problema original) planteamos el siguiente problema lineal continuo (S):

$$\text{minimizar } \sum_k \sum_{j \in J(k)} \omega_{jk}^- \tag{5.6}$$

sujeto a:

$$\sum_i \sum_r D_i p_{irjk} \theta_{ir} + \omega_{jk}^+ - \omega_{jk}^- = m_j L_k \xi_{jk} \quad \forall k, J(k) \tag{5.7}$$

$$\sum_r \theta_{ir} = 1 \quad \forall i$$

$$\theta_{ir} \geq 0 \quad \forall i, r; \omega_{jk}^+, \omega_{jk}^- \geq 0 \quad \forall k, J(k)$$

El segundo término de las restricciones (5.7) es constante al estar fijada la solución ξ . Estas restricciones han sido modificadas introduciendo dos variables (ω) que no se harán simultáneamente básicas. Las variables ω^+ recogen las holguras, mientras que las ω^- absorben las violaciones. Dada una solución ξ admisible en las restricciones tipo de la mochila, una función objetivo positiva del problema (S), indica que la solución pertenece al Tipo B. Además, a mayor valor de la función objetivo peor será la solución.

De igual forma que en las soluciones Tipo A, si denominamos z a la función objetivo, ésta vendrá dada por:

$$z = 2 - \frac{1}{1 + \sum_k \sum_{j \in J(k)} \omega_{jk}^-} \quad (5.8)$$

Obsérvese que para las soluciones pertenecientes al Tipo B, la z estará entre 1 y 2. Dada una solución ξ admisible en las restricciones tipo de la mochila, $z = 1$ cuando la función objetivo de (S) es cero, esto es, cuando la solución es admisible en el problema original.

Tipo C. Para mantener la jerarquía de soluciones mostrada en la Figura 5.2, la función objetivo z de las soluciones admisibles para el problema original, debe ser menor o igual a 1. Además, estas soluciones serán mejores a menor diferencia entre las cargas máxima y mínima. En definitiva, la función objetivo z de estas soluciones viene definida por:

$$z = 1 - \frac{1}{1 + u_{\max} - u_{\min}} \quad (5.9)$$

Obsérvese que una solución admisible para el problema original, que da lugar a una distribución de cargas totalmente equilibrada, presenta función objetivo $z = 0$. Soluciones admisibles en el problema original no totalmente equilibradas dan $0 < z < 1$. Además, a mayor desequilibrio, mayor z .

Dada una solución ξ del Tipo C (admisible para el problema original), el cálculo de $u_{\max} - u_{\min}$ se realiza resolviendo el problema lineal continuo(Q):

$$\begin{aligned} & \text{minimizar } u_{\max} - u_{\min} \\ & \text{sujeto a:} \\ & \sum_i \sum_r D_i p_{irjk} \theta_{ir} \leq m_j L_k \xi_{jk} \quad \forall k, J(k) \\ & \sum_r \theta_{ir} = 1 \quad \forall i \\ & \sum_i \sum_r \sum_k D_i p_{irjk} \theta_{ir} = m_j u_j \quad \forall j \\ & u_{\min} \leq u_j \leq u_{\max} \quad \forall j \\ & \theta_{ir} \geq 0 \quad \forall i, r; u_j \geq 0 \quad \forall j; u_{\max} \geq 0; u_{\min} \geq 0 \end{aligned} \quad (5.10)$$

Obsérvese que el segundo término de las restricciones (5.10) es constante al estar fijada la solución ξ .

A continuación se representa de forma esquemática la valoración de las soluciones según el tipo al que pertenezcan:

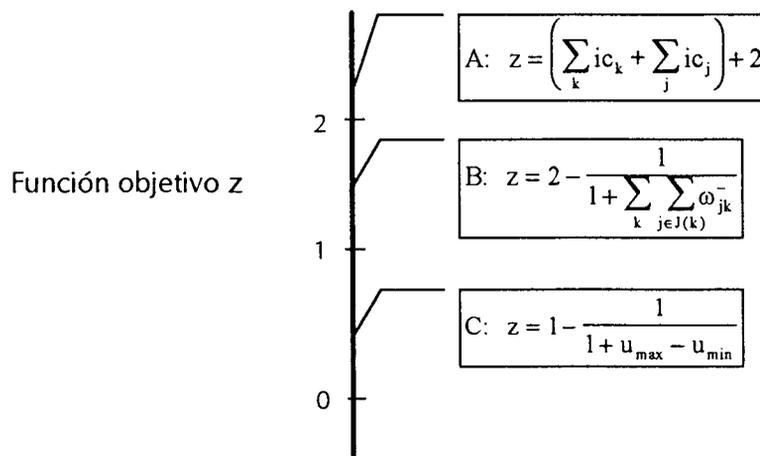


Figura 5.3. Evaluación de soluciones según el Tipo

Dada una solución ξ , para asignar la solución a uno de los tres tipos se comenzaría chequeando primero el Tipo A. Si la solución fuese admisible en todas las restricciones tipo de la mochila ((2.4), (2.5)), se pasa a resolver el problema (S). Únicamente en el caso de que la función objetivo del problema (S) fuese igual a cero, esto es, solución admisible en problema original, se pasaría a resolver el problema (Q) para asignarle la correspondiente función objetivo z.

5.2.4 SOLUCIÓN INICIAL.

Para comenzar el proceso iterativo de búsqueda en la vecindad necesitamos un punto de partida, esto es, un vector de enteros que represente una asignación de herramientas a máquinas del tipo de la representada en la Figura 5.1. Debido a que la evaluación de soluciones diseñada es capaz de valorar cualquier solución, independientemente de su carácter admisible o no admisible, cualquier vector de enteros puede ser el punto de comienzo de un proceso de búsqueda en la vecindad. No obstante, es claro que será tanto mejor aquella solución inicial que se encuentre más cercana de la zona de admisibilidad del problema original (2.1)-(2.9). En esta línea, recuérdese que cada uno de los dos procedimientos heurísticos diseñados en el capítulo 4 definían asignaciones de herramientas a máquinas. Para determinados problemas difíciles, estas asignaciones no eran admisibles para el problema original. No obstante, parece lógico que cualquier procedimiento de búsqueda en la vecindad comience en estos vectores de enteros, antes que en otros generados sin tener en cuenta el problema original (por ejemplo, generados de forma aleatoria). Por otro lado, otra posible estrategia puede ser comenzar a partir de la relajación continua del problema original (2.1)-(2.9). Esto es, resolviendo la relajación continua del problema original y redondeando las asignaciones de herramientas a máquinas al entero más cercano obtenemos otro posible punto de comienzo. Además, a partir de la relajación continua se ha diseñado un procedimiento para imponer la integridad de la asignación de herramientas a máquinas más exhaustivo que el simple redondeo. A continuación se detalla este procedimiento.

5.2.4.1 BÚSQUEDA ALEATORIA SESGADA.

En este método se parte de la solución óptima de la relajación continua del problema original (2.1)-(2.9). A partir de la solución óptima construimos el conjunto:

$$JK^e = \{jk: \xi_{jk} \text{ enteras}\}$$

el cual representa aquellas asignaciones que toman valor entero en la solución óptima de la relajación continua. Es claro que si JK^e contiene a todas las asignaciones del problema, el óptimo continuo coincide con el entero y no hay necesidad de redondeo ni de proceso heurístico de búsqueda en la vecindad. En caso contrario (cuando existen asignaciones fraccionarias), vamos a iniciar un proceso iterativo aleatorio de imposición de integridad a estas variables (véase Figura 5.4). La imposición de integridad estará sesgada hacia el redondeo hacia el entero inmediatamente superior, siempre que sea posible en cuanto a las múltiples restricciones de la mochila (2.4)-(2.5). La idea de este sesgo se apoya en que será tanto mejor para encontrar admisibilidad (en otras palabras, poder procesar los requerimientos de producción con los recursos de herramientas disponibles) disponer del mayor número de herramientas, obviamente sin violar las múltiples restricciones de la mochila (2.4)-(2.5).

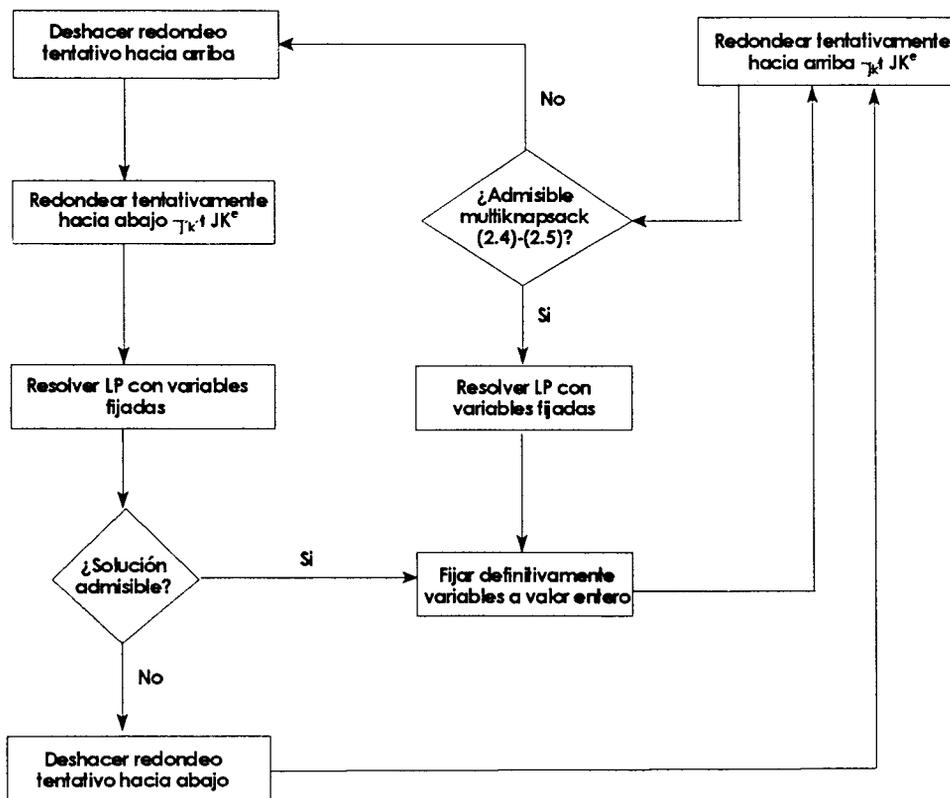


Figura 5.4. Búsqueda aleatoria sesgada

A partir de la resolución de la relajación continua del problema original, y una vez construido el conjunto JK^e , entraríamos en el bloque "Redondeo tentativamente hacia arriba". La asignación a redondear hacia arriba se elegirá de forma aleatoria de entre las fraccionarias, pero teniendo en cuenta las partes fraccionarias. Esto es, pretendemos que una variable tenga más probabilidad de ser elegida para intentar redondearla hacia arriba a mayor parte fraccionaria. Para ello, a cada una de las asignaciones fraccionarias se le asigna una probabilidad de ser seleccionada la cual será proporcional a su parte fraccionaria. Si el redondeo es admisible en (2.4)-(2.5), resolvemos una relajación continua del problema original donde están fijadas a sus valores enteros las asignaciones del

conjunto J_k más la redondeada hacia arriba. A partir de la solución óptima de este problema, se actualiza el conjunto de variables fijadas J_{k+1} con la redondeada, más alguna otra que tome valor entero al resolver el nuevo problema.

En caso de no ser admisible en (2.4)-(2.5) el redondeo hacia arriba de la variable seleccionada, lo deshacemos e intentamos un "redondeo hacia abajo". Así, se selecciona aleatoriamente (asignamos mayor probabilidad a las asignaciones con menor parte fraccionaria) una variable a redondear hacia abajo. Fijando el valor de las asignaciones del conjunto J_k más la que hemos redondeado tentativamente hacia abajo, resolvemos el problema lineal continuo redicido al tener fijadas algunas asignaciones. En caso de admisibilidad, se actualiza el conjunto J_k y entramos de nuevo en el bloque "Redondeo tentativamente hacia arriba". Si el problema es inadmisibile, deshacemos el redondeo que intentabamos hacer hacia abajo y volvemos a intentar un redondeo hacia arriba.

El proceso iterativo finalizará o cuando todas las asignaciones toman valor entero o cuando, para las asignaciones fraccionarias que quedan sin fijar, ni podemos subir ni bajar. Para los problemas difíciles que estamos manejando, la primera circunstancia no ocurre nunca. A partir del vector de asignaciones final de este procedimiento y redondeando al entero más cercano las fraccionarias construimos un vector de enteros a partir del que realizar una búsqueda en la vecindad.

3 BÚSQUEDA TABÚ.

Búsqueda Tabú es una metaheurística introducida por Glover (1986) que puede ser usada como herramienta de resolución de problemas de optimización combinatoria. Una amplia descripción de esta técnica así como de aplicaciones de la misma puede encontrarse en Glover (1990) y Glover y Laguna (1995, 1996).

En su forma más simple BT puede ser acoplada sobre otras heurísticas para evitar que se queden atascadas en óptimos locales. El uso de la palabra tabú tiene el significado cotidiano, esto es, ciertas regiones del espacio de soluciones se encuentran temporalmente inaccesibles por razones históricas, no obstante las restricciones de acceso a esas zonas prohibidas puede ser levantada en determinadas circunstancias. Por otro lado, BT también puede ser visto como un paradigma de búsqueda "inteligente" (Glover y Greenberg 1989), donde la información histórica recogida durante el proceso de búsqueda es guardada con el objetivo de que BT pueda utilizar dicho conocimiento histórico para guiar la búsqueda en direcciones prometedoras. Esto es conseguido utilizando estructuras de memoria flexibles que permitan el almacenamiento y explotación de la información histórica en mejores condiciones que otras estructuras más rígidas (por ejemplo en un algoritmo de ramificación y acotación) o que otros procesos de búsqueda sin memoria (como en RS u otros procesos de búsqueda aleatorios). Estas estructuras de memoria funcionan en distintos horizontes de tiempo, desde el corto al largo plazo, y pueden ser usadas para implementar estrategias para diversificar o intensificar el proceso de búsqueda.

Dentro del campo de la programación lineal entera mixta (como es el caso de la formulación del problema de carga que nos ocupa), BT ha sido aplicada en dos líneas básicas. La primera, incorporándola en procedimientos clásicos de optimización (Aboudi y Jörnsten 1994, Glover y Lokketangen 1996, Lokketangen y Glover 1995 y Lokketangen y Glover 1996). La segunda, introduciéndola como mecanismo de control del proceso de exploración de variables enteras (Laguna 1994, Wen y Huang 1996 y Hindi 1996).

5.3.1 DISEÑO DEL ALGORITMO.

Para diseñar el algoritmo definimos las siguientes estructuras de memoria:

Corto plazo o lista tabú:

$$\text{lista_tabu}[(j,k),l] \quad l = -1,1; \forall k, J(k)$$

(j,k): cada una de las asignaciones de herramientas a máquinas.

l: tipo de movimiento (-1: bajar, 1: subir)

Dada una asignación y un movimiento, la estructura "lista_tabu" almacena el número de iteraciones que está prohibido ese movimiento

Largo plazo:

$$\text{mat_frec}[(j,k),l] \quad l = -1,1; \forall k, J(k)$$

De forma paralela, dada una asignación y un movimiento, la estructura "mat_frec" recoge el número de veces que se ha realizado el movimiento sobre la asignación.

El algoritmo tabú a lo dividimos en dos fases:

Primera fase:

El objetivo de la primera fase es posicionarnos en la primera solución admisible. Para ello dotamos a la búsqueda de mecanismos de diversificación a partir de la estructura "mat_frec". Definiendo el parámetro "iter_sin_mej", como el número de iteraciones sin mejora, la primera fase es una búsqueda local en la que se realiza un movimiento de diversificación si tras "iter_sin_mej" iteraciones consecutivas empeoramos. El movimiento diversificador se escoge en base a las frecuencias de forma que se penalicen los movimientos más comunes. De igual forma, debido que en esta fase no estamos interesados en hacer una búsqueda exhaustiva en el entorno de las soluciones actuales, las restricciones tabú no prohibirán únicamente los últimos movimientos que permitan volver a soluciones ya visitadas. Así, dado cada uno de los movimientos, las soluciones prohibidas serán:

$$\begin{aligned} \text{si } \xi_{jk} &\leftarrow \xi_{jk} + 1 & \text{lista_tabu}[(j,k),-1] &\leftarrow \text{iter} + \text{LT} \quad \forall k \\ \text{si } \xi_{jk} &\leftarrow \xi_{jk} - 1 & \text{lista_tabu}[(j,k),1] &\leftarrow \text{iter} + \text{LT} \quad \forall k \end{aligned}$$

Esto es, si se incrementa la asignación (j,k) en la iteración número "iter", no se prohíbe el decremento de la misma asignación (j,k), sino de todas aquellas que involucren a la misma herramienta. La longitud de la lista tabú (LT) es el parámetro que define el número de iteraciones que permanecerán prohibidas las soluciones.

El estado de prohibido de una determinada solución puede ser levantado si se verifica el criterio de aspiración. En nuestro caso, se ha usado como criterio de aspiración el basado en la mejor función objetivo, es decir, si una solución prohibida da lugar a una solución que es mejora que la mejor hasta el momento se acepta como nueva solución actual. Esta primera fase finalizará cuando se alcance la primera solución admisible.

Segunda fase:

Una vez posicionados en una solución admisible, el objetivo será intensificar la búsqueda en torno a ella. Para ello, se realiza una búsqueda local intensiva donde únicamente utilizamos memoria a corto plazo, donde en las restricciones tabú únicamente prohibimos los opuestos de los movimientos realizados:

$$\begin{array}{ll} \text{si } \xi_{jk} \leftarrow \xi_{jk} + 1 & \text{lista_tabu}[(j,k),-1] \leftarrow \text{iter} + \text{LT} \\ \text{si } \xi_{jk} \leftarrow \xi_{jk} - 1 & \text{lista_tabu}[(j,k),1] \leftarrow \text{iter} + \text{LT} \end{array}$$

Obsérvese que las restricciones tabú de la segunda fase son menos restrictivas que las de la primera. Esto es, en la primera fase, dado un movimiento, se prohibían los inversos que involucrasen a la misma herramienta. En la segunda fase, y debido que estamos interesados en hacer una búsqueda más exhaustiva, se prohíbe únicamente el inverso del movimiento realizado. Como criterio de aspiración se define el mismo que el correspondiente a la primera fase, esto es, el basado en la función objetivo. La búsqueda continuará hasta que se verifique alguna condición de parada como por ejemplo, un número máximo de iteraciones, un tiempo de computación máximo, número máximo de iteraciones sin mejora, etc.

5.3.2 EXPERIMENTACIÓN.

En este apartado nos planteamos evaluar el algoritmo diseñado con los 26 problemas para los que aún no disponemos de método heurístico que proporcione solución admisible. Para cada uno de los problemas, el algoritmo diseñado se ejecutará partiendo de las cuatro posibles soluciones iniciales: H1 (heurística 1), H2 (heurística 2), RC (relajación continua truncando las variables continuas al entero más cercano) y BAS (búsqueda aleatoria sesgada truncando de igual forma las continuas al entero más cercano). Inicialmente, se presentan los parámetros especificados para el algoritmo. A continuación, se muestran los resultados obtenidos, así como las distintas evoluciones del algoritmo para un mismo problema. Por último, se analizan los resultados, extrayendo las conclusiones más importantes de los mismos.

5.3.2.1 PARÁMETROS DEL ALGORITMO.

Los parámetros especificados en el algoritmo son:

- Solución inicial: Las cuatro opciones anteriormente comentadas, esto es, H1, H2, RC y BAS.
- Tamaño de la lista Tabú (LT): Se ha usado un tamaño dinámico de la lista tabú comenzando en $\lceil \sqrt{VE} \rceil$ donde VE es el número de variables enteras del problema. El tamaño de la lista se varía de forma aleatoria entre este número y $2 \times \lceil \sqrt{VE} \rceil$, cada 10 iteraciones.
- Número de iteraciones sin mejora (iter_sin_mej) en la fase 1: Este parámetro se ha especificado a 10.
- Criterio de finalización: para cada problema se finaliza la búsqueda transcurrido un número máximo de iteraciones el cual será función del número de variables enteras del mismo.

5.3.2.2 RESULTADOS.

En la Tabla 5.1 se muestran las soluciones ofrecidas por la heurística Tabú para cada uno de los 26 problemas objeto de esta experimentación. Para cada problema, identificado con su correspondiente número, se indica la solución ϵ -óptima ($\epsilon=1\%$) obtenida en el capítulo 4. Téngase en cuenta que el primero de los problemas (0022) corresponde al primer conjunto de problemas por lo que la solución mostrada es la óptima. Dado un problema, y cada una de las cuatro posibles soluciones iniciales de la heurística, se

presenta la siguiente información:

- **BT¹**: valor de la función objetivo del problema original correspondiente a la primera solución admisible obtenida. Entre paréntesis se indica la iteración del algoritmo en la que se obtiene esta primera solución admisible.
- **BT***: valor de la función objetivo del problema original correspondiente a la mejor solución obtenida durante la evolución del algoritmo. De forma paralela, entre paréntesis se indica la iteración en la que se encuentra dicha solución.

Para una mayor claridad en la exposición de los resultados, se han usado caracteres en negrita y en cursiva en la siguiente forma. Dado un problema, con la negrita se señala el mínimo de las primeras soluciones admisibles. En el caso de que el mismo valor se obtenga para más de un tipo de solución inicial, todos se podrán en negrita. De igual modo, con la negrita denotamos el valor más bajo de iteración para la que se obtiene solución admisible del problema. La cursiva se usa de forma análoga pero en lo referido a las mejores soluciones obtenidas para un problema dado. Por último, para cada problema se presenta la desviación (en %), entre la mejor de las soluciones obtenidas para el mismo y la solución ϵ -óptima. Esta desviación se calcula tomando como referencia la solución ϵ -óptima.

Paralelamente, en la Tabla 5.2 se muestran los tiempos de computación requeridos para la obtención de las primeras y mejores soluciones admisibles del algoritmo. La negrita y la cursiva se usan para denotar los menores tiempos de computación requeridos en obtener la primera y mejor solución respectivamente. En la última columna se muestra para cada problema, el cociente entre los tiempos requeridos en la obtención de la mejor solución para el problema, y la ϵ -óptima.

