IX Congreso de Ingeniería de Organización
Gijón, 8 y 9 de septiembre de 2005

Lotes óptimos de producción mediante algoritmos genéticos

Manuel Rojas Guerrero[1], Ángel Antonio Sarabia Viejo[2]

[1] Escuela Técnica Superior de Ingeniería ICAI.Universidad Pontificia Comillas de Madrid. Alberto Aguilera, 23. 28015 Madrid. m.rojas@atasa.net.
[2] Dpto. de Organización Industrial. Escuela Técnica Superior de Ingeniería ICAI. Universidad Pontificia Comillas de Madrid. Alberto Aguilera, 23. 28015 Madrid. asarabia@doi.icai.upco.es.

Resumen

Los problemas resolubles mediante técnicas de optimización heurística como los algoritmos genéticos son muy abundantes en la ingeniería, y de especial importancia en la organización industrial. En este artículo, se propone una aplicación de los algoritmos genéticos a un proceso de fabricación industrial que se debe llevar a cabo en lotes, basado en el problema del taller en carga (job-shop problem). La utilización de esta técnica de optimización heurística persigue la mejora en cuanto a tiempo de cálculo respecto al resultado obtenido mediante un modelo de optimización lineal clásico. En el proceso de fabricación en lotes se busca obtener un tamaño de lote óptimo de producción que responda a las características, en cuanto a costes, tiempo y calidad, requeridas por los responsables de fabricación. El algoritmo genético desarrollado tiene la peculiaridad de que cada gen tiene un alelo doble que permite identificar si se incurre o no en costes de montaje al procesar una máquina dos lotes sucesivos. La potencia de desarrollo que permiten los algoritmos genéticos posibilita recoger la casuística del problema de manera sencilla, y evaluar las salidas del algoritmo genético en tiempos de cálculo muy aceptables. Como medida del resultado, se presentan los tiempos de cálculo de CPU obtenidos a través de un modelo de programación lineal en variables enteras frente a los conseguidos mediante el algoritmo genético. En este sentido ha sido necesario introducir algunas hipótesis simplificadoras en orden a hacer posible un modelo lineal, simplificaciones que, sin incremento de tiempo de computación, pueden ser eliminadas para el algoritmo genético propuesto. El modelo lineal puede encontrarse en la página Web http:// www.atasa.net/articulo/modelo.pdf.

Palabras clave: Optimización lineal, algoritmo genético, planificación, job-shop problem.

1. Descripción del problema

1.1. El problema del taller en carga

Es un problema clásico de secuenciación o programación de tareas, que juega un papel clave en la organización de la producción. Básicamente se presenta como una situación que afecta a un taller de producción (no es difícil imaginar entornos de actividad en los que se dan situaciones similares: hospitales, administración, etc) en el que tienen que fabricarse ***n*** tareas o trabajos, cada uno de los cuales tiene una secuencia, en general distinta, de procesos sobre un conjunto de ***m*** máquinas o parte de ellas. Si las secuencias son idénticas para todas las tareas nos encontramos ante una situación simplificada conocida como ***taller en flujo*** (“*flow-shop*”).

Circunstancias, entre otras posibles, a considerar en cada caso concreto son:

- los tiempos de ejecución de los distintos procesos de cada tarea en cada máquina,
- las posibles ventanas de tiempos para la ejecución de los procesos,
- la posibilidad de que una tarea sufra más de un proceso en una máquina dada,
- existencia de más de una máquina para realizar un mismo proceso.

El problema que se ha modelado es una variante del problema del taller en carga, cuya mayor singularidad es la fabricación de los diferentes artículos en lotes de tamaño dado.

1.2. Descripción y particularidades del problema

Se tiene un número **n** de artículos distintos a fabricar. De cada artículo $\boldsymbol{i}$, $\boldsymbol{i} \in \{\mathbf{1}, \mathbf{2}, \mathbf{....}, \boldsymbol{n}\}$, existe una demanda determinista de unidades $\boldsymbol{d}(\boldsymbol{i})$ que deben ser entregadas en un plazo fijado $\boldsymbol{T_e}(\boldsymbol{i})$.

El artículo $\boldsymbol{i}$ debe ser procesado en un subconjunto $\boldsymbol{S}(\boldsymbol{i})$ de un conjunto de **m** máquinas $\boldsymbol{S}(\boldsymbol{i}) \subset \{1, 2, ..., \boldsymbol{m}\}$ en un orden dado, que puede ser distinto para cada artículo, invirtiendo en la operación realizada sobre la máquina $\boldsymbol{j}, \boldsymbol{j} \in \boldsymbol{S}(\boldsymbol{i})$, un tiempo $\boldsymbol{t}_{i,j}$.

Producir unidades del artículo $\boldsymbol{i}$ en la máquina $\boldsymbol{j}$ supone un coste, constituido por dos componentes:

- un coste de montaje $\boldsymbol{p}_{i,j}$
- un coste unitario de fabricación $\boldsymbol{u}_{i,j}$

Las unidades del artículo **i** que serán procesadas en la máquina **j** son colocadas junto a dicha máquina, constituyendo un inventario de artículos en fase de procesamiento, que incurren en un coste de almacenamiento por unidad y tiempo $\boldsymbol{a}_{i,j}$.

Una vez que unidades de un artículo comienzan a ser procesadas sobre una máquina la operación no puede interrumpirse.

El coste de reproceso de una unidad del artículo **i** tras su paso por la máquina **j** es $\boldsymbol{r}_{i,j}$.

Los artículos a fabricar no se procesan en unidades de un solo artículo por lo general, sino que se fabricarán en lotes. Fabricar lotes de gran tamaño tiene la ventaja de ahorrar costes de operación, por el ahorro de costes de montaje, pero los inconvenientes de incrementar los costes de almacenamiento y el número de piezas defectuosas en caso de desajustes de la máquina.

La solución planteada al problema consiste en obtener para cada máquina una lista de trabajo que muestre la secuencia en que debe ser procesado cada uno de los lotes de los diferentes artículos sobre cada máquina, respetando la secuencia propia de proceso y evitando posibles solapes, ya que una máquina no puede procesar simultáneamente unidades de dos artículos distintos.

El objetivo último es satisfacer la demanda en el plazo previsto, con la mínima demora total posible, para cada tipo de artículo, a coste mínimo. Por tanto, se fija una optimización por metas, donde el primer objetivo es minimizar la demora total de la fabricación del total de la demanda para cada artículo y, en segundo lugar, se minimizan todos los costes en que se incurre en el proceso de producción.

1.3. Hipótesis simplificadoras para elaborar el modelo lineal

Para cada artículo **i** cuya demanda es $\boldsymbol{d_i}$, se va a suponer que los lotes de fabricación son aproximadamente iguales. Para modelar este supuesto se introduce el concepto de tamaño máximo de lote de fabricación $\boldsymbol{L_i}$ para cada artículo **i**, que es conocido de antemano para resolver el problema. La cantidad demandada de cada uno de los artículos será procesada en lotes cuyos tamaños serán establecidos según dos situaciones distintas:

i) Si $d_i = \dot{\overline{L_i}}$, los lotes de fabricación se toman según múltiplos de L_i . Por ejemplo, con d_i=120 y L_i=30, se consideraría establecer 4 lotes de fabricación de 30 artículos cada uno.

ii) Si $d_i = \dot{\overline{L_i}} + k$, el número de lotes a fabricar de un determinado artículo se establece de una forma más compleja que la anterior. La mayoría de los lotes tendrán un tamaño calculado según la expresión (1) y, el resto de artículos, es decir, $L_i + k$ serán fabricados en lotes más pequeños que L_i, que podrán ser iguales o no.

$$N*_i = \frac{d_i - k}{L_i} - 1 \qquad (1)$$

Un ejemplo de este segundo caso sería considerar d_i =130, L_i =30, k =10. En este caso se fabricarán:

$$N*_i = \frac{d_i - k}{L_i} - 1 = \frac{120}{30} - 1 = 3 \text{ lotes de 30 artículos}$$

y luego se fabricarán 30+10=40 artículos en lotes más pequeños que L_i.

2. El algoritmo genético

Las componentes que conforman el algoritmo genético están definidas con precisión con el objetivo de conocer su impacto en el proceso de resolución del problema. Todas las componentes que se describen a continuación tienen especial importancia en cada una de las tareas que tiene que llevar a cabo el algoritmo genético en su proceso iterativo de búsqueda de las soluciones más factibles.

2.1. Descripción de las funciones objetivo y de ajuste

La función objetivo que es evaluada en cada iteración del algoritmo genético se desglosa en dos metas. La primera meta es obtener la demora total mínima posible. Es decir, se pretende conseguir un “individuo”(solución) que codifique una secuencia de lotes a través de las diferentes máquinas por las que éstos deban pasar de manera que sea lo más próxima posible a cero, esto es, que se aproxime al máximo al tiempo de entrega solicitado por el cliente para cada uno de los artículos.