Tabla 5.1. Soluciones del algoritmo de BT. Sensibilidad respecto a la solución inicial

N°	ε-ÓPTIMA (ε=1%)	H1		H2		RC		BAS		DESV (%) BT*-ε-OPT
		BT*	BT*	BT*	BT*	BT*	BT*	BT*	BT*	
0022	3169.79	3421.45 (18)	3169.79 (48)	3415.01 (16)	3169.79 (45)	3456.17 (18)	3169.79 (47)	3323.11 (12)	3169.79 (42)	0.00
0011	2004.49	2514.55 (29)	2135.80 (68)	2514.55 (28)	2135.80 (67)	2625.65 (29)	2135.80 (69)	2326.22 (21)	2055.20 (58)	2.53
0121	4037.67	5175.02 (42)	4124.29 (50)	4564.21 (25)	4013.00 (41)	5100.26 (27)	4016.12 (45)	4534.36 (18)	4010.00 (34)	-0.69
0200	3645.20	4001.08 (27)	3623.52 (53)	4015.08 (26)	3623.52 (52)	4096.17 (25)	3645.20 (54)	3912.89 (23)	3623.52 (41)	-0.59
0201	0.00	296.60 (20)	0.00 (39)	413.10 (22)	0.00 (40)	505.22 (21)	0.00 (39)	251.32 (18)	0.00 (35)	0.00
0202	6945.35	7704.12 (47)	7006.95 (69)	7625.61 (45)	6982.26 (67)	7691.00 (46)	6901.17 (69)	7552.54 (40)	6921.89 (65)	-0.34
0210	0.00	221.85 (24)	0.00 (42)	221.85 (24)	0.00 (42)	254.17 (26)	0.00 (43)	142.12 (20)	0.00 (35)	0.00
0211	0.00	95.78 (25)	0.00 (44)	95.78 (27)	0.00 (45)	101.10 (27)	0.00 (47)	54.85 (22)	0.00 (45)	0.00
0212	525.47	714.50 (22)	525.47 (56)	705.26 (19)	525.47 (54)	705.26 (21)	525.47 (57)	695.75 (16)	525.47 (48)	0.00
0221	0.00	740.21 (38)	63.21 (62)	621.72 (34)	63.21 (59)	621.72 (35)	63.21 (62)	510.01 (29)	63.21 (52)	63.21
0222	9330.79	10884.56 (27)	9563.02 (67)	10415.09 (25)	9563.02 (61)	10415.09 (28)	9563.02 (63)	10233.65 (19)	9563.02 (55)	2.49
1121	0.00	1325.21 (61)	0.00 (155)	1105.10 (52)	0.00 (144)	1396.95 (55)	0.00 (150)	1001.72 (48)	0.00 (135)	0.00
1201	0.00	605.13 (31)	0.00 (58)	632.21 (32)	0.00 (57)	632.21 (33)	0.00 (59)	632.21 (26)	0.00 (42)	0.00
1210	0.00	742.41 (40)	0.00 (60)	742.41 (39)	0.00 (58)	742.41 (43)	0.00 (64)	742.41 (37)	0.00 (51)	0.00
1211	0.00	401.21 (52)	0.00 (63)	372.21 (49)	0.00 (61)	372.21 (52)	0.00 (65)	365.12 (45)	0.00 (54)	0.00
1220	0.00	642.02 (20)	0.00 (66)	625.63 (18)	0.00 (63)	652.02 (20)	0.00 (66)	625.63 (15)	0.00 (59)	0.00
1221	0.00	122.85 (43)	0.00 (65)	114.20 (40)	0.00 (62)	114.20 (41)	0.00 (63)	114.20 (36)	0.00 (55)	0.00
1222	1973.95	2150.50 (39)	1973.95 (69)	2131.25 (37)	1960.85 (65)	2242.75 (38)	1960.85 (66)	2131.25 (32)	1960.85 (52)	-0.66
2100	0.00	205.69 (23)	0.00 (60)	205.69 (21)	0.00 (59)	205.69 (22)	0.00 (60)	205.69 (20)	0.00 (57)	0.00
2200	2236.48	2825.06 (29)	2293.18 (72)	2614.32 (28)	2293.18 (70)	2754.74 (29)	2293.18 (71)	2525.63 (22)	2272.11 (66)	1.59
2202	5193.30	5789.96 (28)	5193.30 (73)	5745.65 (26)	5193.30 (71)	5796.16 (28)	5193.30 (73)	5745.65 (26)	5193.30 (71)	0.00
2211	82.96	201.89 (20)	115.85 (60)	193.24 (20)	115.85 (58)	193.24 (20)	115.85 (58)	125.21 (17)	101.22 (49)	22.01
2212	12881.03	15312.05 (38)	13050.14 (98)	14827.10 (31)	13050.14 (94)	14915.24 (32)	13050.14 (95)	14789.36 (25)	12905.32 (89)	0.19
2220	0.00	105.08 (19)	0.00 (40)	96.31 (17)	0.00 (36)	96.31 (19)	0.00 (38)	96.31 (16)	0.00 (35)	0.00
2221	3219.76	3695.85 (41)	3219.76 (88)	3621.10 (39)	3219.76 (85)	3621.10 (40)	3219.76 (87)	3512.22 (34)	3219.76 (76)	0.00
2222	0.00	1422.43 (52)	0.00 (149)	1015.21 (40)	0.00 (120)	1314.47 (43)	0.00 (124)	963.54 (33)	0.00 (111)	0.00

Tabla 5.2. Tiempos de computación de BT. Sensibilidad respecto a la solución inicial

Nº	VC-VE-RES	CPU		H1		H2		RC		BAS		REL CPU BT*/ε-OPT
		ε-OPT. (ε=1%)	CPU BT*									
0022	89-53-109	0:00:10	0:01:21	0:06:14	0:00:55	0:05:22	0:01:32	0:07:33	0:00:43	0:04:35	2750.00	
0011	34-60-108	0:00:44	0:03:50	0:08:05	0:03:20	0:07:44	0:04:11	0:08:21	0:02:34	0:05:04	690.91	
0121	47-60-113	0:47:10	0:07:11	0:17:03	0:04:39	0:13:27	0:06:13	0:19:31	0:03:39	0:11:47	24.98	
0200	52-50-98	0:03:48	0:03:55	0:09:42	0:02:55	0:08:21	0:03:59	0:10:12	0:02:04	0:07:12	189.47	
0201	50-60-113	0:00:10	0:02:51	0:07:14	0:02:03	0:06:19	0:01:55	0:05:45	0:01:39	0:04:19	2590.00	
0202	52-50-98	0:20:52	0:06:54	0:12:51	0:06:31	0:11:24	0:07:45	0:12:55	0:05:23	0:09:55	61.90	
0210	66-40-88	0:00:11	0:01:45	0:04:15	0:01:45	0:04:15	0:02:33	0:04:58	0:00:56	0:02:23	1300.00	
0211	66-40-88	0:00:06	0:01:58	0:05:41	0:02:15	0:05:01	0:02:01	0:06:14	0:01:05	0:03:15	3250.00	
0212	32-40-78	0:00:49	0:03:15	0:08:08	0:02:10	0:07:12	0:03:29	0:08:29	0:01:35	0:05:35	683.67	
0221	62-50-103	0:00:22	0:04:36	0:09:19	0:03:56	0:07:07	0:05:01	0:09:28	0:02:14	0:04:07	1122.73	
0222	46-50-98	0:01:18	0:03:48	0:10:35	0:02:59	0:08:24	0:03:54	0:10:05	0:01:56	0:06:01	462.82	
1121	91-150-219	5:15:08	0:28:54	1:06:03	0:21:44	0:58:22	0:24:01	1:09:07	0:19:25	0:45:46	14.52	
1201	114-60-118	0:00:21	0:05:54	0:10:50	0:05:21	0:10:17	0:06:52	0:11:55	0:03:02	0:07:35	2166.67	
1210	96-40-88	0:00:06	0:02:55	0:05:22	0:02:15	0:04:33	0:04:02	0:08:39	0:01:10	0:03:22	3366.67	
1211	120-40-88	0:00:06	0:04:12	0:08:01	0:03:41	0:06:52	0:05:08	0:09:54	0:02:05	0:04:47	4783.33	
1220	60-50-93	0:00:07	0:02:20	0:07:47	0:01:14	0:05:55	0:03:31	0:08:04	0:00:55	0:04:03	3471.43	
1221	82-50-98	0:00:08	0:04:14	0:11:54	0:03:25	0:08:01	0:04:57	0:10:12	0:02:54	0:06:12	4650.00	
1222	56-40-78	0:00:29	0:02:20	0:07:29	0:01:22	0:05:42	0:02:58	0:09:21	0:00:48	0:03:14	668.97	
2100	67-50-93	0:00:30	0:02:14	0:06:59	0:01:05	0:06:02	0:02:21	0:08:14	0:00:56	0:05:32	1106.67	
2200	154-60-118	0:16:49	0:05:45	0:14:13	0:04:41	0:12:55	0:05:56	0:14:04	0:02:04	0:08:21	49.65	
2202	93-50-98	0:05:20	0:03:54	0:09:56	0:02:35	0:06:43	0:04:54	0:09:37	0:02:35	0:06:43	125.94	
2211	128-60-118	0:01:19	0:02:47	0:10:11	0:02:09	0:08:56	0:03:25	0:10:27	0:01:23	0:05:19	403.80	
2212	129-60-118	0:18:50	0:06:15	0:17:54	0:04:01	0:13:21	0:05:45	0:15:32	0:02:25	0:10:32	55.93	
2220	102-60-113	0:00:21	0:02:14	0:05:02	0:01:55	0:03:58	0:03:01	0:05:26	0:01:21	0:03:07	890.48	
2221	92-40-83	0:00:15	0:03:55	0:10:21	0:02:47	0:08:14	0:04:27	0:10:57	0:00:51	0:04:03	1620.00	
2222	91-100-156	0:54:47	0:16:24	0:46:24	0:10:05	0:33:56	0:13:47	0:37:08	0:08:55	0:27:50	50.81	

En las Figuras 5.5 y 5.6, se muestra la evolución de la solución actual con la iteración correspondiente al problema 0121 y cada una de las posibles soluciones iniciales. Para la primera fase, se representa en FO(1) la inadmisibilidad medida según (5.1). Para las soluciones admisibles, se representa en la mitad inferior de las gráficas, la valoración de las mismas según la función objetivo del problema original (2.1).

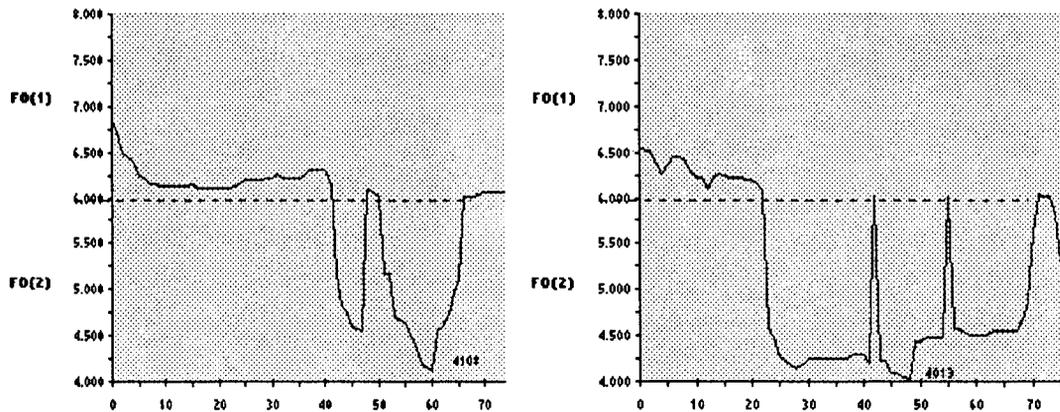


Figura 5.5. Evolución del algoritmo Tabú para H1 y H2. Problema 0121

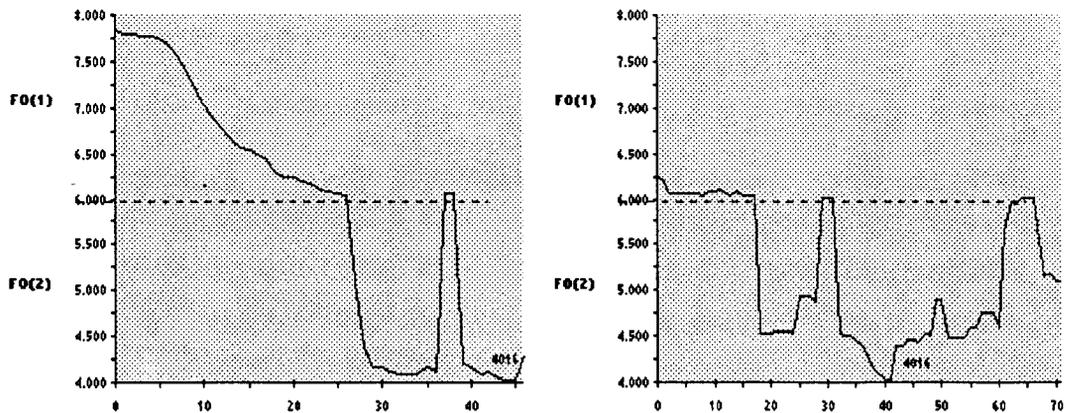


Figura 5.6. Evolución del algoritmo Tabú para RC y BAS. Problema 0121

5.3.2.3 DISCUSIÓN.

A partir de los resultados mostrados en las Tablas 5.1 y 5.2 puede comentarse lo siguiente:

Referido a la obtención de la primera solución inicial.

- Excepto en un problema (1201), la primera solución inicial se obtiene antes (en tiempo de computación o número de iteración) tomando como solución inicial la ofrecida por BAS.

Referido a las mejores soluciones admisibles.

- Partiendo de las soluciones ofrecidas por la BAS, en todos los problemas menos en uno (0202), se ofrece la mejor solución del problema y en un menor tiempo que el requerido al partir de las otras soluciones iniciales.
- Las desviaciones respecto a las soluciones ϵ -óptimas se encuentran en unos rangos bastante buenos.
- Es claro que los tiempos de computación requeridos son globalmente mayores que los de la ϵ -óptima. No obstante, esta afirmación merece una pequeña reflexión. En la mayoría de los problemas, los tiempos de computación de la solución ϵ -óptima son inferiores al minuto. Sin embargo, en el peor de los casos, los procedimientos estándares de ramificación pueden requerir unos tiempos prohibitivos aún para ofrecer una solución ϵ -óptima. Obsérvense por ejemplo los problemas 0121, 0202, 1121, 2200, 2212 y 2222. Aquí, los tiempos requeridos por la mejor solución Tabú son competitivos con los de la ϵ -óptima. En otras palabras, que el algoritmo Tabú diseñado es consistente en cuanto a tiempo de computación para el abanico de problemas considerado, mientras que los procedimientos estándares pueden requerir en el peor de los casos unos tiempos bastante superiores.

Globalmente, a partir de estos resultados se puede decir que, aún sin ofrecer solución admisible para el problema de carga original, los cuatro procedimientos usados como punto de partida en el proceso del algoritmo de BT funcionan de una forma consistente. Además, las soluciones ofrecidas serán tanto mejores cuanto más exhaustivo es el análisis que de las múltiples restricciones de la mochila (2.4)-(2.5) se haga en el método para ofrecer solución inicial de la BT. Recuérdese que tanto en la BAS como en la H2, se hacía un tratamiento diferenciado de estas restricciones.

En lo referido a las evoluciones mostradas en las Figuras 5.5 y 5.6, obsérvense que la estrategia de oscilación en torno a la frontera de admisibilidad permite, una vez alcanzada la región de admisibilidad, retornar de forma controlada a soluciones inadmisibles. Estas oscilaciones dan lugar a exploraciones de distintas regiones, las cuales permiten mejorar las soluciones obtenidas.

5.4 ALGORITMO MIXTO.

En este apartado nos planteamos explorar las soluciones correspondientes a la primera fase siguiendo los mecanismos de evolución entre soluciones correspondientes a RS (Kirkpatrick et al 1983 y Cerny 1985). Los algoritmos basados en RS son procesos de búsqueda en la vecindad, donde de forma controlada se aceptan empeoramientos de las soluciones. Dada una solución, el mecanismo para generar la próxima solución actual será extraer aleatoriamente de la vecindad una solución. Si la solución vecina es mejor que la actual, ésta será aceptada automáticamente. En caso contrario (la vecina extraída es peor), se aceptará con una determinada probabilidad:

$$\text{prob}(\text{iter}) = \exp\left(-\frac{1}{T(\text{iter})} \Delta \text{coste}\right) \quad (5.11)$$

La función de probabilidad depende de dos factores: temperatura (T) y Δcoste (cuánto peor es la solución vecina comparada con la actual). El parámetro temperatura decrece gradualmente con la iteración. Inicialmente se tomará una temperatura T_0 que permita aceptar con cierta probabilidad malos movimientos. Al final, la temperatura se habrá

decrementado de tal forma que la probabilidad de aceptación sea inferior y sólo se acepten movimientos de mejora.

Al contrario que en BT, en RS no se almacena ninguna información de las soluciones visitadas durante la búsqueda. En otras palabras, los algoritmos de RS son "desmemoriados".

5.4.1 DISEÑO DEL ALGORITMO.

De igual forma al algoritmo de BT diseñado, la búsqueda la dividimos en dos fases. La primera de ellas corresponde a la búsqueda de la primera solución admisible. La segunda fase será una búsqueda intensiva para la mejora de la misma siguiendo lo descrito en 5.3.1 (segunda fase). En la primera fase, además del mecanismo de control de empeoramientos basado en la temperatura, introducimos un segundo control. Esto es, dada una solución actual admisible en las restricciones de la mochila (2.4)-(2.5), pero inadmisibile para el problema original (solución tipo B), en ningún caso permitimos movernos a soluciones inadmisibles en las restricciones (2.4)-(2.5).

5.4.2 EXPERIMENTACIÓN.

Paralelamente a la experimentación anterior, en este apartado nos planteamos evaluar el algoritmo diseñado con los 26 problemas. Para cada uno de los problemas, el algoritmo diseñado se ejecutará partiendo de BAS. Inicialmente, se presentan los parámetros especificados para el algoritmo. A continuación, se muestran los resultados obtenidos, así como la evolución del algoritmo para un mismo problema. Por último, se analizan los resultados, extrayendo las conclusiones más importantes de los mismos.

5.4.2.1 PARÁMETROS DEL ALGORITMO.

Los parámetros a especificar son los correspondientes a cada una de las dos fases en los que se ha dividido el proceso de búsqueda. Para la segunda fase, los parámetros usados son los descritos en el apartado 5.3.2.1. Por lo que respecta a la primera fase, tomaremos como solución inicial la ofrecida por BAS. Además, tendremos que especificar los parámetros correspondientes al programa de enfriamiento. En nuestro caso se ha optado por un programa de enfriamiento geométrico definido por:

- T_0 (temperatura inicial): vendrá definida por:

$$T_0 = \frac{\mu}{-\ln p_0} \text{coste}(0) \quad (5.12)$$

donde p_0 es la probabilidad de aceptar inicialmente soluciones peores en un tanto por uno μ del coste de la solución inicial $\text{coste}(0)$. En nuestro caso se ha optado por $p_0 = 0.8$ y $\mu = 0.15$.

- L (número de iteraciones a temperatura constante): 20.
- α (grado de enfriamiento): 0.95.

5.4.2.2 RESULTADOS.

En la Tabla 5.3 se muestran los resultados para el algoritmo mixto diseñado. Para cada problema se presentan tanto de la primera (SOL¹) como la mejor solución admisible (SOL^{*}) obtenidas. Además, se muestran los requerimientos computacionales para la obtención de cada una de ellas. Por último, se comparan tanto las soluciones como los tiempos con sus equivalentes correspondientes a la BT partiendo de la BAS. Posteriormente, se presenta la evolución de algoritmo correspondiente al problema 0121 en la misma forma que en el apartado 5.3.2.2.

En la Figura 5.7 se representa la evolución del algoritmo mixto para el problema 0121. Téngase en cuenta que el concepto de iteración varía entre la primera fase y la segunda. Así, en la primera fase (RS) en cada una de las iteraciones se evalúan soluciones individuales, mientras que en la segunda (BT), la transición entre soluciones se hace tras una búsqueda local (evaluando más de una solución).

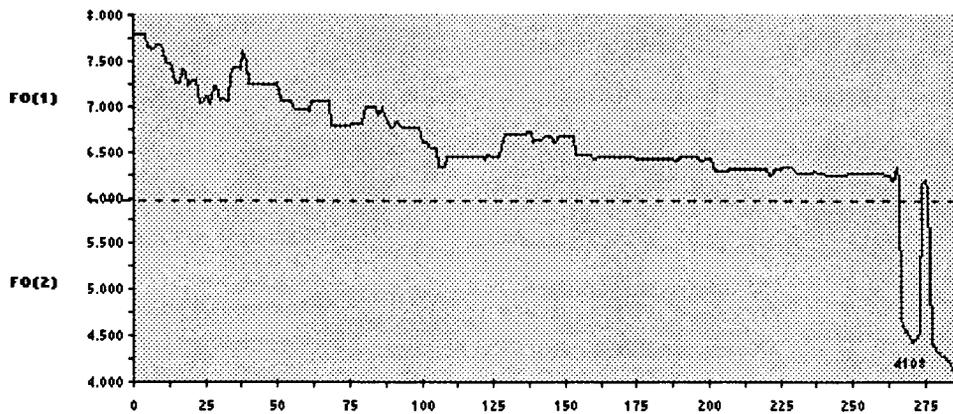


Figura 5.7. Evolución del algoritmo mixto. Problema 0121

Tabla 5.3. Resultados del algoritmo mixto

Nº	SOL¹	SOL²	CPU SOL¹	CPU SOL²	DESU (%) SOL¹/BT¹(BAS)	REL (%) CPU SOL¹/BT¹(BAS)	DESU (%) SOL²/BT²(BAS)	REL (%) CPU SOL²/BT²(BAS)
0022¹	4010.56	3206.14	0:00:22	0:07:21	20.69	51.16	1.15	160.36
0011	2840.87	2189.89	0:01:04	0:07:04	22.12	41.56	6.55	139.47
0121	4685.14	4108.27	0:02:09	0:17:45	3.33	58.90	2.45	150.64
0200	4257.36	3645.20	0:01:42	0:08:47	8.80	82.26	0.60	121.99
0201	951.61	0.00	0:00:55	0:05:55	278.64	55.56	0.00	137.07
0202	8101.91	7152.32	0:03:25	0:13:44	7.27	63.47	3.33	138.49
0210	815.72	0.00	0:00:59	0:03:25	473.97	105.36	0.00	143.36
0211	922.21	0.00	0:00:58	0:08:12	1581.33	89.23	0.00	252.31
0212	956.77	525.47	0:01:21	0:07:01	37.52	85.26	0.00	125.67
0221	840.05	89.54	0:01:01	0:07:12	64.71	45.52	41.65	174.90
0222	11221.18	9712.54	0:00:45	0:09:57	9.65	38.79	1.56	165.37
1121	1878.41	0.00	0:08:54	0:58:54	87.52	45.84	0.00	128.70
1201	815.62	0.00	0:01:57	0:10:05	29.01	64.29	0.00	132.97
1210	921.87	0.00	0:00:45	0:08:57	24.17	64.29	0.00	265.84
1211	624.63	0.00	0:01:54	0:07:41	71.08	91.20	0.00	160.63
1220	1050.17	0.00	0:01:04	0:07:47	67.86	116.36	0.00	192.18
1221	450.39	0.00	0:01:34	0:09:41	294.39	54.02	0.00	156.18
1222	3010.62	2121.31	0:01:10	0:10:23	41.26	145.83	8.18	321.13
2100	625.33	0.00	0:00:46	0:09:11	204.02	82.14	0.00	165.96
2200	2941.09	2305.17	0:01:25	0:10:12	16.45	68.55	1.46	122.16
2202	5901.45	5193.30	0:01:14	0:08:21	2.71	47.74	0.00	124.32
2211	362.02	115.85	0:01:12	0:09:47	189.13	86.75	14.45	184.01
2212	16312.78	13254.46	0:01:52	0:11:24	10.30	77.24	2.71	108.23
2220	215.59	0.00	0:00:57	0:08:52	123.85	70.37	0.00	284.49
2221	3806.17	3219.76	0:01:21	0:08:08	8.37	158.82	0.00	200.82
2222	1523.56	0.00	0:06:42	0:31:47	58.12	75.14	0.00	114.19

5.4.2.3 DISCUSIÓN.

A la vista de los resultados mostrados en la Tabla 5.3 se puede comentar que aunque globalmente el algoritmo mixto obtiene en menos tiempo la primera solución admisible, la calidad de ésta es tal que en la segunda fase la BT requiere más tiempo para encontrar las mejores soluciones. En resumen, que el ahorro de tiempo al encontrar la primera solución admisible debe ser contrapesado con la peor calidad de la misma.