En segundo lugar, y debido a que se pueden obtener diferentes individuos con la misma demora total, se define como segunda meta la minimización de los costes totales. Estos costes son la suma de los diferentes costes en los que se incurre durante el proceso de fabricación, es decir, fabricación, almacenamiento, reproceso y montaje.

La función de ajuste determina la optimalidad de un individuo de una determinada generación para entrar a formar parte de la siguiente. El cálculo de la función de ajuste consiste en una comparativa de cada individuo con respecto al resto de individuos, basándose en los resultados de las evaluaciones de los individuos que forman parte de la misma generación.

2.2. Elección de la codificación

En el algoritmo descrito en este trabajo se utiliza una codificación directa, mediante alfabeto numérico entero. Un cromosoma está compuesto por un conjunto de genes dobles, es decir, cada gen tiene dos alelos, siendo esta una importante variante respecto a la codificación utilizada por Tsujimura, Gen y Kubota (1995).

Cada gen de un cromosoma está asociado a un lote de un cierto artículo. Las posiciones o alelos impares de un gen cualquiera de un cromosoma hacen referencia a un determinado artículo. Las posiciones o alelos pares de un gen concreto determinan un lote.

La información proveniente de la cadena genética del individuo, junto con la procedente de la matriz de orden, similar a la del ejemplo (Tabla1), permite asignar en cada momento los lotes que deben ser procesados en cada una de las máquinas del taller.

Tabla 1. Orden de los artículos por las máquinas del taller.

		Número de Máquina	
		1	2
Número de Artículo	1	1	2
	2	0	1
	3	2	1

La interpretación del cromosoma resulta bastante intuitiva mediante la codificación directa que se ha utilizado. El ejemplo que se muestra (Tabla 2) se ha generado considerando una demanda de una unidad para cada uno de los artículos a fabricar.

Tabla 2. Individuo genético de una población.

Individuo	Generación	Demora Total	Coste Total
(1-1)(3-1)(1-1)(3-1)(2-1)	3	171	40.6

La interpretación del cromosoma sería:

En primer lugar se encuentra el gen doble (1-1) que implica que el lote 1 del artículo 1 pasa en primer lugar por la máquina que tenga asignada en la matriz de orden (Tabla 1) como número uno, es decir, por la máquina número uno.

A continuación se ha generado el gen doble (3-1) que significa que el lote 1 del artículo 3 pasa por la máquina que deba en siguiente lugar, que consultando la matriz es la máquina dos.

En tercer lugar está el gen doble (1-1), lo que significa que el lote 1 del artículo 1 pasa por la máquina que le corresponda. Como este lote ya ha pasado por su primera máquina, ahora deberá hacerlo por la siguiente que indique la matriz de orden, que en este caso es por la máquina dos.

En penúltimo lugar se encuentra el gen doble (3-1) que indica de forma análoga a los anteriores que el lote 1 del artículo 3 debe pasar por al máquina que le corresponda, que en este caso es la máquina uno.

El último gen doble del individuo es el (2-1) que significa que el lote 1 del artículo 2 debe pasar por la máquina que le corresponda, que en este caso es la 2, puesto que sólo debe pasar por ésta y tiene un 1 en la matriz.

Asociado a cada "individuo" de la población también existe otro tipo de información que complementa a la lista de trabajo que representa el cromosoma. Los datos asociados son el número de individuo dentro del total de la población, el cromosoma o lista de trabajo, la generación a la que pertenece el individuo y las evaluaciones en cuanto a demora total y coste total.

2.3. Generación de la población inicial

La existencia de una población inicial de individuos que sea bastante variada favorecerá el buen comportamiento del algoritmo genético, a través de un desarrollo equilibrado de la especie. Para conseguir obtener una población inicial buena, es necesario determinar de manera óptima dos parámetros: el mecanismo de generación de individuos y el tamaño de la población inicial.

El algoritmo genético desarrollado utiliza un sistema de generación de la población inicial de tipo aleatorio, aunque posee una parte determinista que implica la reordenación de material genético para evitar la aparición de monstruos. Para cada uno de los individuos de la población inicial se realiza la generación aleatoria mediante pares de valores de la siguiente forma:

- Los genes impares, que se corresponden con los artículos a producir, toman valores entre uno y el máximo número de artículos distintos a producir.
- Cada uno de los genes pares, que se corresponden con los lotes del artículo al que se refiere el gen impar anterior, toman un valor acotado entre 1 y el máximo número de lotes a fabricar para ese determinado artículo.

Además de este control que se lleva durante el proceso de generación de números aleatorios, existe un control sobre la generación del cromosoma completo. A medida que se van generando los genes dobles, si no es válida la aparición en el cromosoma de un gen doble que ya haya sido generado, se añada a una matriz para que en caso de aparición se elimine de forma automática y se vuelva a generar un nuevo gen doble, hasta que aparezca uno que sea válido en la secuencia generada.

Con respecto al segundo parámetro importante para la población inicial, el tamaño de ésta, no tiene un valor inicial obligatorio, pero se ha incluido en el programa una utilidad que aconseja mediante una heurística muy sencilla cuál debiera ser un tamaño aproximado al óptimo.

La heurística seguida para establecer la recomendación acerca del tamaño de la población inicial se calcula en función de un porcentaje respecto del tamaño total del cromosoma genético. El tamaño del cromosoma depende de tres parámetros independientes y, demás, hay otros dos que se deducen de los primeros.

- Número de artículos distintos que hay que fabricar ***i*.**
- Demanda solicitada de cada uno de los artículos ***d*(*i*).**
- Número de máquinas por las que pasa cada artículo m_i.
- Número de lotes de cada artículo que hay que fabricar l_i.
- Considerar que los genes del cromosoma son dobles.

El tamaño del cromosoma (T_c) se refleja por la ecuación

$$T_c = 2 \times \sum_{i=1}^{n} l_i \times m_i \tag{2}$$

Una vez obtenido el tamaño del cromosoma, se estima el tamaño de la población inicial (T_o), recomendándose al usuario que utilice:

$$T_o = \left[0.6 \times T_c \right] \tag{3}$$

2.4. Evaluación de la población

A través de la evaluación de los distintos individuos de cada una de las generaciones se establece la demora obtenida y los costes totales de cada uno de estos individuos.

Previo a la evaluación de la demora y los costes, hay que calcular que artículos van a ser reprocesados y que lotes incurren en un proceso de montaje. Esto se realiza mediante matrices auxiliares, a la vez que se recorre el cromosoma a evaluar.

En la evaluación de la demora total de cada uno de los individuos, se utilizan básicamente dos matrices auxiliares. Considerando los siguientes subíndices auxiliares:

i : Número de artículo

k: Número de lote

j: Número de máquina

$Tlote_{i,k,j}$: Indica el tiempo en el que acaba de fabricarse un determinado lote de un artículo a su paso por una determinada máquina.

$Tmaq_{j}$: Indica el tiempo en que terminó el último lote de un artículo que tuvo que fabricarse sobre una determinada máquina.

La evaluación de los costes utiliza los datos obtenidos por el algoritmo de evaluación de la demora, además de una matriz auxiliar que representa los tiempos de espera.

2.5. Obtención de una nueva generación

En esta etapa se ha implantado el proceso de reproducción, pero no se han introducido efectos como la mutación, la inmigración o la inversión. La razón de no introducir tales fenómenos se debe a la complejidad que pudieran generar en el algoritmo de evaluación y en el ajuste de los individuos de la población, ya que la aparición de un cambio en un gen doble implica validar la corrección del cromosoma, para que este no resulte ser un monstruo.

El proceso de reproducción está definido en torno a dos decisiones que se detallan en los siguientes apartados.

2.5.1. Individuos que tendrán acceso a la reproducción

En el algoritmo desarrollado, se considera en cada generación un individuo como genio. Este individuo es el que posee en su evaluación el menor valor de la demora total. Este individuo pasa de forma automática como el primero de la siguiente generación, a la vez que es seleccionado como “semental”.
El individuo genio, a su vez “semental”, es seleccionado para el cruce con el resto de individuos de su generación.

2.5.2. Posición del cromosoma en que se producirá el intercambio de material genético

El cruce en cada generación se realiza entre el individuo semental y cada uno del resto de individuos que conforman la generación, sin existir otro tipo de emparejamiento. Se generan aleatoriamente dos posiciones de cruce tanto en el individuo semental como en el que corresponda de su generación que hay que cruzar. El conjunto de genes que corresponde al intervalo entre las dos posiciones de cruce generadas en uno y otro individuo, son los que intervienen en el proceso de intercambio de material genético entre individuos. De esta forma, cada intercambio entre el semental y otro de los individuos, produce dos nuevos individuos, los cuales son reordenados para que no resulten monstruos y son evaluados, pasando a la siguiente generación sólo el mejor.