5.5 RESUMEN.

En este capítulo se han diseñado los elementos necesarios para estrategias de búsqueda local en el problema de carga. La característica básica está introducción de mecanismos de oscilación en torno a la frontera de admisibilidad a partir de una modificación en la evaluación de soluciones. A partir de estos elementos se ha diseñado un algoritmo basado en BT donde la búsqueda se divide en dos fases (búsqueda de la primera solución admisible y mejora de ésta). El algoritmo diseñado se evalúa bajo un conjunto de problemas, mostrando los resultados que el método diseñado ofrece buenas soluciones consistentemente para todos los problemas considerados. Además, se presenta un análisis de las soluciones ofrecidas por el algoritmo al variar la solución inicial. Una modificación sobre el algoritmo Tabú se presenta donde, la primera fase es implementada mediante un esquema basado en técnicas de RS con la idea de acelerar la obtención de la primera solución admisible. No obstante, los resultados muestran que esto se consigue a costa de alargar la segunda de mejora (segunda fase).

6.1 INTRODUCCIÓN.

En este capítulo se abordan los problemas de diseño y carga de células de fabricación flexible teniendo en cuenta la existencia de múltiples planes de proceso para las partes a procesar. La metodología propuesta descompone la problemática en dos fases: diseño y carga de las células. La primera, diseño de las células, es resuelta una única vez y consiste en agrupar las máquinas considerando los distintos planes de proceso alternativos para cada una de las partes a procesar. La segunda, carga de las células, se resuelve una vez para cada período de planificación considerado, teniendo en cuenta los requerimientos de producción de las distintas partes a procesar.

Para la fase de diseño de las células se plantean dos aproximaciones distintas. En la primera de ellas, cada uno de los planes de proceso alternativos de las distintas partes es considerado como una parte independiente. El agrupamiento de las máquinas se realizará utilizando técnicas estándares de clusterización. Adicionalmente, se ha diseñado una Red Neuronal Auto-organizativa (RNA) para tal fin. En la segunda aproximación, la información del requerimiento de las máquinas por los distintos planes de proceso es transformada en una matriz difusa que denota el grado de requerimiento de las máquinas por parte de cada una de las partes. Para el agrupamiento de las máquinas se diseñan dos procedimientos basados en técnicas de clusterización difusas.

En la fase de carga de las células se determina para cada una de las partes con requerimientos de producción la utilización de los distintos planes de proceso alternativos con el doble objetivo de minimizar el número de movimientos intercelulares y equilibrar las cargas tanto de máquinas como de células. Para esta fase se propone un modelo de programación lineal continua.

6.2 TECNOLOGÍA DE GRUPOS GENERALIZADA.

La fabricación celular consiste en el diseño y operación de un número de células de producción compuestas de máquinas complementarias entre sí las cuales estarán físicamente próximas. El objetivo es descomponer el sistema de producción en subsistemas de forma que cada una de ellos pueda operar de forma lo más independiente posible. Entre los beneficios potenciales se encuentran la reducción y simplificación de las operaciones de manejo de materiales, reducción de los tiempos de puesta a punto, reducción del inventario en proceso, mejora de la productividad, simplificación de la secuenciación de operaciones y mejor control de la operación del taller (Wemmerlöv and Hyer 1989, Singh 1993).

Por el contrario, entre los posibles problemas de la fabricación celular está la posibilidad de desequilibrios entre las cargas de trabajo de las máquinas los cuales podrían traer como consecuencia acumulación de inventario a la entrada de las máquinas "cuello de botella" y altos tiempos de residencia de las piezas en el taller. Diferentes estudios de simulación (Flynn and Jacobs 1986, Morris and Tersine 1990, Suresh and Meredith 1994, Seifoddini and Djassemi 1996, Jensen et al 1996) han analizado la influencia tanto de la estructura de la matriz de incidencias máquina-parte como de la variabilidad de los requerimientos de producción sobre estos posibles efectos adversos. Una forma de reducir estos posibles efectos negativos es la utilización de rutas de procesado alternativas para las distintas partes (Burbridge 1992). El uso de planes de proceso alternativo para cada parte da lugar al concepto de Tecnología de Grupos Generalizada (Kusiak 1987). A continuación se revisan los distintos enfoques propuestos para el diseño y carga de células en el contexto de la Tecnología de Grupos Generalizada.

Kusiak (1987) propone un modelo de p-mediana basado en las similitudes entre planes

de proceso de forma que el número de familias de partes formadas es fijado a priori y donde para cada parte sólo uno de sus planes de proceso alternativo es seleccionado. Posteriormente las máquinas son asignadas a cada familia de partes en la misma línea que en el caso de plan de proceso único, esto es, minimizando el número de movimientos intercelulares. Shtub (1989) usa la misma metodología pero reformulando el problema como uno de asignación generalizada.

Choobineh (1988) inicialmente forma familias de partes usando coeficientes de similitud basados en las secuencias. A continuación asigna las máquinas teniendo en cuenta la capacidad de las mismas, demanda anual y consideraciones de costes. En su aproximación se consideran al menos, dos planes de proceso para cada parte, los cuales son tratados como si fuesen partes independientes. De esta forma puede ocurrir que más de una célula produzca una parte, dando lugar a más flexibilidad en la fase de carga de las células, especialmente cuando la demanda fluctúa.

Nagi et al (1990) consideran múltiples rutas para cada parte, número máximo de máquinas por célula y restricciones de capacidad de las máquinas. Estos autores presentan un algoritmo iterativo que comenzando con un diseño de células admisible itera entre dos subproblemas. El primero modela las decisiones referidas a las rutas de procesado, suponiendo la configuración como fijada, mediante programación lineal continua donde la función objetivo es minimizar el número de movimientos intercelulares. En el segundo subproblema, suponiendo como fijas las rutas de las partes y usando el mismo criterio del primer subproblema, se determina de forma heurística un diseño de las células.

Rajamani et al (1990) consideran múltiples planes de proceso así como múltiples máquinas para realizar cada operación. No obstante, para cada parte sólo un plan de proceso puede ser seleccionado y para cada una de sus operaciones sólo una máquina puede ser seleccionada. Un modelo de programación entera basado en demandas, capacidades de las máquinas y consideraciones de coste y disponibilidad presupuestaria es propuesto para conjuntamente identificar las familias de partes y los grupos de máquinas. También se imponen restricciones sobre el máximo número de máquinas por célula. Rajamani et al (1992) presentan un modelo de programación lineal continua para la carga de las células las cuales se suponen anteriormente formadas. El modelo usa como criterio la maximización del número total de partes producidas. Las variables son la fracción de la demanda de cada parte a ser procesada. En cuanto a las restricciones, se consideran las de capacidad de las máquinas y las referidas al presupuesto de operación. Además, los autores proponen otro modelo para simultáneamente diseñar y cargar las células. La función objetivo es la misma pero se introducen nuevas variables enteras para asignar las máquinas disponibles a las células sujeto a límites sobre tamaño máximo de las mismas. No se consideran ni distribución de cargas de trabajo ni movimientos intercelulares en ninguno de los modelos.

Sankaran y Kasilingam (1990) consideran múltiples rutas, cada una de las cuales requiere un conjunto dado de máquinas. Estos autores presentan dos modelos de programación lineal entera. El primero de ellos minimiza el coste total sujeto a tamaño máximo de las células y restricciones de disponibilidad de las máquinas. Únicamente una de las múltiples rutas alternativas puede ser elegida para el procesado de cada parte. En el segundo modelo se relaja esta imposición y se maximiza el número total de rutas usadas con restricciones sobre el tamaño máximo de las células y presupuesto de operación.

Kang y Wemmerlöv (1993) presentan una heurística orientada a cargas de trabajo que usa las secuencias de operaciones para generar las familias de las partes. La información sobre qué tipo de máquina puede realizar cada tipo de operación permite la asignación de operaciones a máquinas y de éstas a células teniendo en cuenta las restricciones de capacidad. El método ofrece como salida varios diseños de células y para cada uno,

diferentes evaluaciones tales como máxima utilización de cada célula y grado de equilibrio de la carga de las células.

Sankaran y Kasilingam (1993) consideran que cada operación puede ser realizada en un conjunto definido de máquinas. Sin embargo, sólo una máquina puede ser seleccionada. Un modelo de programación entera no lineal es presentado para asignar operaciones a máquinas y éstas a células con el objetivo de minimizar el coste total teniendo en cuenta restricciones sobre espacio disponible y capacidad de las máquinas. El modelo es linealizado ignorando los costes intra e intercelulares y proponen un procedimiento heurístico para modificar la solución teniendo en cuenta ambas componentes de los costes.

Logendran et al (1994) consideran múltiples planes de proceso por parte y múltiples máquinas por operación. No obstante, para cada parte sólo un plan de proceso puede ser seleccionado y para cada una de sus operaciones, únicamente una máquina puede ser seleccionada. Presentan un modelo que selecciona los planes de proceso y asigna máquinas a las operaciones, con el objetivo de minimizar los costes anuales sujeto a las restricciones de capacidad de las máquinas. El modelo de programación lineal entera resultante es solucionado heurísticamente mediante técnicas basadas en Búsqueda Tabú. La solución puede ser representada como una matriz de incidencias máquina-parte que puede ser entrada para cualquier algoritmo convencional de formación de células para así determinar las máquinas de las células y las familias de partes asociadas.

Mohamed (1996) considera que cada operación puede ser realizada en cualquiera de las máquinas flexibles de un conjunto dado. Un modelo de programación entera mixta para varios períodos teniendo en cuenta restricciones capacidad tanto de las máquinas como de los almacenes de herramientas es usado para seleccionar las partes a procesar y cargar las máquinas. La solución da lugar a matrices de incidencias máquina-parte distintas para los diferentes períodos, lo que le permite avanzar la idea de reconfiguración flexible o dinámica. En otras palabras, las máquinas y las partes son agrupadas de forma distinta en cada período. Presentan un procedimiento heurístico para el equilibrado de cargas de trabajo. La reconfiguración de las células es un proceso lógico el cual no tiene porqué implicar traslado físico de las máquinas.

Shanker y Agrawal (1997) consideran que cada operación puede ser realizada en un conjunto dado de máquinas pero debe ser asignada sólo a una de ellas. Un modelo de programación lineal entera para asignar operaciones, asignar las máquinas disponibles y fijar un número definido de grupos de partes es usado con el objetivo de minimizar el coste de los movimientos intercelulares. Consideran cotas superiores e inferiores sobre los tamaños de las células y los grupos de partes y presentan distintos procedimientos de resolución, unos jerarquizados y otros no. En los jerarquizados, en primer lugar se forman las familias de partes, las operaciones son asignadas usando una heurística, y luego las máquinas son asignadas a los grupos usando un modelo de transporte; o se asignan las operaciones de forma heurística y luego se agrupan las partes y las máquinas usando otra heurística. Para el método no jerarquizado proponen un algoritmo de optimización Lagrangiana basado en el subgradiente.

Resumiendo los enfoques anteriores se puede destacar:

- Algunos de ellos imponen o que sólo uno de los planes de proceso alternativos puede ser seleccionado, o que una operación puede ser asignada a una única máquina. Como consecuencia, sólo se aprovecha parcialmente la flexibilidad ofrecida por los planes de proceso alternativos de las partes.
- Otros usan la multiplicidad de rutas para minimizar costes (téngase en cuenta que los costes del sistema de manejo de materiales son función de los movimientos

intercelulares), pero sin tener en cuenta la distribución de cargas de trabajo a pesar del impacto de las mismas sobre la tasa de producción e inventario en proceso.

- La mayoría de las aproximaciones son estáticas. El problema es resuelto una vez y para siempre. La variabilidad en las demandas, las cuales influyen sobre el sistema, no son tenidas en cuenta.

Entre las ideas a explorar:

- Las máquinas deberían ser agrupadas considerando los planes de proceso alternativos.
- Las familias de partes no deberían ser formadas de forma estática. La asignación de los planes de procesamiento alternativos debería ser calculada teniendo en cuenta los requerimientos de producción de las partes. Además, cuestiones a tener en cuenta serán tanto las capacidades de las máquinas como el equilibrado de las cargas de trabajo.

6.3 METODOLOGÍA PROPUESTA.

La metodología propuesta para el diseño y carga de células de fabricación flexible está compuesta de dos etapas. La inicial, resuelta una sólo vez, corresponde al diseño de las células. Para esta etapa se presentan dos aproximaciones que difieren en la forma en la que son tratados los planes de proceso alternativos de cada una de las partes. En la primera aproximación, cada plan de proceso será tratado como una parte independiente (virtual), ignorando la información de qué planes de proceso corresponden a la misma parte (real). El objetivo es identificar grupos de máquinas que sean conjuntamente requeridas por grupos de planes de proceso. En la segunda aproximación, a partir de los planes de proceso alternativos de una parte, se construye un plan de proceso difuso para la misma. El objetivo será identificar grupos de máquinas que sean conjuntamente requeridas por estos planes de proceso difusos.

En la segunda etapa se determinará la proporción de la demanda de cada parte que será procesada por los correspondientes planes de proceso alternativos. Debido a que las células de máquinas han sido previamente formadas (en la primera etapa), el número de movimientos intercelulares a lo largo de cada ruta es conocido. El objetivo de esta fase es minimizar el número de movimientos intercelulares, así como equilibrar las cargas de trabajo, tanto dentro de cada célula, como entre células. Los objetivos de equilibrado de las cargas serán modelados mediante restricciones sobre las desviaciones de las cargas de trabajo de máquinas y células respecto de sus valores medios.

6.3.1 NOTACIÓN.

La notación usada será la siguiente:

Subíndices:

- i partes
- j planes de proceso
- k,l máquinas
- r operaciones
- s células

Datos:

- P Número de partes
- M Número de máquinas

- PP_i Número de planes de proceso de la parte i
 C_{max} Número máximo de células
 m_{ijr} Máquina requerida por la operación r del plan de proceso j de la parte i
 $a_{kij} = \begin{cases} 1 & \text{si } \exists r : m_{ijr} = k \\ 0 & \text{en caso contrario} \end{cases}$ matriz de incidencias máquina-plan de proceso
 t_{ijr} Tiempo de procesado de la operación r del plan de proceso j de la parte i
 D_i Demanda de la parte i
 H_k Capacidad de la máquina k (en unidades de tiempo)
 α Máximo desequilibrio entre células
 β Máximo desequilibrio dentro de las células

Variables de decisión en la fase del diseño de células:

- C Número de células
 $K(s)$ Conjunto de máquinas en la célula s
 $J(s)$ Conjunto de planes de proceso en familia s

Salida adicional de la fase de diseño de células:

- n_{ij} Número de movimientos intercelulares a lo largo de la ruta j de la parte i
 q_s Número de máquinas en la célula s

Variables de decisión en la fase de carga de células:

- θ_{ij} Proporción de la demanda de la parte i asignada a su ruta j

Salida adicional de la fase de carga de células:

- u_k Carga de trabajo de la máquina k
 w_s Carga de trabajo total de la célula s
 w_{med} Carga de trabajo media de las células

6.3.2 PLANES DE PROCESO COMO PARTES INDEPENDIENTES.

Para la determinación de las máquinas que han de configurar cada célula se han usado dos enfoques. Uno consiste en usar los algoritmos de clusterización tradicionales (Chen et al 1995). Otro es utilizar una RNA. En este apartado sólo se presenta el enfoque que se ha diseñado como alternativa a los métodos tradicionales.

6.3.2.1 RED NEURONAL AUTO-ORGANIZATIVA.

El problema de formación de células puede ser modelado mediante programación combinatoria. Usando como variables de decisión:

$$X_{ks} = \begin{cases} 1 & \text{si maquina } k \text{ esta en celula } s \\ 0 & \text{en caso contrario} \end{cases}$$

la problemática puede ser formulada como un modelo de programación binaria con función objetivo cuadrática en la siguiente forma:

$$\text{maximizar } F(X) = \sum_{k=1}^M \sum_{l=1}^M S_{kl} \sum_{s=1}^{C_{\max}} X_{ks} X_{ls} \tag{6.1}$$

sujeto a:

$$\sum_{s=1}^{C_{\max}} X_{ks} = 1 \quad \forall k \tag{6.2}$$

$$X_{kl} = 0,1 \quad \forall k, s \tag{6.3}$$

En (6.3) se impone que cada máquina es asignada a una única célula. Las restricciones (6.4) imponen la integridad sobre las variables de decisión. La función objetivo (6.2) representa la suma de similitudes entre cada par de máquinas dentro de cada célula.

Para computar los coeficientes de similitud (CS) entre dos máquinas, se propone el uso de una extensión de los coeficientes de similitud de Jaccard (Lozano et al (1997)):

$$S_{kl}(p) = \frac{(1-p)e_{kl} - p(\hat{e}_{kl} + \hat{e}_{lk})}{e_{kl} + \hat{e}_{kl} + \hat{e}_{lk}} \tag{6.4}$$

donde se hace uso de un parámetro $0 \leq p \leq 1$ que representa el peso asignado a los huecos dentro de las células. A los movimientos intercelulares se le asigna un peso de $1-p$. Con e_{kl} representamos el número de planes de proceso (partes vistuales) que requieren ambas máquinas k y l , mientras que \hat{e}_{kl} es el número de planes de proceso que requieren la máquina k pero no la l . Si las máquinas k y l no caen en la misma célula, e_{kl} movimientos intercelulares deberán hacerse. Por lo tanto, a e_{kl} se le da un peso de $1-p$. Por el contrario, si las máquinas k y l forman parte de la misma célula, $\hat{e}_{kl} + \hat{e}_{lk}$ representará el número de huecos. El hecho de que los CS puedan tomar valores negativos ($-p \leq S_{ij}(p) \leq 1-p$) impide que configuraciones formadas por una única célula sean necesariamente óptimas. Los CS definidos según (6.4) coinciden con los de Jaccard para $p=0$.

La RNA propuesta en este apartado es similar a las diseñadas para un problema de secuenciación en una línea de montaje de coches (Smith 1996) y para el problema de asignación de frecuencias en telefonía celular (Smith 1997). La arquitectura de la red para nuestro problema de agrupamiento de máquinas en Tecnología de Grupos Generalizada es la mostrada en la Figura 6.1.

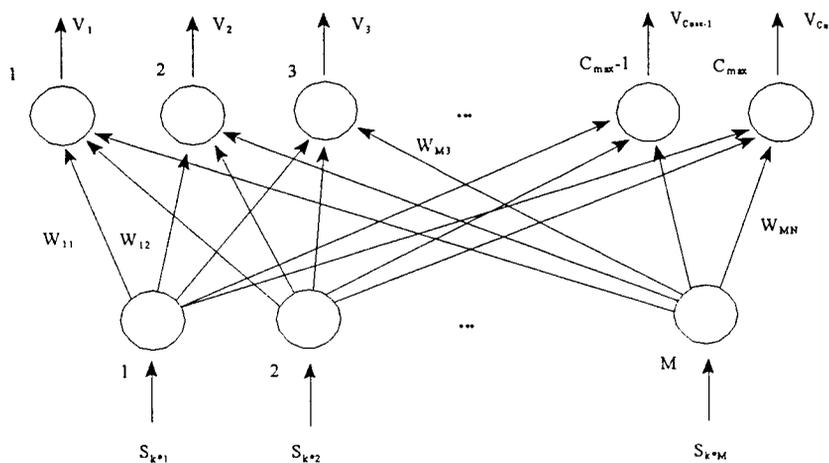


Figura 6.1. Arquitectura de la RNA

La RNA es una red hacia adelante compuesta por dos niveles. El nivel de entrada está compuesto por M unidades (máquinas) y el de salida por C_{max} nodos (células). El peso entre la unidad k del nivel de entrada y la s del de salida viene dado por W_{ks} y representa la relajación continua de las variables de decisión X_{ks} en (6.1)-(6.3). En otras palabras, W_{ks} es una medida del grado de pertenencia de la máquina k a la célula s . Como patrones de entrada se usan M vectores, uno para cada máquina. El patrón de entrada correspondiente a la máquina k^* es un vector de M componentes donde la componente k -ésima representa el CS (calculado según (6.4)) entre las máquinas k y k^* .