2.6. Criterios de parada

El algoritmo genético desarrollado contempla dos criterios de parada alternativos, es decir, en cuanto se satisface alguno de los criterios de parada, el algoritmo genético concluye y aporta una solución al usuario.

i) El criterio de parada más sencillo que se ha utilizado es el número de generaciones a obtener. El valor a utilizar es flexible, aunque se ha incorporado un valor recomendado que ha sido obtenido en base a la experimentación con el algoritmo

$$\boldsymbol{Número\_de\_generaciones} = \left[\mathbf{0.3} \times \boldsymbol{T_c}\right] \tag{4}$$

donde T_c el tamaño del cromosoma.

ii) El criterio de parada alternativo se ha denominado factor de ajuste. El factor de ajuste mide la convergencia de los valores de la función objetivo, siendo un parámetro controlador de esta. Su valor representa el porcentaje en el cual pueden diferenciarse los valores de la función objetivo demora total entre cada uno de los individuos de una generación.

Como se ha mencionado anteriormente, si se alcanza el número de generaciones expresadas en el parámetro número de generaciones o si la diferencia en la función objetivo no supera el valor porcentual expresado en el parámetro factor de ajuste, el algoritmo genético finalizará, y ya habrá obtenido una solución.

3. Análisis de resultados y conclusiones

El algoritmo genético proporciona resultados fiables y rápidos gracias al desarrollo en paralelo del modelo de optimización lineal, que ha permitido modificar los parámetros del A.G. que introducen problemas indeseados como tamaños de población inicial excesivos u obtención de una solución por convergencia lenta. Además, el modelo ha permitido contrastar la fiabilidad del algoritmo.

En el proceso de desarrollo del algoritmo genético no se han aprovechado algunas características que permiten utilizar este tipo de técnicas, como es el caso de la mutación. La negativa a no utilizar mutación se debe al coste computacional que supone reordenar todo los genes de un individuo como consecuencia de la alteración de cualquier alelo de un gen doble, por las características inherentes del problema.

Las listas de trabajo generadas con el algoritmo genético han demostrado ser tan fiables y óptimas como las obtenidas por métodos clásicos y, han sido obtenidas en cantidades de tiempo inferiores a las necesitadas mediante optimización lineal.

Los tiempos de proceso se han obtenido de la realización de diferentes pruebas (Tabla 3) en las que se han variado los parámetros que más influyen en el tiempo de cálculo, como son el número de lotes que se deben fabricar en el taller y las demandas de cada uno de los artículos.

Tabla 3. Resultados experimentales obtenidos.

Prueba	Número de Lotes	Demandas de Artículo			Resultados	
		1	2	3	Modelo Opt.	A.G.
1	4	5	8	4	29ms	63ms
2	6	10	8	8	40ms	78ms
3	8	10	16	8	84ms	141ms
4	10	15	24	12	865.55s	204ms
5	12	20	16	16	7021.39s	271ms

Referencias

Esquivel S.; Ferrero S.; Gallard R.; Salto C.; Alfonso H.; Schütz M. (2002). Enhanced evolutionary algorithms for single and multiobjective optimization in the job shop scheduling problem. *Journal of Knowledge Based Systems, Vol 15/1-2, pp 13-25, Elsevier*.

Goldberg, D.E. (1989). Genetic Algorithms in Search, Optimization and Machine Learning, *Addison-Wesley, Reading, MA.*

Pham, D.T. ; Karaboga, D. (2000). Intelligent Optimization Techniques. *Springer-Verlag, Londres.*

Raz, T; Kaspi M. (1991). Location and sequencing of imperfect inspections in serial multistage production systems, *Int. J. of Production Research, Vol.29, No.8, pp.1645-1659.*
Reeves, C.R. (1995). Modern Heuristic Techniques for Combinatorial Problems, *McGraw-Hill, Maidenhead, UK.*
Tsujimura, Y.; Gen, M.; Kubota, E. (1995). Solving Job-shop Scheduling Problem with Fuzzy Processing Time Using Genetic Algorithms. *Japanese Journal of Fuzzy Theory and Systems, Vol 7, Number 5*.
Wang, L.; Zhang, L.; Zheng, DZ. (2003). A class of order-based genetic algorithm for flow shop scheduling. *The International Journal of Advanced Manufacturing Technology, Springer-Verlag, London.*