Cuando presentamos el patrón de entrada correspondiente a la máquina k^* , para cada nodo s del nivel de salida definimos su potencial, V_{sk^*} , como la contribución a la función objetivo al asignar la máquina k^* a la célula s :

$$V_{sk^*} = \sum_{k=1}^M W_{ks} S_{k^*k} \quad (6.5)$$

Por competición entre los nodos del nivel de salida obtenemos el nodo vencedor m_0 como aquel para el que se obtiene la máxima contribución:

$$V_{m_0k^*} \geq V_{sk^*} \quad \forall s \quad (6.6)$$

Además, ordenando los nodos de salida de acuerdo a su potencial, definimos los nodos vecinos del vencedor según:

$$V_{m_0k^*} \geq V_{m_1k^*} \geq V_{m_2k^*} \geq \dots \geq V_{m_\eta k^*} \quad \forall s \quad (6.7)$$

donde $\eta \geq 1$ es el tamaño de la vecindad. Obsérvese que los vecinos son determinados de acuerdo a consideraciones de contribución a la función objetivo en lugar de consideraciones espaciales. Una vez que tanto el nodo vencedor como sus vecinos han sido determinados, los pesos son modificados de acuerdo a una modificación de la regla de adaptación de pesos de Kohonen. Durante el proceso de aprendizaje, tanto el tamaño de la vecindad como la magnitud de la adaptación de los pesos son decrementadas de forma gradual.

El algoritmo completo es el siguiente:

1. Inicializar aleatoriamente los pesos y normalizarlos según:

$$\begin{aligned} 0 \leq W_{ks} \leq 1 \quad \forall k, s \\ \sum_{s=1}^{C_{max}} W_{ks} = 1 \quad \forall k \end{aligned} \quad (6.8)$$

2. Seleccionar aleatoriamente una máquina k^* y presentar su correspondiente patrón de entrada $S_{k^*k} \forall k$.

3. Calcular el potencial de cada nodo s del nivel de salida V_{sk^*} siguiendo (6.5).

4. Determinar por competición el nodo vencedor m_0 :

$$V_{m_0k^*} = \max_s V_{sk^*} \quad \forall s \quad (6.9)$$

así como sus nodos vecinos:

$$m_1, m_2, \dots, m_\eta$$

y el resto de nodos del nivel de salida:

$$m_{\eta+1}, m_{\eta+2}, \dots, m_{c_{\max}-1}$$

5. Modificar los pesos W_{k^*s}

5.1. Para el nodo vencedor y sus vecinos:

$$\Delta W_{k^*s} = \alpha(h, t) [1 - W_{k^*s}] \quad \forall s: V_{sk^*} \geq V_{m_{\eta}k^*} \quad (6.10)$$

5.2. Para el resto de nodos del nivel de salida:

$$\Delta W_{k^*s} = -\beta(t) W_{k^*s} \quad \forall s: V_{sk^*} \leq V_{m_{\eta+1}k^*} \quad (6.11)$$

donde

$$\alpha(\eta, t) = \beta(t) \exp \left[- \frac{|V_{m_0k^*} - V_{sk^*}|}{|V_{m_0k^*} - V_{m_{c_{\max}-1}|} \right] \quad (6.12)$$

β es un escalar positivo que se decrementa a medida que el aprendizaje avanza. El resto de los pesos W_{ks} con $k \neq k^*$ no se modifican, esto es, $\Delta W_{ks} = 0$. Los nuevos pesos vendrán dados por:

$$W_{ks} \leftarrow W_{ks} + \Delta W_{ks} \quad (6.13)$$

a los cuales se les aplicará la normalización (6.8).

6. Repetir desde el paso 2. hasta que todas las máquinas han sido seleccionadas como vectores de entrada. Esto será una iteración t . Decrementar β y η . Repetir desde el paso 2. hasta $|\Delta W_{ks}| \approx 0 \quad \forall k, s$. Decrementar el tamaño de la vecindad η linealmente.

7. Repetir el proceso entero desde el paso 2 hasta que $\eta = 1$.

De forma general, una vez terminado el proceso de aprendizaje de la RNA, no se verificarán las restricciones de exclusividad en las pertenencias de las máquinas a las células, esto es, dada una máquina k , esta puede presentar pertenencias estrictamente positivas para más de una célula. Como consecuencia, un método que imponga las restricciones de exclusividad debe ser aplicado para poder asignar las máquinas a las células.

6.3.2.2 ALGORITMO DE ENFRIAMIENTO BASADO EN LOS CAMPOS MEDIOS.

Con el objetivo de forzar las restricciones de exclusividad se propone una variante del algoritmo de Enfriamiento basado en la Teoría del Campo Medio (AECM) utilizado por Van Den Bout y Miller (1989) para el problema de partición de grafos.

A partir de los pesos ofrecidos por la RNA al final del proceso de aprendizaje, el AECM aproxima la función de energía a optimizar:

$$E(W) = -\frac{1}{2} F(W) = -\frac{1}{2} \left(\sum_{k=1}^M \sum_{l=1}^M S_{kl} \sum_{s=1}^{C_{\max}} W_{ks} W_{ls} \right) \quad (6.14)$$

por su valor medio:

$$\langle E(W) \rangle = -\frac{1}{2} \left\langle \sum_{k=1}^M \sum_{l=1}^M S_{kl} \sum_{s=1}^{C_{\max}} W_{ks} W_{ls} \right\rangle = -\frac{1}{2} \sum_{k=1}^M \sum_{l=1}^M S_{kl} \sum_{s=1}^{C_{\max}} \langle W_{ks} W_{ls} \rangle = -\frac{1}{2} \sum_{k=1}^M \sum_{l=1}^M S_{kl} \sum_{s=1}^{C_{\max}} \langle W_{ks} \rangle \langle W_{ls} \rangle \quad (6.15)$$

Las ecuaciones del AECM son:

$$\langle W_{ks} \rangle = \frac{\exp\left(-\frac{1}{T} \frac{\partial \langle E(W) \rangle}{\partial \langle W_{ks} \rangle}\right)}{\sum_{s=1}^{C_{\max}} \exp\left(-\frac{1}{T} \frac{\partial \langle E(W) \rangle}{\partial \langle W_{ks} \rangle}\right)} = \frac{\exp\left(-\frac{1}{T} h_{ks}\right)}{\sum_{s=1}^{C_{\max}} \exp\left(-\frac{1}{T} h_{ks}\right)} = \frac{\exp\left(\frac{1}{T} V_{sk}\right)}{\sum_{s=1}^{C_{\max}} \exp\left(-\frac{1}{T} V_{sk}\right)} \quad (6.16)$$

donde

$$h_{ks} = \frac{\partial \langle E(W) \rangle}{\partial \langle W_{ks} \rangle} = \langle E(W) \rangle \Big|_{W_{ks}=1, W_{kr}=0 \forall s' \neq s} - \langle E(W) \rangle \Big|_{W_{ks}=0 \forall s} = -V_{sk} \quad (6.17)$$

Obsérvese que con las ecuaciones (6.16) los pesos siguen verificando la normalización (6.8). Las restricciones de exclusividad se imponen a través de un enfriamiento en la temperatura T. En la Figura 6.2 se presenta un pseudocódigo del AECM. Para cada temperatura y mientras no se alcance un punto de equilibrio, se selecciona aleatoriamente una máquina k. Todas las pertenencias de esta máquina k a las distintas células $\langle W_{ks} \rangle$ son igualadas a cero. A continuación, cada pertenencia individualmente es forzada a 1 calculando el cambio provocado en la función de energía según (6.17).

Para valores altos de T, las pertenencias de una máquina estarán uniformemente distribuidas entre las distintas células. A medida que se va decrementando la temperatura T, la pertenencia $\langle W_{ks} \rangle$ comenzará a "coagular" en aquella célula s con menor h_{ks} , o lo que es lo mismo, mayor potencial V_{sk} . La normalización (6.16) garantiza que cuando la convergencia es completa, cada máquina estará presente en una sólo célula.

```

T ← temperatura inicial
while (T > Tmin)
  do while (no se alcanza el equilibrio)
    seleccionar aleatoriamente una máquina k
    for (s ← 1; s ≤ Cmax; s ← s+1)
      <Wks> ← 0
    for (suma ← 0, s ← 1; s ≤ Cmax; s ← s+1)
      Vsk ← Vsk | <Wks>=1
      suma ← suma+exp (Vsk / T)
    for (s ← 1; s ≤ Cmax; s ← s+1)
      <Wks> ← exp (Vsk / (T.suma))
  T ← α T
    
```

Figura 6.2. Pseudocódigo del AECM

Debido a que el proceso de aprendizaje de la RNA dejará las pertenencias cercanas a cumplir las restricciones de exclusividad, la temperatura inicial podrá no ser elevada lo cual dará lugar a un número pequeño de iteraciones a realizar en el AEEM hasta alcanzar la condición de saturación:

$$\frac{1}{M \cdot C_{\max}} \sum_{k=1}^M \sum_{s=1}^{C_{\max}} W_{ks} (1 - W_{ks}) \leq 0.01 \quad (6.18)$$

6.3.3 FORMACIÓN DE CÉLULAS BASADA EN TÉCNICAS DIFUSAS.

Desde su origen a mitad de los años 60, la Teoría de los Conjuntos Difusos ha sido usada en diversas áreas tales como reconocimiento de patrones, sistemas expertos, control y gestión de la producción. En el campo de gestión de la producción, estos métodos han sido usados en problemáticas tales como diseño de instalaciones, gestión de proyectos, planificación agregada, tecnología de grupos y secuenciación. Una revisión tanto de la metodología como de las aplicaciones en gestión de la producción se puede encontrar en Karwowski y Evans (1986).

Dentro de la tecnología de grupos se han propuesto diversas aproximaciones difusas para los problemas de agrupamiento de máquinas y partes (Xu y Wang 1989, Chu y Hayya 1991, y Zhang y Wang 1995). En este apartado se presentan dos nuevos algoritmos basados en el algoritmo de la C-medias (Bezdek 1981) para el problema de agrupamiento de máquinas en Tecnología de Grupos Generalizada. En el primero de los algoritmos se utilizan como variables de decisión las pertenencias de las máquinas a las células (algoritmo difuso basado en pertenencias de máquinas a células). En el segundo, las variables empleadas serán las pertenencias de las partes a las células (algoritmo difuso basado en pertenencias de partes a células).

Ambos enfoques están basados en la construcción de una matriz de incidencias difusa máquina-parte (b_{ki}) a partir de la matriz de incidencias máquina-plan de proceso (a_{kij}):

$$b_{ki} = \frac{1}{PP_i} \sum_{j=1}^{PP_i} a_{kij} \quad (6.19)$$

Esta matriz estará compuesta por M filas (máquinas) y P columnas (partes) donde la coordenada b_{ki} computa el grado de utilización de la máquina k por parte de los planes de proceso de la parte i. En el caso de que todos los planes de proceso de la parte i requieran la máquina k tendremos $b_{ki}=1$. Por el contrario, si ningún plan de proceso de la parte i utiliza la máquina k entonces $b_{ki}=0$. En el resto de los casos $b_{ki} \in (0,1)$.

6.3.3.1 ALGORITMO DIFUSO BASADO EN PERTENENCIAS DE MÁQUINAS A CÉLULAS.

Definiendo como variables de decisión el grado de pertenencia de cada una de las máquinas a las células de forma que verifiquen:

$$\begin{aligned} 0 &\leq \chi_{ks} \leq 1 && \forall ks \\ \sum_{s=1}^{C_{\max}} \chi_{ks} &= 1 && \forall ks \\ \sum_{k=1}^M \chi_{ks} &> 0 && \forall s \end{aligned} \quad (6.20)$$

Debido a que el número de matrices de pertenencias verificando (6.20) es infinito, para determinar la solución óptima necesitamos una función objetivo a optimizar:

$$G(\chi, y) = \sum_{k=1}^M \sum_{s=1}^{C_{\max}} (\chi_{ks})^f \|b_k - y_s\|^2 \quad (6.21)$$

donde y_s es un vector fila de la misma dimensión que los b_k , que representa el centro de gravedad formado por los distintos b_k ponderados con su pertenencia a la célula s :

$$y_{si} = \frac{\sum_{k=1}^M (\chi_{ks})^f b_{ki}}{\sum_{k=1}^M (\chi_{ks})^f} \quad \forall si \quad (6.22)$$

y el parámetro $f > 1$ representa el grado de difusión. Al vector \bar{y}_s lo denominaremos como centroides de la célula s . Obsérvese que para $f = 1$, los centroides coinciden con los centros de gravedad.

Algoritmo.

1. Inicialización aleatoria de las pertenencias verificando (6.20).
2. Cálculo de los centroides de cada célula según (6.22).
3. Adaptar pertenencias según:

$$\chi_{ks} = \frac{1}{\sum_{s'=1}^{C_{\max}} \left[\frac{\|b_k - y_{s'}\|^2}{\|b_k - y_s\|^2} \right]^{\frac{1}{f-1}}} \quad \forall ks \quad (6.23)$$

4. Repetir desde 2. hasta que se verifique alguna de las condiciones de finalización siguientes:

$$\begin{aligned} \max_{ks} |\chi_{ks}^{t+1} - \chi_{ks}^t| &\leq \varepsilon \\ \max_{si} |y_{si}^{t+1} - y_{si}^t| &\leq \varepsilon \end{aligned} \quad (6.24)$$

Una vez alcanzada alguna de las condiciones de finalización (6.24), las células se formarán usando los centroides como núcleos. Así pues, dada una máquina k se le asigna la célula $s(k)$ cuyo centroide está más cerca de b_k :

$$s(k) = \arg \min_s \|b_k - y_s\| \quad (6.25)$$

6.3.3.2 ALGORITMO DIFUSO BASADO EN PERTENENCIAS DE PARTES A CÉLULAS.

Definiendo como variables de decisión el grado de pertenencia de cada una de las partes a las células de forma que verifiquen:

$$\begin{aligned} 0 &\leq z_{is} \leq 1 \quad \forall ks \\ \sum_{s=1}^{C_{\max}} z_{is} &= 1 \quad \forall ks \\ \sum_{i=1}^P z_{is} &> 0 \quad \forall s \end{aligned} \quad (6.26)$$

De forma paralela al algoritmo anterior, la función objetivo a optimizar será:

$$G(z, \psi) = \sum_{k=1}^M \sum_{s=1}^{C_{\max}} (z_{is})^f \|b_i - \psi_s\|^2 \quad (6.27)$$

donde ahora el centroide de la células s , ψ_s , será un vector columna de igual dimensión que el vector columna b_i , esto es, tantas componentes como máquinas. Cada componente se calcula teniendo en cuenta las nuevas pertenencias definidas según:

$$\psi_{ks} = \frac{\sum_{i=1}^P (z_{is})^f b_{ki}}{\sum_{i=1}^P (z_{is})^f} \quad \forall ks \quad (6.28)$$

Algoritmo.

1. Inicialización aleatoria de las pertenencias verificando (6.26).
2. Cálculo de los centroides de cada célula según (6.28).
3. Adaptar pertenencias según:

$$z_{is} = \frac{1}{\sum_{s'=1}^{C_{\max}} \left[\frac{\|b_i - \psi_{s'}\|^2}{\|b_i - \psi_s\|^2} \right]^{\frac{1}{\Gamma-1}}} \quad \forall is \quad (6.29)$$

4. Repetir desde 2. hasta que se verifique alguna de las condiciones de finalización siguientes:

$$\begin{aligned} \max_{is} |z_{is}^{t+1} - z_{is}^t| &\leq \epsilon \\ \max_{ks} |\psi_{ks}^{t+1} - \psi_{ks}^t| &\leq \epsilon \end{aligned} \quad (6.30)$$

Una vez alcanzada alguna de las condiciones de finalización (6.30), las células se formarán en base a los centroides. Así pues, dada una máquina k se le asigna la célula $s(k)$ cuyo centroide requiera en mayor medida esa máquina:

$$s(k) = \arg \max_s \psi_{ks} \quad (6.31)$$

6.3.4 CARGA DE LAS CÉLULAS.

Una vez que la configuración de las células ha sido establecida, los tamaños de las células q_s y el número de movimientos intercelulares a lo largo de cada ruta n_{ij} pueden ser computados. Estos datos, junto con las demandas de cada parte D_i , la capacidad de las máquinas H_k y el tiempo de procesado total en cada máquina k debido a cada plan de proceso j de la parte i

$$T_{kij} = \sum_{r: m_{ir} = k} t_{ijr}$$

permiten la formulación del siguiente modelo de programación lineal continua para el

problema de carga de las células:

$$\text{minimizar } \sum_i \sum_j D_i n_{ij} \theta_{ij} \tag{6.32}$$

sujeito a:

$$\sum_j \theta_{ij} = 1 \quad \forall i \tag{6.33}$$

$$\sum_i \sum_j D_i T_{kij} \theta_{ij} = u_k \quad \forall k \tag{6.34}$$

$$u_k \leq H_k \quad \forall k \tag{6.35}$$

$$w_s = \sum_{k \in K(s)} u_k \quad \forall s \tag{6.36}$$

$$w_{med} = \frac{\sum_s w_s}{C} \tag{6.37}$$

$$(1 - \alpha) w_{med} \leq w_s \leq (1 + \alpha) w_{med} \quad \forall s \tag{6.38}$$

$$(1 - \beta) \frac{w_s}{q_s} \leq u_k \leq (1 + \beta) \frac{w_s}{q_s} \quad \forall k \in K(s) \tag{6.39}$$

$$\theta_{ij} \geq 0 \quad \forall i, j \quad u_k \geq 0 \quad \forall k \quad w_s \geq 0 \quad \forall s \quad w_{med} \geq 0 \tag{6.40}$$

La función objetivo (6.32) minimiza el número de movimientos intercelulares. En (6.33) se impone que cada parte será procesada en su totalidad. En las restricciones (6.34) se computa la carga de trabajo de cada una de las máquinas, la cual no podrá exceder su capacidad de acuerdo con (6.35). En (6.36) se computa la carga de trabajo de las células. Con la restricción (6.37) se determina la carga de trabajo media de las células. En (6.38) se impone que el máximo desequilibrio de las cargas de las células medido respecto a la carga media de las células no podrá exceder αw_{med} . De forma análoga, en (6.39) se impone que el máximo desequilibrio de las cargas de trabajo de las máquinas de una célula medido respecto a la carga media de las máquinas de esa célula w_s/q_s , no podrá ser mayor que $\beta w_s/q_s$.

6.4 ILUSTRACIÓN.

En este apartado se ilustra la metodología propuesta, incluyendo todos los enfoques presentados, a partir de los datos de un ejemplo. Inicialmente, se diseñan las células, las cuales son cargadas posteriormente.

6.4.1 DISEÑO DE LAS CÉLULAS.

La fase de diseño de las células se realiza usando las dos aproximaciones propuestas. En la primera de ellas, cada uno de los planes de proceso de las distintas partes se trata como una parte independiente. En la segunda aproximación, a partir de los planes de proceso de una parte se forma un plan de proceso difuso. Inicialmente, se presentan los datos de referidos al sistema de producción considerado.

6.4.1.1 DATOS DE ENTRADA.

Consideremos un sistema de producción compuesto por 14 máquinas y 10 partes. En la Tabla 6.1 se muestran las distintas rutas alternativas para cada parte.

Tabla 6.1. Descripción de planes de proceso

Parte (i)	Plan de Proceso (r)	Secuencia de operaciones (máquina, tiempo de procesado)					
		1ª operación	2ª operación	3ª operación	4ª operación	5ª operación	6ª operación
1	1	(8,5)	(6,12)	(8,7)			
	2	(5,8)	(7,6)	(4,5)	(5,5)	(7,6)	
	3	(4,10)	(7,5)	(4,10)	(7,8)	(5,4)	
2	1	(10,6)	(2,5)	(11,4)	(10,5)	(11,5)	(2,5)
	2	(11,3)	(3,9)	(11,5)	(2,10)		
3	1	(8,2)	(9,8)	(8,3)			
	2	(13,9)	(1,7)	(13,9)			
	3	(12,8)	(1,4)	(12,6)	(7,4)	(13,4)	(1,5)
4	1	(13,7)	(12,10)	(13,6)			
	2	(9,7)	(8,4)	(14,12)	(9,9)	(8,5)	(6,8)
5	1	(14,4)	(6,6)	(14,4)			
	2	(8,4)	(6,4)	(9,4)	(8,4)	(9,5)	
6	1	(14,3)	(9,6)				
	2	(6,8)	(8,7)				
7	1	(8,4)	(9,3)	(6,5)	(8,6)	(9,3)	(14,5)
	2	(8,5)	(6,4)				
	3	(5,7)	(4,8)	(7,5)	(5,6)		
8	1	(13,7)					
	2	(4,6)					
9	1	(5,6)	(4,6)	(7,5)			
	2	(3,4)	(11,3)	(3,6)			
10	1	(5,5)	(4,5)	(13,5)	(5,7)	(4,5)	
	2	(10,10)					

Obsérvese que los datos representados en la Tabla 6.1 representan un escenario lo más general posible en cuanto al número de planes de proceso por parte, número de operaciones por plan de proceso y la posibilidad de visitar más de una vez misma máquina dentro del mismo plan de proceso.

6.4.1.2 PLANES DE PROCESO COMO PARTES INDEPENDIENTES.

A partir de los datos de la Tabla 6.1, construimos la matriz de incidencias máquina-plan de proceso (véase Tabla 6.2). Como algoritmo de clusterización usaremos, además de la RNA, ZODIAC (Chandrasekharan y Rajagopalan 1987). Inicialmente, se muestra la configuración de ZODIAC (Tabla 6.3). Posteriormente, se presentan las células obtenidas con la RNA (Tabla 6.4).

Tabla 6.2. Matriz de incidencias máquina-plan de proceso

		Parte, Plan de Proceso																							
		1,1	1,2	1,3	2,1	2,2	3,1	3,2	3,3	4,1	4,2	5,1	5,2	6,1	6,2	7,1	7,2	7,3	8,1	8,2	9,1	9,2	10,1	10,2	
1							1	1																	
2				1	1																				
3					1																1				
4		1	1															1	1	1					1
5		1	1															1		1					1
6	1									1	1	1	1	1	1	1	1								
7		1						1										1		1					
8	1					1				1	1	1	1	1	1	1	1								
9						1				1	1	1	1	1	1	1	1								
10				1																					1
11				1	1																1				
12								1	1																
13							1	1	1										1						1
14										1	1	1	1	1	1	1	1								

Máquina

Tabla 6.3. Resultados de ZODIAC: células y familias

		Parte, Plan de Proceso																						
		1,2	1,3	7,3	8,2	9,1	10,1	3,2	3,3	4,1	8,1	1,1	3,1	4,2	5,1	5,2	6,1	6,2	7,1	7,2	10,2	2,3	2,1	9,2
4		1	1	1	1	1	1																	
5		1	1	1		1	1																	
7		1		1		1		1																
1								1	1															
12								1	1															
13							1	1	1	1	1													
6												1		1	1	1	1	1	1	1				
8												1	1	1		1	1	1	1	1				
9												1	1	1	1	1	1	1	1	1				
10																					1		1	
14												1	1	1	1	1	1	1	1	1				
2																						1	1	
3																						1		1
11																						1	1	1

Máquina

Tabla 6.4. Resultados de la RNA (p = 0.1): células y familias

		Parte, Plan de Proceso																							
		1,2	1,3	7,3	8,2	9,1	10,1	3,2	3,3	4,1	8,1	1,1	3,1	4,2	5,1	5,2	6,1	6,2	7,1	7,2	10,2	2,3	2,1	9,2	
4		1	1	1	1	1	1																		
5		1	1	1	1	1	1																		
7		1		1	1	1		1																	
1								1	1																
12								1	1																
13							1	1	1	1	1														
6												1	1	1	1	1	1	1	1	1					
8												1	1	1	1	1	1	1	1	1					
9												1	1	1	1	1	1	1	1	1					
14												1	1	1	1	1	1	1	1	1					
2																						1	1		
3																						1			1
10																					1				1
11																						1	1		1

Máquina

Obsérvese que para ambas configuraciones de las células, algunas partes (1, 3, 4, 7, 8, 9 y 10) presentan planes de proceso en distintas células, aumentando así la flexibilidad para la fase de carga. Por otro lado, en el caso de ZODIAC, únicamente tres planes de proceso dan lugar a movimientos intercelulares ($n_{2,1}=3$, $n_{3,3}=2$ y $n_{10,1}=2$). De igual forma, en la configuración de la RNA sólo aparecen movimientos intercelulares en dos planes de proceso ($n_{3,3}=2$ y $n_{10,1}=2$). Referido a los requerimientos computacionales, ZODIAC necesita 2.234 segundos, frente a los 1.484 segundos de la RNA, ambos en un Pentium PC a 100 Mhz.

6.4.1.3 PLANES DE PROCESO DIFUSOS.

Los parámetros especificados para ambos algoritmos difusos son:

C_{max} : 5 (número máximo de células a formar), f : 1.5 (grado de difusión) y ϵ : 0.001 (error la condición de parada).

En la Tabla 6.5 se muestran las pertenencias de máquinas a células ofrecidas por la variante descrita en el apartado 6.3.4.1.

Tabla 6.5. Pertenencias de máquinas a células (χ_{ks})

Máquina	Célula				
	1	2	3	4	5
1	0.944	0.018	0.023	0.007	0.008
2	0.008	0.004	0.980	0.004	0.004
3	0.020	0.019	0.954	0.003	0.004
4	0.010	0.976	0.007	0.001	0.006
5	0.002	0.994	0.002	0.001	0.001
6	0.011	0.011	0.002	0.971	0.005
7	0.036	0.925	0.021	0.008	0.010
8	0.007	0.005	0.002	0.986	0.000
9	0.016	0.003	0.003	0.976	0.002
10	0.121	0.061	0.792	0.026	0.000
11	0.005	0.005	0.988	0.000	0.002
12	0.958	0.009	0.014	0.017	0.002
13	0.921	0.031	0.022	0.002	0.024
14	0.009	0.003	0.003	0.983	0.002

Aplicando las ecuaciones (6.25) para asignar las máquinas a las células, se obtiene igual configuración que la de la RNA (Tabla 6.4). Además, en la Figura 6.3 se presenta la la función objetivo (6.21) optimizada en el algoritmo (FO) para cada iteración (ITER). Obsérvese, que el algoritmo únicamente realiza 8 iteraciones, en las que se invierten un tiempo de 1.34 segundos.

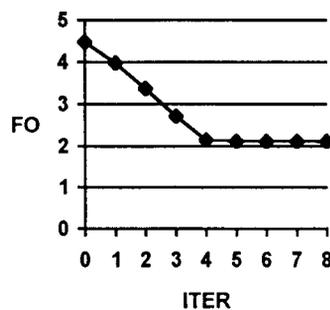


Figura 6.3. Evolución de la función objetivo (6.21)

Para el segundo algoritmo difuso (descrito en la sección 6.3.4.2), en la Tabla 6.6 mostramos los centroides, ψ_{ks} , a partir de los cuales se obtiene la asignación de máquinas a células.

Tabla 6.6. Centroides (ψ_{ks})

Máquina	Célula				
	1	2	3	4	5
1	0.956	0.006	0.013	0.017	0.008
2	0.006	0.006	0.984	0.000	0.004
3	0.010	0.024	0.958	0.004	0.003
4	0.001	0.985	0.003	0.005	0.006
5	0.002	0.996	0.000	0.001	0.001
6	0.021	0.010	0.000	0.973	0.005
7	0.015	0.946	0.025	0.004	0.010
8	0.012	0.000	0.000	0.988	0.000
9	0.010	0.009	0.000	0.979	0.002
10	0.100	0.071	0.802	0.002	0.024
11	0.003	0.007	0.988	0.001	0.001
12	0.962	0.005	0.011	0.020	0.002
13	0.952	0.000	0.002	0.022	0.024
14	0.008	0.006	0.001	0.985	0.000

A partir de los centroides, y aplicando las ecuaciones (6.31) para asignar las máquinas a las células, se obtiene nuevamente igual configuración que la de la RNA (Tabla 6.4). Además, en la Figura 6.4 se presenta la la función objetivo (6.27) optimizada en el algoritmo (FO) para cada iteración (ITER). En este caso, se realizan 9 iteraciones en las que se invierten 1.78 segundos.

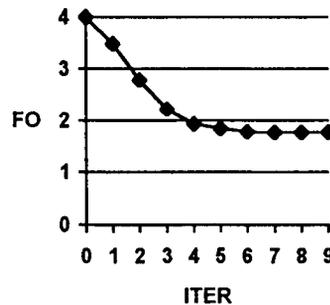


Figura 6.4. Evolución de la función objetivo (6.27)

6.4.2 CARGA DE CÉLULAS.

De la fase de diseño hemos obtenido dos configuraciones distintas. En este apartado nos planteamos cargar las células para cada una de las configuraciones, dados unos requerimientos de producción. Así, en la Tabla 6.7 se presentan los requerimientos de producción para el período actual de planificación.

Tabla 6.7. Requerimientos de producción para el período actual

Parte (i)	1	2	3	4	5	6	7	8	9	10
Demanda (D _i)	-	18	15	8	9	-	16	12	18	-

Obsérvese que no todas las partes son requeridas en este período. Además, se supone para todas las máquinas igual capacidad ($H_k=240$ unidades de tiempo).

6.4.2.1 CONFIGURACIÓN DE ZODIAC.

Dando la configuración de células mostrada en la Tabla 6.3 y solucionando el modelo de programación lineal (6.32)-(6.40) con $\alpha=0.25$ y $\beta=0.15$, obtenemos el mix de rutas mostrado en la Tabla 6.8 y un valor de la función objetivo de 49.5 movimientos intercelulares. El tiempo de computación requerido por el optimizador XA (1994) es 0.7 segundos.

Tabla 6.8. Rutas de procesado para el período actual ($\alpha=0.25, \beta=0.15$) (I)

Parte (i)	Mix de rutas		
	θ_{11}	θ_{12}	θ_{13}
1	-	-	-
2	0.544	0.456	-
3	0	0.327	0.673
4	0.148	0.852	-
5	0.887	0.113	-
6	-	-	-
7	0	0.595	0.405
8	0.232	0.768	-
9	0.634	0.366	-
10	-	-	-

En la Tabla 6.9 se muestran las cargas de trabajo de máquinas y células. Para cada célula se presentan las cargas de cada una de las máquinas que la conforman. La carga de cada célula viene dada por la acumulación de la carga de las máquinas que forman parte de ella. La carga media de la célula se ha computado como cociente entre la total de la célula y el número de máquinas de la misma. Además, se presenta la carga media de las células.

Tabla 6.9. Cargas de trabajo para el período actual ($\alpha=0.25, \beta=0.15$) (I)

Célula	Máquina														Carga de célula	Carga media	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14			
1				176	153		130									459	153
2	125											153	164			442	147
3						145		117	118	108					146	634	127
4		180	140								174					494	165
Carga media de las células:																507	

6.4.2.2 CONFIGURACIÓN DE LA RNA Y LOS ALGORITMOS DIFUSOS.

Resolviendo el problema de carga para la configuración de células de la RNA y los algoritmos difusos (Tabla 6.4) con los requerimientos de producción de la Tabla 6.7 e iguales restricciones de equilibrado de las cargas, obtenemos los resultados de las Tablas 6.10 y 6.11. El número de movimientos intercelulares en este caso (función objetivo del problema de carga) es 17.2. El tiempo de computación requerido para la resolución del modelo de programación lineal es 0.5 segundos.

Tabla 6.10. Rutas de procesado para el período actual ($\alpha=0.25, \beta=0.15$) (II)

Parte (i)	Mix de rutas		
	θ_{11}	θ_{12}	θ_{13}
1	-	-	-
2	0.673	0.327	-
3	0.149	0.277	0.574
4	0.060	0.940	-
5	1	0	-
6	-	-	-
7	0.077	0.577	0.346
8	0.342	0.658	-
9	0.552	0.448	-
10	-	-	-

Tabla 6.11. Cargas de trabajo para el período actual ($\alpha=0.25, \beta=0.15$) (II)

Célula	Máquina														Carga de célula	Carga media	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14			
1				151	132		112									395	132
2	107											125	144			376	125
3						157		137	146					168		609	152
4		180	134							133	180					376	94
Carga media de las células:																502	

6.5 EXPERIMENTOS ADICIONALES.

En este apartado nos planteamos algunas experiencias adicionales, tanto en lo referido al modelo de carga, como a la RNA. En el caso del modelo de carga, nos planteamos diversos experimentos de análisis de sensibilidad. En el caso de la RNA, el objetivo será comparar las soluciones de la misma con las de otros procedimientos heurísticos.

6.5.1 ANÁLISIS DE SENSIBILIDAD EN EL MODELO DE CARGA.

En lo referido al modelo de carga nos planteamos dos experiencias. En la primera de ellas pretendemos analizar la sensibilidad de la función objetivo del modelo de carga (recuérdese que representa el número de movimientos intercelulares) al variar los desequilibrios máximos permitidos, esto es, los parámetros α y β . En la segunda cuestión nos planteamos ilustrar el cambio en el mix de planes de proceso al cambiar los requerimientos de producción.

6.5.1.1 FUNCIÓN OBJETIVO.

Dados los requerimientos de producción de la Tabla 6.7, resolvemos el modelo de carga (6.32)-(6.40) para distintos valores de los parámetros α y β . Así, variamos ambos parámetros desde 0.05 a 0.95, en intervalos de 0.1. Para cada combinación de los parámetros resolvemos el modelo de carga resultante. En la Figura 6.5 se muestra que la función objetivo (número de movimientos intercelulares) es mucho más sensible a β que a α , y que cuando ambos parámetros son muy pequeños, esto es, cuando las restricciones de desequilibrio son muy estrictas, puede que no exista solución admisible en (6.32)-(6.40).

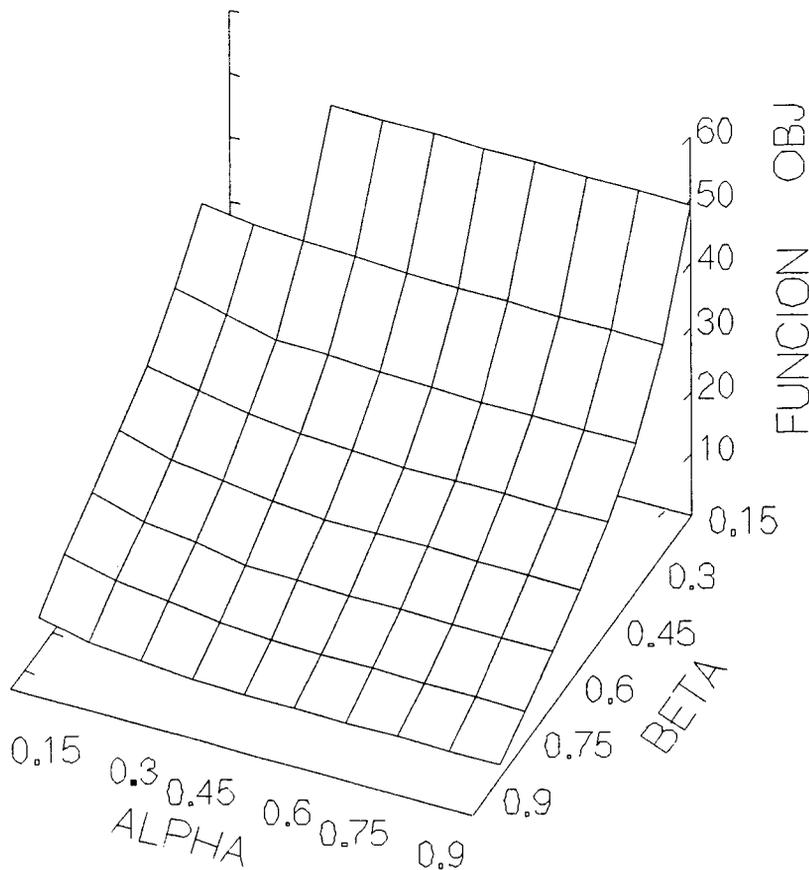


Figura 6.5. Resultados de sensibilidad

6.5.1.2 MIX DE RUTAS.

Para ilustrar la naturaleza dinámica del problema de carga de las células, proponemos la resolución del mismo para el próximo período de planificación donde los requerimientos de producción son los mostrados en la Tabla 6.12.

Tabla 6.12. Requerimientos de producción para el próximo período

Parte (i)	1	2	3	4	5	6	7	8	9	10
Demanda (D _i)	10	18	18	8	10	8	-	-	-	12

Obsérvese que tanto partes como cantidades requeridas, cambian con respecto al período de planificación previo (Tabla 6.7). Tomando como configuración de las células la ofrecida por ZODIAC (Tabla 6.3), resolvemos un nuevo modelo de programación lineal continua (6.32)-(6.40) en iguales condiciones de desequilibrio que las impuestas para el período anterior. La solución óptima, mostrada en la Tabla 6.13, tiene un valor inferior de la función objetivo (35.6). En otras palabras, el mix de rutas óptimo de este período da lugar a un número menor de movimientos intercelulares. En la Tabla 6.14 se presentan las cargas de trabajo de máquinas y de células.

Tabla 6.13. Rutas de procesado para el próximo período ($\alpha=0.25$, $\beta=0.15$)

Parte (i)	Mix de rutas		
	θ_{11}	θ_{12}	θ_{13}
1	0	0.548	0.452
2	0.113	0.887	-
3	0	0.353	0.647
4	0.207	0.793	-
5	0.802	0.198	-
6	0.312	0.688	-
7	-	-	-
8	-	-	-
9	-	-	-
10	0.258	0.742	-

Tabla 6.14. Cargas de trabajo para el próximo período ($\alpha=0.25$, $\beta=0.15$)

Célula	Máquina														Carga de célula	Carga media	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14			
1				149	126		171									446	149
2	149											180	198			527	176
3						151		112	134	112					148	657	131
4		180	144								146					470	157
Carga media de las células:																525	

Obsérvese que, aún para partes con requerimientos en ambos períodos (2, 3, 4, 5), la utilización de las rutas de procesado cambia al variar los requerimientos de producción de los períodos (véanse Tablas 6.8 y 6.13). Así por ejemplo, la parte 2 en el período actual presenta un mix de rutas de (0.544, 0.456), frente al mix del próximo período (0.113, 0.887). En otras palabras, dada una misma configuración de células, la utilización de las rutas por parte de las partes, varía de una forma dinámica con los requerimientos de producción de los distintos períodos.

6.5.2 COMPARACIÓN DE LA RNA CON OTRAS HEURÍSTICAS.

Con objeto de evaluar la calidad de las soluciones ofrecidas por la RNA, se ha resuelto el problema de agrupación de máquinas de un conjunto de problemas extraído de la literatura. En la Tabla 6.15 se presentan los problemas, incluyendo las referencias, sobre los que se han realizado las puebas.

Tabla 6.15. Problemas de la literatura

PROBLEMA	Nº DE MÁQUINAS	REFERENCIA	COMENTARIOS
CHA89A0	24	Chandrasekharan y Rajagopalan (1989)	Problema nº 1
CHA89B0	24	Chandrasekharan y Rajagopalan (1989)	Problema nº 2
CHA89C0	24	Chandrasekharan y Rajagopalan (1989)	Problema nº 3
CHA89E0	24	Chandrasekharan y Rajagopalan (1989)	Problema nº 5
CHA89F0	24	Chandrasekharan y Rajagopalan (1989)	Problema nº 6
CHA89G0	24	Chandrasekharan y Rajagopalan (1989)	Problema nº 7
BOC91A0	16	Boctor (1991)	Problema nº 1
BOC91B0	16	Boctor (1991)	Problema nº 2
BOC91C0	16	Boctor (1991)	Problema nº 3
BOC91D0	16	Boctor (1991)	Problema nº 4
BOC91E0	16	Boctor (1991)	Problema nº 5
BOC91F0	16	Boctor (1991)	Problema nº 6
BOC91G0	16	Boctor (1991)	Problema nº 7
BOC91H0	16	Boctor (1991)	Problema nº 8
BOC91I0	16	Boctor (1991)	Problema nº 9
BOC91J0	16	Boctor (1991)	Problema nº 10
SOF97A0	16	Sofianopoulou (1997)	

PROBLEMA	Nº DE MÁQUINAS	REFERENCIA	COMENTARIOS
SOF97B0	16	Sofianopoulou (1997)	
SOF97C0	15	Sofianopoulou (1997)	
SOF97D0	16	Sofianopoulou (1997)	
SOF97E0	16	Sofianopoulou (1997)	
SOF97F0	16	Sofianopoulou (1997)	
SOF97G0	16	Sofianopoulou (1997)	
SOF97H0	16	Sofianopoulou (1997)	
SOF97I0	16	Sofianopoulou (1997)	
SOF97J0	16	Sofianopoulou (1997)	

Los parámetros usados para la RNA son los siguientes:

$$\begin{aligned}
 \beta(0) &= 0.95 \\
 \beta(t + 1) &= 0.95\beta(t) \\
 C_{\max} &= 10 \\
 \eta(0) &= \text{int} \frac{C_{\max}}{2} \\
 \eta(t + 1) &= \eta(t) - 1
 \end{aligned}
 \tag{6.41}$$

En el caso del AECM, los parámetros usados son:

$$\begin{aligned}
 \text{temperatura inicial:} & \quad 0.1 \text{ F(W)} \\
 T_{\min}: & \quad 0.0001 \text{ F(W)} \\
 \alpha: & \quad 0.95
 \end{aligned}$$

donde W es la matriz de pesos obtenida al final de la RNA.

Como referencias de comparación de la RNA se han usado dos procedimientos heurísticos diseñados en Lozano et al (1997) para el mismo problema. El primero de ellos es un algoritmo del "Maximum Spanning Tree" (MST). El segundo es un procedimiento basado en técnicas de Búsqueda Tabú (BT). Toda la experimentación ha sido realizada con el parámetro p=0.1 sobre un ordenador Pentium PC a 100 Mhz bajo Windows 95. En la Tabla 6.16 se muestra, para cada uno de los problemas y cada uno de los tres procedimientos de solución, la función objetivo (FO), el número de células formadas (Nº CEL) y el tiempo de computación requerido (CPU).

Tabla 6.16. Resultados de la RNA, MST y BT ($p = 0.1$)

	MST			BT			RNA		
	FO	Nº CEL.	CPU	FO	Nº CEL.	CPU	FO	Nº CEL.	CPU
CHA89A0	29.700	7	2.25	29.700	7	9.54	29.700	7	8.73
CHA89B0	20.972	7	2.52	20.972	7	9.36	20.972	7	8.68
CHA89C0	13.020	8	2.63	13.587	7	6.35	13.587	7	9.94
CHA89E0	4.851	8	2.52	4.862	7	12.42	4.862	7	8.73
CHA89F0	2.759	9	3.57	3.034	7	6.59	3.034	7	7.63
CHA89G0	2.473	8	3.29	2.605	6	7.88	2.646	6	8.13
BOC91A0	9.261	2	0.65	9.261	2	2.74	9.261	2	2.30
BOC91B0	10.364	4	0.98	10.364	4	2.55	10.364	3	1.92
BOC91C0	9.703	4	1.20	9.703	4	4.02	9.703	4	2.30
BOC91D0	6.288	2	0.98	6.288	2	2.03	6.288	2	1.04
BOC91E0	11.685	4	0.82	11.685	4	2.36	11.685	4	2.08
BOC91F0	14.100	4	0.76	14.100	4	4.01	14.100	4	2.25
BOC91G0	14.218	4	0.87	14.218	4	2.51	14.218	4	2.30
BOC91H0	8.823	3	0.38	8.823	3	2.71	8.823	3	2.25
BOC91I0	15.170	4	0.98	15.170	4	2.35	15.170	4	2.30
BOC91J0	11.681	4	1.31	11.681	3	2.46	11.681	3	2.19
SOF97A0	7.593	2	0.38	7.593	2	2.15	7.593	2	1.12
SOF97B0	7.232	3	1.09	7.232	3	2.36	7.232	3	0.65
SOF97C0	7.664	2	0.44	7.664	2	2.26	7.664	2	1.86
SOF97D0	5.776	4	0.98	5.819	3	1.96	5.819	3	2.25
SOF97E0	7.912	3	1.20	7.912	3	2.63	7.912	3	1.70
SOF97F0	5.087	3	0.98	5.087	3	2.02	5.087	3	2.19
SOF97G0	6.844	2	0.76	6.844	2	2.02	6.844	2	0.70
SOF97H0	7.226	2	0.38	7.226	2	1.89	7.226	2	2.30
SOF97I0	7.177	3	1.15	7.202	2	2.23	7.202	2	1.92
SOF97J0	5.178	2	0.87	5.178	2	2.14	5.178	2	1.97

Observando los resultados de la Tabla 6.16, podemos comentar lo siguiente:

- Calidad de las soluciones: La RNA ofrece los mejores resultados junto a BT, excepto para un problema donde la RNA mejora a BT.
- Tiempo de computación: La RNA presenta en promedio unos tiempos de CPU comparables a los de BT aunque con una menor variabilidad en los mismos.

6.6 RESUMEN.

En este capítulo se ha abordado la problemática del diseño y carga de células en la presencia de planes de proceso alternativos para las distintas partes. La metodología seguida ha sido la de dividir el problema en dos etapas. En la primera etapa se determina

la configuración de las células. En la segunda etapa, dada la configuración de células, se cargan las máquinas que componen las mismas. La primera etapa se aborda una sólo vez a partir de la información de los planes de proceso alternativos de las partes a procesar en el taller. Por el contrario, la segunda etapa se aborda una vez para cada período de planificación.

Para la etapa de diseño se han presentado dos aproximaciones. En la primera de ellas, los planes de proceso alternativos se tratan como partes independientes. En la segunda aproximación se transforman los planes de proceso alternativos de cada una de las partes en un único plan de proceso difuso. Para la primera aproximación, se ha presentado una RNA que resuelve el problema de clusterización planteado. La resolución del problema de clusterización difuso se ha abordado con dos algoritmos. El problema de carga de la segunda etapa se ha modelado mediante programación lineal continua. A partir de los datos de un ejemplo, se ha ilustrado la metodología propuesta. Además, se han presentado diversas experiencias en las que se ha evaluado la calidad de las soluciones ofrecidas por la RNA, así como la funcionalidad del modelo de carga.

7.1 CONCLUSIONES Y APORTACIONES.

Este último capítulo contiene un resumen de los temas que se han abordado específicamente en cada uno de los capítulos del trabajo, resumiendo las aportaciones realizadas y las conclusiones alcanzadas. También contiene una descripción de las posibles líneas por donde puede evolucionar el mismo.

7.1.1 SISTEMAS DE FABRICACIÓN FLEXIBLE

Una vez descrito el escenario básico que conforman los SFF, así como las diversas problemáticas que surgen en el entorno de la planificación de la producción de los mismos, en este capítulo se hace un detallado análisis de las aproximaciones existentes en la literatura para la problemática de carga de los SFF. Las distintas aproximaciones se clasifican atendiendo a la problemática abordada, objetivo usado y características del modelo. El análisis crítico de las referencias permite identificar las limitaciones de las mismas.

7.1.2 PROBLEMAS DE CARGA Y SELECCIÓN DE ÍTEMS.

Teniendo en cuenta las limitaciones identificadas en el capítulo 1 al analizar las aproximaciones existentes, en el capítulo 2 se formula el problema de carga como uno de programación lineal entera mixta donde las variables de decisión son las referidas al mix agregado de rutas para cada uno de los ítems a procesar, y las correspondientes a la asignación de herramientas a máquinas. En cuanto a las restricciones, se tienen en cuenta las de los almacenes de herramientas de las máquinas, limitación del número de copias y del tiempo disponible de cada tipo de herramienta en el sistema. El criterio usado es el de equilibrar las cargas de trabajo de las máquinas. El modelo de carga es extendido para englobar, además de las anteriores, las decisiones referidas a los ítems a procesar conjuntamente. En esta segunda formulación, se usa como criterio la maximización de la suma de prioridades de los ítems seleccionados, mientras que el equilibrado de las cargas es modelado mediante restricciones sobre los desequilibrios de las cargas de las máquinas. Ambas formulaciones son ilustradas en el contexto de la planificación de la producción.

Las aportaciones fundamentales de las formulaciones son las siguientes:

Rutas de procesado.

- Identificación y uso de las rutas de procesado alternativas para los distintos ítems.
- Consideración explícita de la flexibilidad de rutas en la formulaciones, esto es, los ítems podrán ser procesados no únicamente por una ruta.

Asignación de herramientas.

- Consideración la posibilidad de asignar más de una copia de cada tipo de herramienta.
- Inclusión en la formulación de la vida útil, número de copias de cada tipo de herramienta y las limitaciones que imponen las mismas.

Decisiones.

- La asignación de operaciones no se hace individualmente, sino que se realiza de una forma conjunta a través del mix agregado de rutas para cada ítem.
- Tratamiento conjunto de las decisiones, esto es, asignación de operaciones y herramientas (y selección de ítems en el problema de selección) se tratan globalmente y no de forma aislada.

Funciones objetivo.

El modelado del equilibrado de la carga de trabajo de las máquinas, correspondiente a la función objetivo del problema de carga, se ha realizado tras un exhaustivo análisis de distintas formas aparentemente equivalentes. En el problema de selección de ítems se han tenido en cuenta además del objetivo del equilibrado, modelado mediante restricciones del modelo, el de la maximización de la suma de las prioridades de los ítems seleccionados.

7.1.3 SISTEMA DE COSTES PARA SISTEMAS DE FABRICACIÓN FLEXIBLE.

En el capítulo 3 se diseña un sistema ABC para la estimación de los costes totales de producción en los sistemas de fabricación flexible, donde los costes indirectos se imputan a los distintos ítems de acuerdo al consumo de los recursos de producción por parte de las actividades requeridas para el procesado de los mismos. El sistema de costes adapta de una forma dinámica los costes, a las distintas formas de operación, a través de su enlace con la fase de planificación de la producción en la que se usa una modificación de la formulación propuesta para el problema de selección de ítems. La funcionalidad del sistema de costes diseñado es ilustrada. Además, se presenta un detallado análisis de sensibilidad de los costes de producción frente a distintos parámetros de entrada.

El sistema de costes ABC diseñado para la estimación de los costes de producción es una aportación original.

7.1.4 SOLUCIONES HEURÍSTICAS.

En el capítulo 4 se aborda la resolución eficiente de la formulación de programación lineal entera mixta del modelo de carga. El enfoque propuesto es el de su resolución heurística. Inicialmente, se proponen dos procedimientos basados en relajaciones continuas del problema original. En ambos métodos la problemática global es descompuesta en dos etapas. En la primera etapa se fija una asignación de herramientas a máquinas, y en la segunda, se determina un mix agregado de rutas compatible con la misma. Para evaluar los procedimientos heurísticos se diseñan dos baterías de problemas generados aleatoriamente. Las soluciones ofrecidas por las heurísticas es comparada en calidad y tiempo con la obtenida a través de procedimientos estándares de ramificación y acotación. Los resultados muestran que, aún sin garantizar admisibilidad, los dos procedimientos ofrecen soluciones para un gran número de problemas de diversa dificultad, con muy bajos requerimientos computacionales.

Tanto los procedimientos heurísticos diseñados como el diseño de las dos baterías de problemas son completamente originales.

7.1.5 ESTRATEGIAS DE BÚSQUEDA EN LA VECINDAD.

En el capítulo 5 se diseñan los elementos necesarios para estrategias de búsqueda en la vecindad. A partir de una codificación entera de soluciones, y de una estrategia de oscilación en torno a la frontera de admisibilidad, introducida mediante una modificación en la evaluación de soluciones, se exploran distintas asignaciones de herramientas a máquinas. A continuación, los elementos anteriores se introducen en esquemas de Búsqueda Tabú para controlar la evolución entre soluciones. Se diseña un algoritmo Tabú que divide la búsqueda en dos fases. La primera fase consiste en buscar la primera solución admisible para el problema original. En ella se hacen uso de un mecanismo de diversificación. En la segunda fase, se hace una búsqueda intensiva para mejorar la calidad de las soluciones. El algoritmo diseñado se evalúa bajo un conjunto de problemas, mostrando los resultados que el método diseñado ofrece buenas soluciones

consistentemente para todos los problemas considerados. Además, se presenta un análisis de las soluciones ofrecidas por el algoritmo al variar la solución inicial. Una modificación sobre el algoritmo Tabú se presenta donde, la primera fase es implementada mediante un esquema basado en técnicas de Recocido Simulado con la idea de acelerar la obtención de la primera solución admisible. No obstante, los resultados muestran que esto se consigue a costa de alargar la segunda fase.

El diseño de la oscilación estratégica en torno a la frontera de admisibilidad que se introduce mediante una modificación en la evaluación en la función objetivo del problema original, es totalmente original. También es novedoso el empleo de modelos LP subordinados a la exploración del espacio de variables discretas. También es original el mecanismo de búsqueda aleatoria sesgada.

7.1.6 DISEÑO DE CARGA DE CÉLULAS DE FABRICACIÓN FLEXIBLE.

En el capítulo 6 se aborda el diseño y la carga de células de fabricación flexible. Se propone una metodología que descompone la problemática en dos etapas. En la primera etapa, se diseñan las células de producción. En la segunda etapa, se cargan las células y las máquinas que las componen. Para la etapa de diseño de las células se proponen dos aproximaciones. La primera está basada en considerar cada uno de los planes de proceso de las partes a procesar como una parte (virtual) independiente. Se diseña una Red Neuronal Auto-organizativa (RNA) para resolver el problema de clusterización planteado, comparando los resultados ofrecidos por la misma, con los de otros procedimientos heurísticos. Los resultados muestran que la RNA ofrece las mejores soluciones en unos tiempos de computación razonables. La segunda aproximación propuesta para la fase de diseño se basa en la construcción de un único plan de proceso difuso para cada parte, a partir de los planes alternativos de la misma. Se diseñan dos procedimientos de clusterización basado en el algoritmo de las C-medias. Para la etapa de carga de las células se propone un modelo de programación lineal continua que es una modificación del formulado para la carga de los SFF.

Las aportaciones originales realizadas en este capítulo son las siguientes:

- Metodología que descompone la problemática global en las fases de diseño y carga.
- Aproximación para el diseño de las células en la que cada plan de proceso se trata como una parte independiente (virtual).
- Diseño y aplicación de la RNA al problema de clusterización.
- Aproximación para el diseño de las células mediante planes de proceso difuso.
- Diseño y aplicación de dos modificaciones del algoritmo de la C-medias para el problema de clusterización.

7.2 POSIBLES EXTENSIONES.

La amplitud de los temas tratados en esta tesis sugiere profundizar en los mismos en varias direcciones. En primer lugar, parece conveniente la profundización en las aplicaciones de los modelos de planificación en entornos más globales donde no únicamente se estimen costes de producción a partir de metodologías ABC, sino que también se aborden otras decisiones. Así, el sistema de costes puede englobarse en un sistema para fijar los precios de los productos. Además, la fase de planificación puede ser realimentada a partir de los sistemas de costes y precios.

Por lo que respecta a los métodos de resolución del modelo de carga, se puede abordar su resolución a partir de la metodología de Lokketangen y Glover (1995). En esta línea, otra

posible línea de trabajo puede ser la de dotar a procedimientos estándares de ramificación y acotación de estructuras de memoria flexibles del tipo de las usadas en BT, de forma que se aproveche el conocimiento ofrecido por las soluciones exploradas. Otra posible aproximación para la resolución del modelo de carga puede ser el empleo de algoritmos genéticos. Debido a las muchas restricciones del modelo de carga, alguna de ellas podría ser relajada para así facilitar la evolución entre soluciones.

En lo referido al diseño de las células, será necesario la evaluación de los algoritmos difusos en diversos escenarios. Para ello, podría ser conveniente el diseño de problemas de distinto grado de dificultad. En lo referido a los algoritmos de clusterización, puede ser conveniente hibridizar la RNA con los algoritmos difusos, para de esta forma poder usar la misma en estos entornos.

I.1. INTRODUCCIÓN.

En este anexo se muestra el diseño de las dos baterías de problemas de carga usadas en los capítulos 4 y 5 de la tesis. Además, se modelan diversas formas de equilibrar las cargas de trabajo de las máquinas. Los distintos modelados de la función objetivo de equilibrar las cargas se evalúan a partir de la resolución de un conjunto de problemas de carga. Paralelamente, se muestran los efectos de diversos parámetros sobre el carácter equilibrable de los problemas.

I.2. PRIMER CONJUNTO DE PROBLEMAS.

En el diseño de la batería de problemas se han utilizado doce parámetros (P1,...,P12) que se muestran en la Tabla I.1. Dentro de los doce parámetros distinguiremos un subconjunto, a los que llamaremos parámetros fundamentales, que nos caracterizarán cada uno de los problemas generados. Cada parámetro fundamental describe distintos escenarios de un SFF. En concreto, los parámetros fundamentales son tres y los llamaremos PA (igual a P2), PB (igual a P7) y PC (igual a P10).

Tabla I.1. Parámetros para la generación de problemas (fase 1)

P1	Número de items	10 / 20 / 30
P2	Número máximo de rutas por item (Parámetro PA)	2, 5 / 10 / 15
P3	Número total de herramientas	20 / 30 / 40
P4	Número de estaciones	2 / 4 / 6
P5	Número máximo de máquinas por estación	1 / 2 / 3
P6	Número máximo de posiciones que ocupa una herramienta en una máquina	1 / 2 / 3
P7	Parámetro PB: $\frac{\sum_i D_i \max_r \left\{ \sum_j p_{irjk} \right\}}{L_k A_k}$	0.5 / 0.7 / 0.9
P8	Demanda de cada item	$\frac{200 \times n^\circ \text{ maximo de maquinas}}{n^\circ \text{ de items}} \pm 20\% / 40\% / 60\%$
P9	Tiempos de procesado	1, 10 / 20 / 30
P10	Parámetro PC: $\frac{\sum_k \left\{ \sum_i \max_r \left(\frac{D_i p_{irjk}}{L_k} \right) \right\} S_{jk}}{C_j}$	0.5 / 0.7 / 0.9
P11	Número de elementos de la matriz {S _{jk} } positivos	n° de herramientas × 1.2 / 1.5 / 1.8
P12	Unidades disponibles de cada tipo de herramienta (A _k)	$\text{Min. } A_k = \sum_{j \sum_i \sum_r p_{irjk} > 0} n^\circ \text{ de maquinas}$ $\text{Max. } A_k = 1.5 \times \text{Min. } A_k$

El parámetro PA representa el número máximo de rutas por ítem para cada problema, el número mínimo de rutas por ítem se ha tomado igual a 2. En la determinación de los datos de un problema, el parámetro PB actúa únicamente en el cálculo de los L_k , esto es, el aumento de PB se traduce en una disminución de la vida útil de las herramientas. El parámetro PC mide la "saturación" de la capacidad de los almacenes de herramientas de las máquinas. Como cada uno de los parámetros PA, PB y PC admite tres posibles definiciones y para cada combinación de PA, PB y PC se han generado tres problemas, el número total de problemas generados es 81. Los problemas se identifican de forma unívoca con cuatro dígitos. Los tres primeros codifican los valores especificados en los parámetros PA, PB Y PC respectivamente. El cuarto dígito codifica los tres posibles valores del parámetro P11. Fijadas las definiciones de PA, PB y PC de un problema, los datos a generar para el problema se extraen teniendo en cuenta las definiciones del resto de los parámetros que se muestran en la Tabla I.1.

I.3. SEGUNDO CONJUNTO DE PROBLEMAS.

La metodología seguida en el diseño de los problemas es similar a la seguida en el primer conjunto de problemas, aunque con algunos matices. En la Tabla I.2 se muestran los doce parámetros (P1,...,P12) usados en la generación. Dentro de los doce parámetros distinguiremos un subconjunto, a los que llamaremos parámetros fundamentales, que nos caracterizarán cada uno de los problemas generados. Cada parámetro fundamental describe distintos escenarios de los SFF. En concreto, los parámetros fundamentales son cuatro y los llamaremos PA (igual a P2), PB (igual a P7), PC (igual a P10) y PD (igual a P11).

Tabla I.2. Parámetros para la generación de problemas (fase 2)

P1	Número de ítems	10 / 15/ 20
P2	Parámetro PA: Número máximo de rutas por ítem	2, 4/ 8/ 12
P3	Número total de herramientas	20 / 25 / 30
P4	Número de estaciones	2 / 4 / 6
P5	Número de máquinas por estación (m_j)	1,1 / 2 / 3
P6	Número de posiciones que ocupa una herramienta en una máquina (S_{jk})	1,1 / 2 / 3
P7	Parámetro PB: $\frac{\sum_i D_i \text{ med } \left\{ \sum_j p_{irjk} \right\}}{L_k A_k}$	0.5 / 0.7 / 0.9
P8	Demanda de cada ítem	5 ÷ 15
P9	Tiempos de procesado	1, 5/ 15 / 25
P10	Parámetro PC: $\frac{\sum_k \left\{ \sum_i \text{ med } \left(\frac{D_i p_{irjk}}{L_k} \right) \right\} S_{jk}}{C_j}$	0.5 / 0.7 / 0.9
P11	Número de operaciones por ruta (Parámetro PD)	{(nest x nherr) x parámetro PD} x (40÷70 %) PD= 0.6/0.7/0.8
P12	Duración de cada tipo de herramienta (L_k)	220 ÷ 300

A continuación se hace una descripción de cada uno de los parámetros fundamentales:

Parámetro A (PA): Se corresponde con P2. Representa el número máximo de rutas por ítem para cada problema. Se ha considerado un número mínimo de rutas por ítem igual a 2. Dado un problema, el PA tomará los valores 4, 8 ó 12, de manera que dicho problema vendrá caracterizado por alguno de los tres valores anteriores. El número de rutas para los ítems del problema será un número entero aleatorio entre 2 y el valor de PA. Para un problema dado, los distintos ítems tendrán en general distinto número de rutas.

Parámetro B (PB): Se corresponde con P7. El PB es el cociente entre el tiempo requerido por una herramienta si los ítems se procesan por una ruta promedio y el tiempo disponible de la misma. Para un problema dado, el PB se aplica a todas las herramientas del mismo. El parámetro PB únicamente tiene como función el cálculo del número de unidades disponibles de cada tipo de herramienta (A_k), tal y como se verá más adelante.

Parámetro C (PC): Se corresponde con P10. El parámetro PC mide la "saturación" de la capacidad de los almacenes de herramientas de las máquinas. El PC es el cociente entre el número medio de posiciones requeridas por las distintas herramientas, en cada una de las máquinas de una estación, y el número de posiciones disponible C_j . Este parámetro se aplicará a las distintas estaciones de un problema.

Parámetro D (PD): Corresponde con P11. El parámetro PD mide la densidad total de la matriz de tiempos de las distintas rutas. Fijado cualquier ítem y ruta, el número total de posibles operaciones por ruta vendrá dado por $\{ (n^\circ \text{ de estaciones} \times n^\circ \text{ de herramientas}) \times PD \}$.

El número real de operaciones por ruta se obtiene de aplicar, para todo ítem y ruta, un porcentaje aleatorio (40%÷70%) al número total de posibles operaciones descrito anteriormente:

$$\{ (n^\circ \text{ de estaciones} \times n^\circ \text{ de herramientas}) \times PD \} \times (0.4 \div 0.7)$$

Un incremento en el parámetro PD supondría un mayor requerimiento de las herramientas por parte de las máquinas. Por lo cual, se puede decir que PD mide la interrelación máquina-herramienta.

Como cada uno de los parámetros fundamentales admite tres posibles definiciones y para cada combinación de PA, PB, PC y PD se ha generado un problema, el número total de problemas generados es 81. Cada problema se identifica de forma unívoca a partir de 4 dígitos que representan los posibles valores de los parámetros PA, PB, PC y PD respectivamente.

Fijadas las definiciones de los parámetros fundamentales de un problema, los datos a generar para el problema se extraen teniendo en cuenta las definiciones del resto de los parámetros que se muestran en la Tabla 30.

Veamos el procedimiento seguido:

P1. N° de ítems: Se toma aleatoriamente 10, 15 ó 20.

P3. N° de herramientas: El número de herramientas se elige aleatoriamente entre 20, 25 ó 30.

P4. N° de estaciones: El número de estaciones se toma aleatoriamente entre 2, 4 ó 6.

P5. N° de máquinas por estación: Podrá ser igual a un número aleatorio entero comprendido entre 1 y los valores máximos 1, 2 ó 3. La elección de dichos máximos también se realiza de forma aleatoria y se aplica a todo el problema. De esta manera queda fijada m_j .

P6. N° de posiciones que ocupa una herramienta en una máquina: Podrá ser igual a un número aleatorio entero comprendido entre 1 y los valores máximos 1, 2 ó 3. La elección de los valores máximos también se realiza de forma aleatoria, y una vez elegido, se aplica a todo el problema. De esta manera queda fijada la matriz $\{S_{jk}\}$.

P8. Demanda: La demanda de los distintos ítems serán enteros aleatorios comprendidos en el intervalo $5 \div 15$.

P9. Tiempos de procesado: Los tiempos de procesado son valores enteros aleatorios comprendidos entre la unidad y los valores máximos 5, 15 ó 25. Dado un problema el tiempo máximo se elige aleatoriamente, y se aplica a todo el problema.

Dado un ítem y una ruta, se fija el número de operaciones para dicha ruta mediante el procedimiento mostrado en la descripción del parámetro PD. Luego, el número de tiempos de procesado que se hacen positivos $p_{irjk} > 0$ en dicha ruta, no puede exceder al número de operaciones establecido.

Así, una vez conocido el número de operaciones para una ruta de un ítem, se van eligiendo aleatoriamente pares estación-herramienta (j,k) y se les asigna un tiempo de procesado positivo ($p_{irjk} > 0$) hasta que se completa el número de operaciones permitido.

P12. Duración de cada tipo de herramienta: La duración de cada copia de un tipo de herramienta (L_k) es un valor entero aleatorio comprendido en el intervalo $200 \div 300$.

Unidades disponibles de cada tipo de herramienta: Del PB (ó P7) se despeja el denominador, es decir, obtendremos que el producto de L_k y A_k es igual a una determinada constante. A partir de esta constante y con el valor asignado aleatoriamente a L_k , tal y como se ha descrito anteriormente en (P12), se obtiene el valor que toma A_k .

$$\frac{\sum_i D_i \text{ med}_r \left\{ \sum_j p_{irjk} \right\}}{L_k A_k} = \text{PB} \Rightarrow A_k = \frac{\sum_i D_i \text{ med}_r \left\{ \sum_j p_{irjk} \right\}}{L_k \text{PB}}$$

En la determinación de los datos de un problema, el parámetro PB actúa únicamente en el cálculo de A_k , esto es, el aumento de PB se traduce en una disminución en el número de copias disponibles de las herramientas.

Número de posiciones del almacén de herramientas de cada una de las máquinas en cada estación: Se obtiene despejando de PC (ó P10) para cada una de las estaciones.

En la determinación de los datos de un problema, el parámetro PC actúa únicamente en el cálculo de C_j , esto es, el aumento de PC se traduce en una disminución del espacio disponible en el almacén de herramientas de las máquinas de la estación j correspondiente,

$$\frac{\sum_k \left\{ \sum_i \text{ med}_r \left(\frac{D_i p_{irjk}}{L_k} \right) \right\} S_{jk}}{C_j} = \text{PC} \Rightarrow C_j = \frac{\sum_k \left\{ \sum_i \text{ med}_r \left(\frac{D_i p_{irjk}}{L_k} \right) \right\} S_{jk}}{\text{PC}}$$

I.4. MODELADO DEL EQUILIBRADO DE LAS CARGAS.

En este apartado se presentan 8 formas aparentemente equivalentes de modelar el objetivo de equilibrar las cargas de trabajo de las máquinas. En la Tabla I.3, aparecen las ocho funciones objetivo, indicando, además de su significado, las restricciones que definen variables adicionales, en el caso de que sean necesarias.

Tabla I.3. Modelado del objetivo de equilibrar la carga de trabajo de las máquinas

M1	Min. u_{\max}	$u_j \leq u_{\max} \quad \forall j$	minimizar la carga de trabajo de la estación más cargada
M2	Max. u_{\min}	$u_{\min} \leq u_j \quad \forall j$	maximizar la carga de trabajo de la estación menos cargada
M3	Min. $u_{\max} - u_{\min}$	$u_{\min} \leq u_j \leq u_{\max} \quad \forall j$	minimizar la diferencia de carga entre las estaciones más y menos cargada
M4	Min. $\sum_j (u_{\max} - u_j)$	$u_j \leq u_{\max} \quad \forall j$	minimizar la suma de desviaciones de carga medidas respecto a la estación más cargada
M5	Min. $\sum_j (u_j - u_{\min})$	$u_{\min} \leq u_j \quad \forall j$	minimizar la suma de desviaciones de carga medidas respecto a la estación menos cargada
M6	Min. $\max_j u_j - u_{\text{medio}} $	$u_{\text{med}} = \frac{\sum_{j=1}^M u_j}{M}$	minimizar la máxima desviación (medida en valor absoluto) respecto a la carga media
M7	Min. $\sum_j u_j - u_{\text{medio}} $	$u_{\text{med}} = \frac{\sum_{j=1}^M u_j}{M}$	minimizar la suma de desviaciones (medidas en valor absoluto) respecto a la carga media
M8	Min. $\sum_{i=1}^{M-1} \sum_{j=i+1}^M u_i - u_j $		minimizar la suma de desviaciones (medidas en valor absoluto) entre la carga de cada par de estaciones

I.5. EVALUACIÓN DE LAS FUNCIONES OBJETIVO.

El objetivo que nos planteamos es evaluar las 8 formas de equilibrar presentadas en el apartado anterior. Para ello, se construyen los problemas de carga (siguiendo lo descrito en el capítulo 2), para cada uno de los 81 problemas del primer conjunto (apartado I.2), con cada una de las 8 funciones objetivo mostradas en la Tabla I.3. Cada modelo de programación lineal entera mixta se resuelve pseudóptimamente. Cuando hablamos de "resolución pseudóptima" queremos decir lo siguiente:

1. Cada programa lineal ha sido resuelto con el procedimiento llamado "optimalidad- ϵ ", véase Garfinkel (1979), que garantiza que la solución "óptima" obtenida al terminar estará dentro de un $\epsilon\%$ del verdadero óptimo. En esta experimentación se ha tomado $\epsilon\%=1\%$. Aunque no se garantiza la solución óptima, se ahorra gran cantidad de tiempo de computación y se acota el error.
2. Se ha tomado en el proceso de optimización de cada programa lineal un límite máximo de 2.5 millones de iteraciones. Alcanzado este límite, la resolución se detiene ofreciendo como solución la mejor hasta ese momento.

Una vez resuelto cada programa lineal, y partiendo de las soluciones "óptimas" obtenidas

con cada uno de los 8 modelos, se ha determinado el carácter equilibrable y desequilibrado de los 81 problemas generados. En concreto, de los 81 problemas, 52 admiten solución totalmente equilibrada, frente a 29 desequilibrados. En la Figura I.1 se muestra el Número de Problemas Equilibrados (NPE) en los distintos escenarios caracterizados por los parámetros fundamentales (PA, PB y PC).

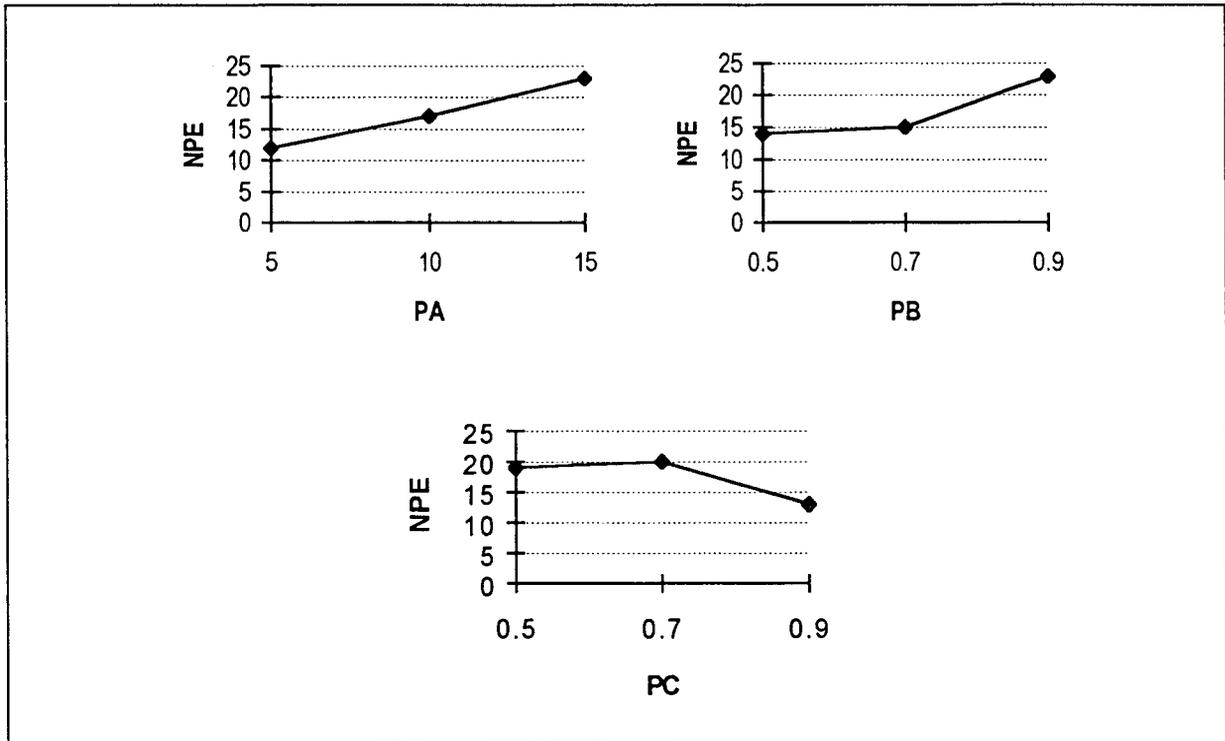


Figura I.1. Variación del Número de Problemas Equilibrables (NPE) con los parámetros fundamentales

En la Figura I.1 se pueden observar tres tendencias claramente definidas: (1) al aumentar PA aumenta NPE, es decir, la disponibilidad por parte de los items de más rutas, hace que sea más probable el equilibrado de las cargas, (2) al aumentar PB aumenta NPE, es decir, la utilización de herramientas con menor vida útil aumenta la probabilidad de equilibrado y (3) al aumentar PC disminuye NPE, es decir, la disminución de la capacidad de los almacenes de herramientas de las máquinas hace que sea más difícil (menos probable) el equilibrado de las cargas.

En la Tabla I.4 se presentan el número de problemas equilibrados con cada modelo (NPE-M1,...,NPE-M8) de los 52 que admiten solución equilibrada. Estos resultados confirman la inferioridad de las funciones objetivo M1 y M2 frente al resto, las cuales presentan unos resultados uniformes.

Tabla I.4. Número de problemas equilibrados con cada función objetivo

NPE-M1	NPE-M2	NPE-M3	NPE-M4	NPE-M5	NPE-M6	NPE-M7	NPE-M8
35	27	48	46	46	47	50	46

Para comparar las distintas funciones objetivo en el entorno de problemas desequilibrados, se ha tomado como referencia para medir el desequilibrio el Coeficiente de Variación (CV) de la distribución de la carga entre las máquinas de las estaciones. En la Tabla I.5 se presenta el CV medio con cada uno de los ocho modelos para los 29 problemas

desequilibrados.

Tabla I.5. Coeficiente de Variación medio con cada función objetivo

CV-M1	CV-M2	CV-M3	CV-M4	CV-M5	CV-M6	CV-M7	CV-M8
0.289	0.338	0.185	0.223	0.234	0.218	0.192	0.201

De las funciones objetivo sin deficiencias (M3,...,M8), las que presentan resultados más prometedores son M3 y M7, en tanto que presentan menor CV medio. Obsérvese, que M3 da ligeramente mejores resultados que M7, a pesar de que sólo involucra a las estaciones más y menos cargadas. Por el contrario, M7 tiene en cuenta la carga de todas las estaciones, comparándolas con la carga media.

Con el objeto de comparar los CV de los distintos modelos se han calculado los Coeficientes de Variación Relativos (CVR). En cada uno de los 29 problemas disequilibrados se toma como CV de referencia el menor de los CV proporcionado por los 8 modelos. Para un problema, los CVR de los distintos modelos son el cociente entre CV y el CV de referencia del problema. En la Tabla 6 se muestran los CVR medios de cada modelo en los 29 problemas disequilibrados. Además se muestra el número de problemas en los que el CV del modelo correspondiente es el mínimo de los CV de los 8 modelos. Obsérvese que M3 es el que presenta menor CVR medio, además de ser el modelo que más veces ofrece disequilibrio mínimo.

Tabla I.6. CVR medio con cada FO y número de veces que el CV es el mínimo

CVR-M1	CVR-M2	CVR-M3	CVR-M4	CVR-M5	CVR-M6	CVR-M7	CVR-M8
4.690	7.393	1.682	2.902	3.323	1.952	1.756	2.311
0	0	13	7	4	5	9	11

En la Figura I.2 se muestra la variación del CVR de M3 con los parámetros fundamentales.

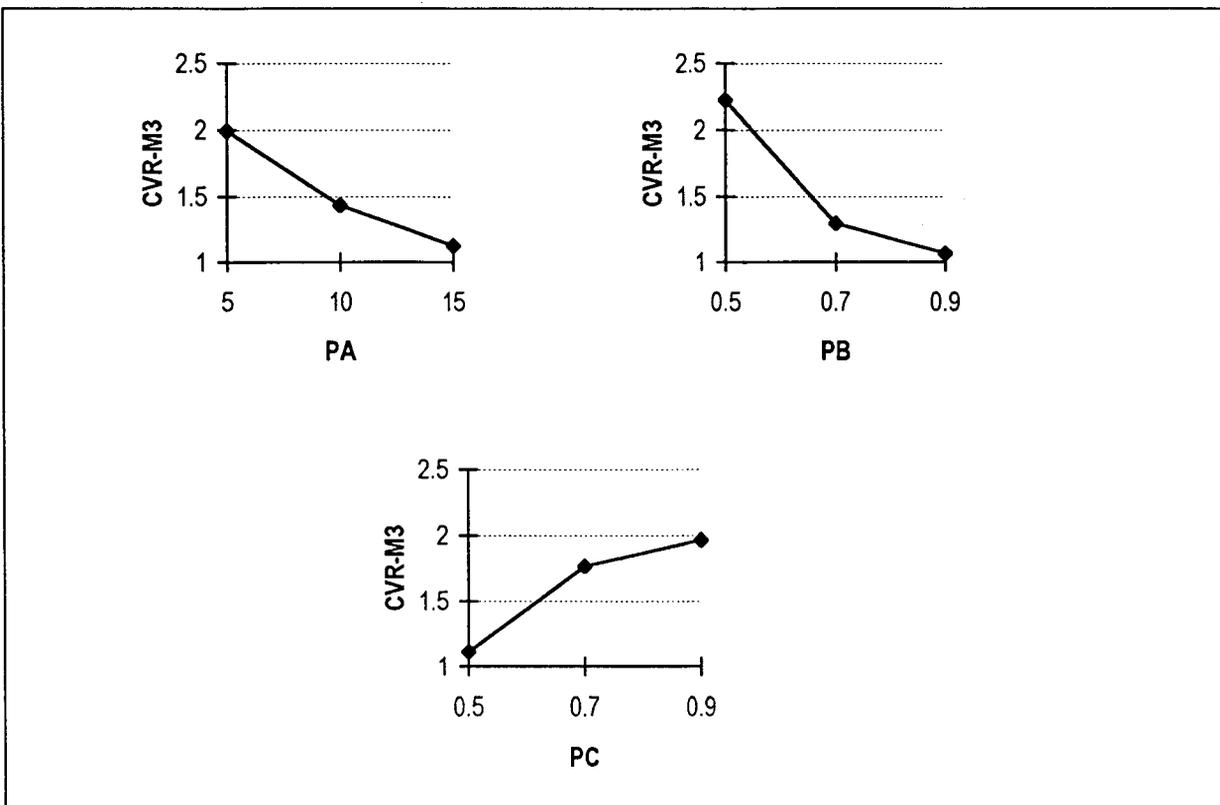


Figura I.2. Variación del CVR de M3 con los parámetros fundamentales

Obsérvese que PA, PB y PC influyen en la misma dirección tanto en el NPE como en el CVR. Esto es, al aumentar PA ó PB, el problema es más probable que sea equilibrado, y en el caso de que sea desequilibrado, el desequilibrio de los modelos se acerca más al mínimo. Por el contrario, al aumentar PC es menos probable el equilibrado, y en los problemas desequilibrados, el desequilibrio de los modelos se aleja del mínimo.

Para argumentar la idea de que para el equilibrado de la carga puede ser conveniente la utilización por los items de más de una ruta, de la solución dada para cada problema por cada uno de los modelos se ha calculado el porcentaje de items con ruta múltiple (IRM). En la Tabla 1.7 se presenta el IRM de las soluciones proporcionadas por los distintos modelos, tanto de forma agregada para los 81 problemas (TOTAL), como desagregada para los 52 y 29 problemas equilibrables (PEQ) y desequilibrados (PDQ) respectivamente. Además, para cada modelo se presenta el número máximo de rutas utilizado por un ítem (NM) en los PEQ y PDQ.

Tabla 1.7. IRM y NM en las ocho funciones objetivo

	M1		M2		M3		M4		M5		M6		M7		M8	
	IRM	NM														
PEQ	11.2	4	15.5	3	17.8	3	12.3	5	17.2	4	15.8	5	17	4	17.7	4
PDQ	8.8	2	11	3	11.7	3	11.7	3	9.8	3	11.8	3	12.4	3	12.7	3
TOTAL	10.3	-	13.9	-	15.6	-	14.6	-	14.5	-	14.4	-	15.5	-	15.9	-

Se observa lo siguiente: (1) el IRM en los distintos modelos es mayor en PEQ que en PDQ y (2) el NM en los distintos modelos en los PEQ es siempre mayor o igual que el correspondiente a los PDQ. Parece inferirse, por tanto, que para el equilibrado puede ser conveniente la utilización de más de una ruta por algunos ítems.

Los experimentos han sido realizados en un ordenador IBM-PC 486 DX2 a 66 Mhz. A continuación se muestra el tiempo de CPU (en segundos) del modelo M3 (en escala logarítmica) desdoblado para los PEQ y PDQ, en función del número de variables binarias (VB), las cuales se utilizan en el paquete informático XA para codificar las variables enteras. Además, se muestra el ajuste potencial realizado tanto para los PEQ como para los PDQ. A partir del ajuste se puede decir que los PDQ son más difíciles que los PEQ en tanto que requieren mayor tiempo de computación.

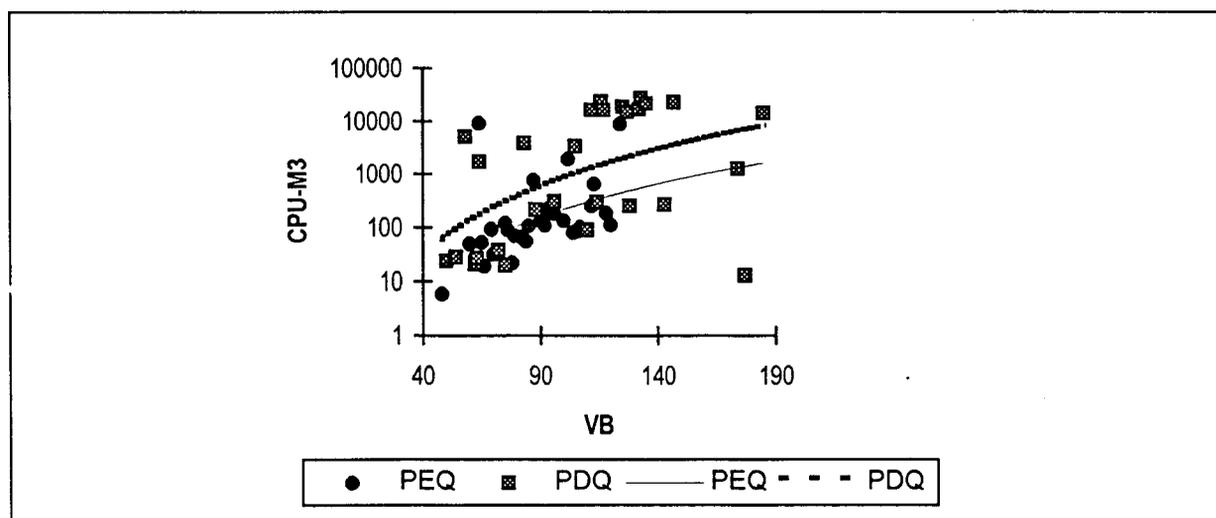


Figura 1.3. Variación del tiempo de CPU de M3 con el número de variables binarias

II.1. INTRODUCCIÓN.

En este anexo se las características básicas de la aplicación informática que implementa el sistema de coste diseñado en el capítulo 3..

II.2. IMPLEMENTACIÓN.

Una vez diseñados cada uno de los elementos del sistema de costes, el siguiente paso será el diseño e implementación de una aplicación informática que realice cada una de las operaciones requeridas. En la Figura II.1 se muestra el flujo de datos de la aplicación realizada.

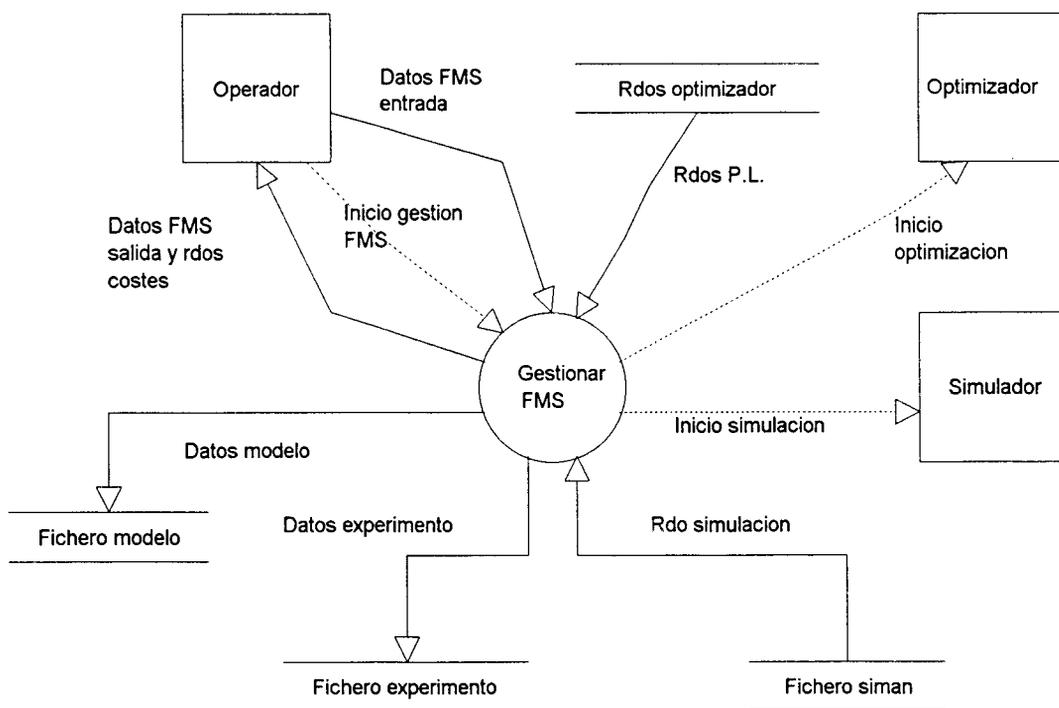


Figura II.1. Diagrama de Contexto de la aplicación

La aplicación consta de tres módulos externos: (1) operador, (2) optimizador y (3) simulador.

1. Operador: Es el encargado de iniciar la sesión suministrando los datos del SFF (pedidos, instalaciones del sistema de producción, etc), los cuales estarán almacenados en bases de datos relacionales.
2. Optimizador (XA Callable Library(1994)): A partir de los datos del SFF, la aplicación construye en estructuras de memoria el programa lineal del modelo de planificación. Una vez resuelto, las salidas (pedidos seleccionados, mix agregado de rutas para cada pedido, asignación de herramientas a máquinas y carga de máquinas) son almacenadas en "Rdos. optimizador".
3. Simulador (SIMAN V.4 (1989)): Su función es realizar la simulación de la operación del taller. El modelo de simulación es implementado en lenguaje SIMAN. Para ello, la

aplicación construye de forma transparente al usuario el modelo ("Fichero modelo") y experimento ("Fichero experimento") de entrada al simulador externo. Posteriormente, modelo y experimento se compilan y enlazan, dando lugar al "fichero Siman". Por último, se realiza la simulación, capturando la aplicación las salidas de interés de la misma (número de piezas de cada pedido que va por cada una de sus rutas y tiempos de finalización de cada pedido y distancias recorridas por los transportes para las piezas de cada pedido).

La aplicación ha sido implementada mediante Visual Basic bajo entorno Windows. A continuación se muestran algunas pantallas de la misma para ilustrar las características del interfase de la misma.

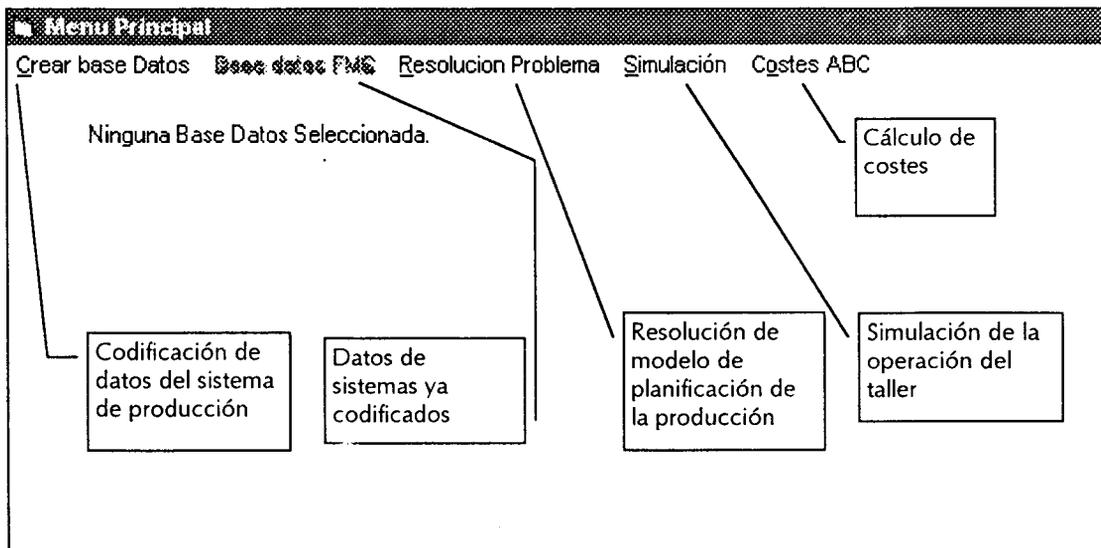


Figura II.2. Menú principal de la aplicación

■ Porcentajes por Pedido				
Pedido	Y	Ruta1	Ruta2	Ruta3
1	0	0	0	0
2	0	0	0	0
3	0	0	0	0
4	1	0,7499	0,2500	0
5	0	0	0	0
6	1	0,6874	0	0,3125
7	1	0	0,4062	0,5937
8	1	3,5714	0,9642	0
9	1	0,1000	0,5499	0,3500
10	0	0	0	0
11	1	0	0	1

Figura II.3. Salida de optimizador: pedidos seleccionados y mix de rutas

■ Cargas de Estaciones							
	Herramier	Herramier	Herramier	Herramier	Herramier	Herramier	Carga
Estacion 1	3	2	1	2	0		1040
Estacion 2	0	1	2	1	2		948,7500
Carga							994

Figura II.4. Salida de optimizador: carga de máquinas

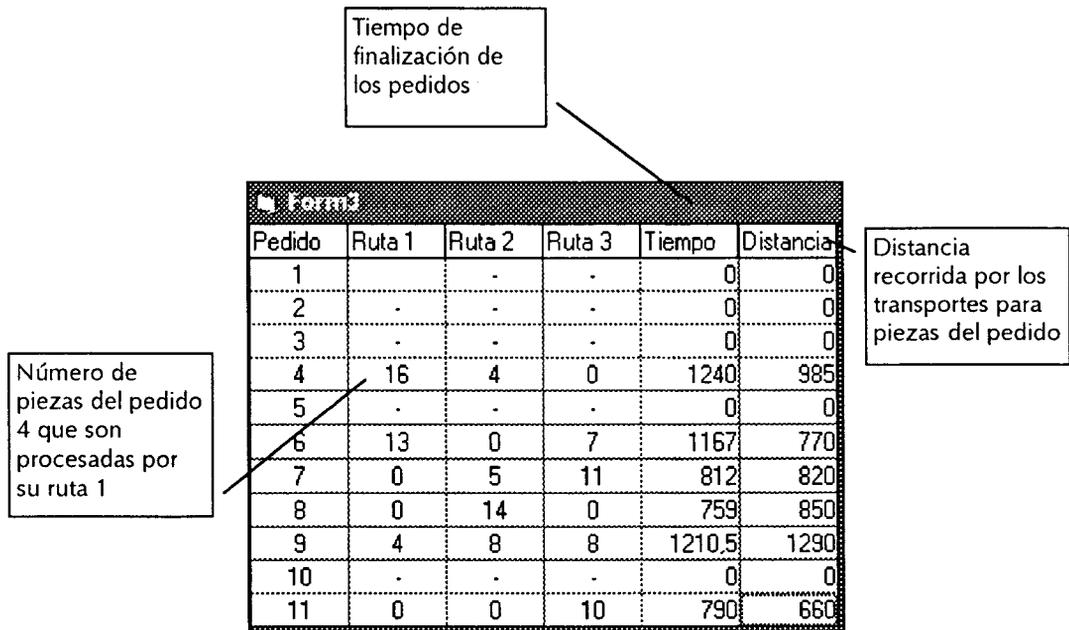


Figura II.5. Salida de simulador

REFERENCIAS.

- ABOUDI, R., y JÖRNSTEN, K., 1994, Tabu Search for General Zero-One Integer Programs Using the Pivot and Complement Heuristic. *ORSA Journal on Computing*, **6**, 82-93.
- ARBIB, C., LUCERTINI, M., y NICOLO, F., 1990, Workload Balance and Part-Transfer Minimization in Flexible Manufacturing Systems. *The International Journal of Flexible Manufacturing Systems*, **3**, 5-25.
- ATAN, T.S., y PANDIT, R., 1996, Auxiliary Tool Allocation in Flexible Manufacturing Systems. *European Journal of Operational Research*, **89**, 642-659.
- BASNET, C., 1996, Technical note: Tabu search heuristic for a loading problem in flexible manufacturing systems. *International Journal of Production Research*, **34**, 1171-1174.
- BASTOS, J.M., 1988, Batching and routing: Two functions in the operational planning of flexible manufacturing systems. *European Journal of Operational Research*, **33**, 230-244.
- BERRADA, M., y STECKE, K.E., 1986, A branch and bound approach for machine load balancing in flexible manufacturing systems. *Management Science*, **32**, 1316-1335.
- BEZDEK, J.C., 1981, *Pattern Recognition with Fuzzy Objective Function Algorithms* (New York: Plenum Press).
- BROWNE, J., DUBOIS, D., RATHMILL, K., SETHI, S.P., y STECKE, K.E., 1984, Classification of flexible manufacturing systems. *The FMS Magazine*, April, 114-117.
- BUZACOTT, J.A., y YAO, D.D., 1986, Flexible manufacturing systems: a review of models. *Management Science*, **32**, 890-905.
- CERNY, V., 1985, Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm. *Journal of Optimization Theory and Applications*, **45**, 41-52.
- CHAKRAVARTY, A.K., y SHTUB, A., 1984, Selecting parts and loading flexible manufacturing systems. *Proceedings of the First ORSA/TIMS Special Conference on Flexible Manufacturing Systems*, Ann Arbor, Michigan, pp. 284-289.
- CHANDRASEKHARAN, M.P., y RAJAGOPALAN, R. 1987, ZODIAC—an algorithm for concurrent formation of part-families and machine cells. *International Journal of Production Research*, **25**, 835-850.
- CHENG, C.H., KUMAR, A. y MOTWANI, J., 1995, A comparative examination of selected cellular manufacturing clustering algorithms. *International Journal of Operations and Production Management*, **15**, 86-97.
- CHEN, I.J., y CHUNG, C-H., 1991, Effects of loading and routing decisions on performance of flexible manufacturing systems. *International Journal of Production Research*, **29**, 2209-2225.
- CHEN, I.J., y CHUNG, C-S., 1996, Sequential Modelling of the Planning and Scheduling Problems of Flexible Manufacturing Systems. *Journal of the Operational Research Society*, **47**, 1216-1227.
- CHOOBINEH, F., 1988, A framework for the design of cellular manufacturing systems. *International Journal of Production Research*, **26**, 1161-1172.
- CHU, C-H., y HAYYA, J.C. 1991, A fuzzy clustering approach to manufacturing cell formation. *International Journal of Production Research*, **29**, 1475-1487.
- COOPER, R., 1988, The rise of activity-based costing-Part two: When do I need an activity

- based cost system. *Cost Management*, **Fall**, 41-48.
- COOPER, R., 1989, The rise of activity-based costing-Part four: What do activity based cost system look like. *Cost Management*, **Spring**, 38-49.
- D'ALFONSO, T.H., y VENTURA, J.A., 1995, Assignment of tools to machines in a flexible manufacturing system. *European Journal of Operational Research*, **81**, 115-133.
- DE WERRA, D., y WIDMER, M., 1990, Loading Problems with Tool Management in Flexible Manufacturing Systems: A Few Integer Programming Models. *The International Journal of Flexible Manufacturing Systems*, **3**, 71-82.
- DHAVALE, D., 1992, Activity-based costing in cellular manufacturing systems. *Industrial Engineering*, **February**, 44-46.
- FLYNN, B.B., y JACOBS, F.R., 1986, A simulation comparison of group technology with traditional job shop manufacturing. *International Journal of Production Research*, **24**, 1171-1192.
- GARFINKEL, R.S. (1979), Branch and Bound Methods for Integer Programming, *Combinatorial Optimization*, editado por Christofides, Mingozzi, Toth and Dand, John Wiley & Sons, 1-20.
- GAUDIOSO, M., y LEGATO, P., 1991, Linear programming models for load balancing. *Computers and Operations Research*, **18**, 59-64.
- GLOVER, F., 1986, Future paths for integer programming and links to artificial intelligence. *Computers and Operations Research*, **13**, 533-549.
- GLOVER, F., 1990, Tabu Search: A tutorial. *Interfaces*, **20**, 74-94.
- GLOVER, F., and GREENBERG, H., 1989, New approaches for heuristic search: A bilateral linkage with artificial intelligence. *European Journal of Operational Research*, **39**, 119-130.
- GLOVER, F., y LAGUNA, M., 1995, Tabu Search, *Modern Heuristic Techniques for Combinatorial Problems*, editado por C.R. Reeves (London: McGraw-Hill) pp. 70-150.
- GLOVER, F., y LAGUNA, M., 1996, Tabu Search, (Kluwer Academic Press).
- GREENE, T.J., y SADOWSKI, R.P., 1986, A mixed integer program for loading and scheduling multiple flexible manufacturing cells. *European Journal of Operational Research*, **24**, 379-386.
- GUNASEKARAN, A., MARTIKAINEN, T., y YLI-OLLI, P., 1993, Flexible manufacturing systems: An investigation for research and applications. *European Journal of Operational Research*, **66**, 1-26.
- HELD, M., WOLFE, P., y CROWDER, H.D., 1974, Validation of Subgradient Optimization. *Mathematical Programming*, **6**, 62-88.
- HINDI, K.S., 1996, Solving the CLSP by a Tabu Search Heuristic, *Journal of the Operational Research Society*, **47**, 151-161.
- HWANG, S.S., y SHOGAN, A.W., 1989, Modelling and solving an FMS part selection problem. *International Journal of Production Research*, **27**, 1349-1366.
- JAIKUMAR, R., y VAN WASSENHOVE, L.N., 1989, A Production Planning Framework for Flexible Manufacturing Systems. *Journal of Manufacturing and Operations Management*, **2**, 52-79.
- JENSEN, J.B., MALHOTRA, M.K., y PHILIPOOM, P.R., 1996, Machine dedication and process

- flexibility in a group technology environment. *Journal of Operations Management*, **14**, 19-39.
- JOHNSON, T.H., 1991, Activity-based management: past, present, and future. *The Engineering Economist*, **36**, 219-238.
- KANG, S.L., y WEMMERLÖV, U., 1993, A work load-oriented heuristic methodology for manufacturing cell formation allowing reallocation of operations. *European Journal of Operational Research*, **69**, 292-311.
- KAPLAN, R.S., 1991, New systems for measurement and control. *The Engineering Economist*, **36**, 201-218.
- KARKOWSKY, W., y EVANS, G.W., 1986, Fuzzy concepts in production management research: a review. *International Journal of Production Research*, **24**, 129-147.
- KIM, Y-D, y YANO, C.A., 1993, Heuristic approaches for loading problems in flexible manufacturing systems. *IIE Transactions*, **25**, 26-39.
- KIM, Y-D, y YANO, C.A., 1994, A New Branch and Bound Algorithm for Loading Problems in Flexible Manufacturing Systems. *The International Journal of Flexible Manufacturing Systems*, **6**, 361-382.
- KIRKAVAK, N., y DINÇER, C., 1993, Analytical loading models in flexible manufacturing systems. *European Journal of Operational Research*, **71**, 17-31.
- KIRPATRICK, S., GELATT, C.D. y VECCHI, M.P., 1983, Optimization by simulated annealing. *Science*, **220**, 671-680.
- KOUVELIS, P., y LEE, C., 1991, Block angular structures and the loading problem in flexible manufacturing systems. *Operations Research*, **39**, 666-676.
- KOUVELIS, P., 1992, Design and planning problems in flexible manufacturing systems: a critical review. *Journal of Intelligent Manufacturing*, **3**, 5-99.
- KUSIAK, A., 1987, The generalized group technology concept. *International Journal of Production Research*, **25**, 561-569.
- LAGUNA, M., 1994, Clustering for the Design of SONET Rings in Interoffice Telecommunications, *Management Science*, **40**, 1533-1541.
- LASHKARI, R.S., DUTTA, S.P., y PADHYE, A.M., 1987, A new formulation of operation allocation problem in flexible manufacturing systems: mathematical modelling and computational experience. *International Journal of Production Research*, **25**, 1267-1283.
- LIANG, M., y DUTTA, S.P., 1990, A mixed-integer programming approach to the machine loading and process planning problem in a process layout environment. *International Journal of Production Research*, **28**, 1471-1484.
- LIANG, M., y DUTTA, S.P., 1993, An integrated approach to the part selection and machine loading problem in a class of flexible manufacturing systems. *European Journal of Operational Research*, **67**, 387-404.
- LOGENDRAN, R., RAMAKRISHNA, P., y SRISKANDARAJAH, C., 1994, Tabu search-based heuristics for cellular manufacturing systems in the presence of alternative process plans. *International Journal of Production Research*, **32**, 273-297.
- LOKKETANGEN, A., y GLOVER, F., 1995, Tabu search for zero-one mixed integer programming with advanced level strategies and learning. *International Journal of Operations and Quantitative Management*, **1**, 89-109.

- LOKKETANGEN, A., y GLOVER, F., 1996, Probabilistic move selection in tabu search for zero-one mixed integer programming problems. *Meta-heuristics: Theory and Applications*, editado por Kelly y Osman (Kluwer Academic Press)
- LOZANO, S., ADENSO-DÍAZ, B., EGUÍA, I., y LARRAÑETA, J., 1997, Tabu Search approach for part-machine grouping in cellular manufacturing. *International Journal of Production Research*, (en revisión).
- MALEKI, R.A., 1991, *Flexible Manufacturing Systems: The Technology and Management* (Englewood Cliffs: Prentice Hall).
- MOHAMED, Z.M., 1996, A flexible approach to (re)configure Flexible Manufacturing Cells. *European Journal of Operational Research*, **95**, 566-576.
- MORENO, A.A., y DING, F., 1993, Heuristics for the FMS-Loading and Part-Type-Selection Problems. *The International Journal of Flexible Manufacturing Systems*, **5**, 287-300.
- MORRIS, J.S., y TERSINE, R.J., 1990, A simulation analysis of factors influencing the attractiveness of group technology layouts. *Management Science*, **36**, 1567-1578.
- NAGI, R., HARHALAKIS, G., y PROTH, J.M., 1990. Multiple routings and capacity considerations in group technology applications. *International Journal of Production Research*, **28**, 2243-2257.
- NESS, A.J., y CUCUZZA, T.G., 1995, Tapping the full potential of ABC. *Harvard Business Review*, **July-August**, 130-138.
- RAJAMANI, D., SINGH, N., y ANEJA, P., 1990, Integrated design of cellular manufacturing systems in the presence of alternative process plans. *International Journal of Production Research*, **28**, 1541-1554.
- RAJAMANI, D., SINGH, N., y ANEJA, P., 1992, Selection of parts and machines: A mathematical programming approach. *European Journal of Operational Research*, **62**, 47-52.
- RAM, B., SARIN, S., y CHEN, C.S., 1990, A model and a solution approach for the machine loading and tool allocation problem in a flexible manufacturing system. *International Journal of Production Research*, **28**, 637-645.
- SANKARAN, S., y KASILINGAM, R.G., 1990, An integrated approach to cell formation and part routing in group technology manufacturing systems. *Engineering Optimization*, **16**, 235-245.
- SANKARAN, S., y KASILINGAM, R.G., 1993, On cell size and machine requirements planning in group technology systems. *European Journal of Operational Research*, **89**, 373-383.
- SARIN, S.C., y CHEN, C.S., 1987, The machine loading and tool allocation problem in a flexible manufacturing system. *International Journal of Production Research*, **25**, 1081-1094.
- SAWIK, T., 1990, Modelling and scheduling of a Flexible Manufacturing System. *European Journal of Operational Research*, **45**, 177-190.
- SEIFODDINI, H., y DJASSEMI, M., 1996, The threshold value of a quality index for formation of cellular manufacturing systems. *International Journal of Production Research*, **34**, 3401-3416.
- SHANKER, K., y AGRAWAL, A.K., 1997, Models and solution methodologies for the generalized grouping problem in cellular manufacturing. *International Journal of Production Research*, **35**, 513-538.

- SHANKER, K., y RAJAMARTHANDAN, S., 1989, Loading problem in FMS: Part movement minimization, *In Proceedings of the Third ORSA/TIMS Conference on FMS: Operations Research Models and Applications*, Stecke and Suri (eds.), Elsevier, pp. 99-104.
- SHANKER, K., y SRINIVASULU, A., 1989, Some solution methodologies for loading problems in a flexible manufacturing system. *International Journal of Production Research*, **27**, 1019-1034.
- SHANKER, K., y TZEN, Y.J., 1985, A loading and dispatching problem in a random flexible manufacturing system. *International Journal of Production Research*, **23**, 579-595.
- SHERALI, H.D., SARIN, S.C., y DESAI, R., 1990, Models and algorithms for job selection, routing, and scheduling in a flexible manufacturing system. *Annals of Operations Research*, **26**, 433-453.
- SHTUB, A., 1989, Modelling group technology cell formation as a generalized assignment problem. *International Journal of Production Research*, **27**, 775-782.
- SIMAN V.4, 1989, *Reference Manual*. Systems Modelling Corporation, USA.
- SINGH, N., 1993, Design of cellular manufacturing systems: An invited review. *European Journal of Operational Research*, **69**, 284-291.
- SMITH, K., PALANISWAMI, M., y KRISHNAMOORTHY, M., 1996, A hybrid neural approach to combinatorial optimization. *Computers and Operations Research*, **23**, 597-610.
- SMITH, K., y PALANISWAMI, M., 1997, Static and dynamic channel assignment using neural networks. *IEEE Journal on Selected areas in Communications*, **15**, 238-249.
- SODHI, M.S., AGNETIS, A., y ASKIN, R.G., 1994-a, Tool Addition Strategies for Flexible Manufacturing Systems. *The International Journal of Flexible Manufacturing Systems*, **6**, 287-310.
- SODHI, M.S., ASKIN, R.G., y SEN, S., 1994-b, A Hierarchical Model for Control of Flexible Manufacturing Systems. *Journal of the Operational Research Society*, **45**, 1185-1196.
- SOLOMON, M.M., MILLEN, R.A., y AFENTAKIS, P., 1995, The formation of subfamilies for machine loading of flow-line cells. *International Journal of Production Research*, **33**, 2357-2374.
- SRIVASTAVA, B., y CHEN, W., 1993, Part type selection problem in flexible manufacturing systems: tabu search algorithms. *Annals of Operations Research*, **41**, 279-297.
- STECKE, K.E., 1983, Formulation and solution of nonlinear integer production planning problems for flexible manufacturing systems. *Management Science*, **29**, 273-288.
- STECKE, K.E., 1986, A hierchical approach to solving machine grouping and loading problems of flexible manufacturing systems. *European Journal of Operational Research*, **24**, 369-378.
- STECKE, K.E., y KIM, I., 1991, A flexible approach to part type selection in flexible flow systems using part mix ratios. *International Journal of Production Research*, **29**, 53-75.
- STECKE, K.E., y RAMAN, N., 1994, Production planning decisions in Flexible Manufacturing Systems with random materials flow. *IIE Transactions*, **26**, 2-17.
- STECKE, K.E., y SOLBERG, J.J., 1985, The optimality of unbalancing both workloads and machine group sizes in closed queueing networks of multiserver queues. *Operations Research*, **33**, 882-910.
- SURESH, N.C., y MEREDITH, J.R., 1994, Coping with the loss of Pooling Synergy in Cellular

- Manufacturing Systems. *Management Science*, **40**, 466-483.
- THOMAS, M.F., y MACKEY, J.T., 1994, Activity-based cost variances for just-in-time. *Management Accounting*, **April**, 49-54.
- VAN DEN BOUT, D.E., y MILLER, T.K., 1990, Graph partitioning using annealed neural networks. *IEEE Transactions on Neural Networks*, **1**, 192-203.
- VAN LOOVEREN, A.J., GELDERS, L.F., y VAN WASSENHOVE, L.N., 1986, A review of FMS planning models. *Modelling and Design of Flexible Manufacturing Systems*, editado por A. Kusiak (Amsterdam: Elsevier Science Publishers B.V.), pp. 3-31.
- WEMMERLÖV, U., y HYER, N.L., 1989, Cellular manufacturing in the U.S. industry: a survey of users. *International Journal of Production Research*, **27**, 1511-1530.
- WEN, U.P., y HUANG, A.D., 1996, A simple Tabu Search method to solve the mixed-integer linear bilevel programming problem, *European Journal of Operational Research*, **88**, 563-571.
- WILSON, J.M., 1989, An alternative formulation of the operation-allocation problem in flexible manufacturing systems. *International Journal of Production Research* **27**, 1405-1412.
- WILSON, J.M., 1992, Approaches to machine load balancing in flexible manufacturing systems. *Journal of the Operational Research Society*, **43**, 415-423.
- XA CALLABLE LIBRARY, 1994, *Reference Manual*. Sunset Software Technology, CA, USA.
- XU, H., y WANG, H-P., 1989, Part family formation for GT applications based on fuzzy mathematics. *International Journal of Production Research* **27**, 1637-1651.
- ZHANG, C., y WANG, H-P., 1992, Concurrent formation of part families and machine cells based on the fuzzy set theory. *Journal of Manufacturing Systems* **11**, 61-67.