Ingeniería Mecánica. Vol. 12. No.2, mayo-agosto de 2009, pag. 37-48 ISSN 1815-5944

Modelo matemático y algoritmo de solución de un flujo con tiempos dependientes de la secuencia

A. Fiol-Zulueta, J. Arzola-Ruíz, A. da Silva-Carvalho

Recibido el 5 de octubre de 2008; aceptado el 2 de febrero de 2009

Resumen

El problema de la obtención de secuencias de trabajo en las actividades productivas consiste en la determinación de una posible combinación, deseablemente la mejor, de opciones de secuencias de trabajo para cada uno de los elementos que conforman la entidad productiva y por todas ellas, a partir de la elaboración previa de secuencias evaluadas considerando un determinado indicador de eficiencia. En el trabajo se propone un modelo matemático para el caso de un taller de flujo híbrido con tiempos de procesamiento dependientes de la secuencia y de las máquinas, y su implementación utilizando una meta heurística, con el propósito de ayudar a la toma de decisiones con vistas a mejorar los instrumentos utilizados para lograr la conciliación de las secuencias de trabajo en los talleres de producción, debido a la necesidad de elevar los indicadores de eficiencia de dichos talleres.

Palabras claves: secuencias de producción en talleres de maquinado, taller de flujo, meta heurística, optimización bajo criterios múltiples.

A mathematical model and solution algorithm to a flow shop task with times depending of the sequence.

Abstract

The problem of obtaining work sequences in the productive activities consists on the determination of a possible combination, desirably the best, of work sequences for each one of the elements of the productive entity and for all them starting from the previous elaboration of evaluated sequences considering a certain efficiency indicator. In the work a mathematical model, for the case of a hybrid flow shop with times depending of the sequence prosecution and of the machines characteristics, and it implementation using a metaheurístic procedure are proposed, with the purpose of aiding the decisions making process with a view of improving the tools used to achieve the work sequences conciliation of the production shops, due to the necessity of raising the efficiency indicators of these workshops.

Key words: work sequences in mechanical industry workshops, flow shop task, metahuristics, multiple criteria optimization.

1. Introducción.

El problema de la obtención de secuencias de trabajo en las actividades productivas consiste en la determinación de una posible combinación, deseablemente la mejor, de opciones de secuencias de trabajo para cada uno de las partes que conforman la entidad productiva y por todas ellas, a partir de la elaboración previa de secuencias evaluadas considerando un determinado indicador de eficiencia.

Los sistemas conocidos por el nombre de taller de flujo híbrido o sistemas de flujo híbrido son sistemas productivos compuestos por varias etapas en serie, cada una de las cuales puede contar con más de una estación o máquina en paralelo. En cada etapa, las máquinas disponibles para realizar una operación pueden ser idénticas, proporcionales o diferentes. En un programa para un taller de flujo híbrido, es tan importante la secuenciación de piezas en cada máquina como la asignación de éstas a una de las diferentes máquinas de la etapa.

En condiciones de producción de diferentes piezas del mismo tipo, la tecnología de fabricación de cada pieza tiene asociada una secuencia estricta de etapas y de operaciones en cada etapa. Estas secuencias son elaboradas previamente por el Sistema CAPP de planeación del proceso de fabricación.

2. Integración del diseño, la generación de tecnologías, la fabricación y la planeación de la producción de piezas maquinadas.

El problema estudiado en este trabajo ha sido objeto de investigación, no siempre con éxito, en numerosos trabajos. Entre ellos se pueden mencionar los trabajos presentados en [1, 2, 3, 4, 5, 6] y otros muchos.

Las tareas de integración de los sistemas CAD/CAPP/CAM (para el diseño. la generación de tecnologías y la fabricación) obedecen a los mismos principios que los restantes sistemas. Así, se estudia en una primera fase, la tarea de dirección de mayor envergadura a la que se subordina, en calidad de subtarea, el diseño de la pieza, herramienta, instalación, etc., que se trate.

En una segunda etapa se realiza el análisis de la tarea de diseño. La tarea de planeación del proceso de fabricación se subordina, necesariamente, al diseño del producto a fabricar. Esta subordinación obedece a los principios de organización de la estructura centralizada, ya que se requiere fabricar exactamente lo que se diseñó. El sistema CAPP es el encargado de generar las diferentes opciones de fabricación del producto para los agregados productivos del taller en cuestión, por lo que suministra propuestas al sistema de dirección de la producción del taller. La propuesta seleccionada para cada fase dada de fabricación de cada producto en cada agregado tecnológico debe ser implementada en el momento establecido por el sistema de dirección de la producción del taller. El cumplimiento de las tareas productivas puede y debe ser realizado mediante la transferencia de la información directiva desde el CAPP al agregado tecnológico que se trate.

Como resultado de la planeación del proceso de fabricación (sistema CAPP) se determina un programa que comprende los siguientes elementos.

- Una asignación de cada trabajo (pieza a elaborar) y las operaciones a efectuar en la etapa sobre cada pieza, a un conjunto de máquinas, comprendidas en cada etapa.
- Una secuencia de trabajos a realizar en cada máquina de cada etapa.

El sistema CAPP se ocupa de definir el orden de realización de los trabajos en cada etapa, el que puede diferir en varias etapas.

3. Análisis de la tarea de secuenciación de fabricación.

El análisis se realiza por la metodología de Análisis de Sistemas de Ingeniería dada en [7], la que incluye las etapas de análisis externo e interno.

Análisis externo de la tarea.

Este análisis consiste en la clasificación de la información involucrada en variables de entrada:

- Variables de decisión
- Variables de coordinación
- Datos de entrada

- Indicadores de eficiencia

A partir del estudio bibliográfico y de encuestas a especialistas y empresarios realizadas, en este trabajo se definen los siguientes indicadores de eficiencia:

- C_{max}: tiempo total de proceso.
- A_T : número total de trabajos atrasados.

Las variables de decisión son aquellas que pueden ser variadas a voluntad por el decisor de la tarea, en este caso, X_{ijl}^{t} toma el valor 1, si el trabajo *j* es programado inmediatamente antes del trabajo *l* en la máquina *i* en la etapa *t*, y el valor 0, en otros casos.

Las variables de coordinación vienen dadas por el material, las dimensiones, ajustes y otras especificaciones técnicas que sean necesarias para cada ítem a ser fabricado.

Los datos de entrada vienen dados por:

- r_j : fecha de realización del trabajo *j*
- m^t**:** número de máquinas paralelas en la etapa *t*
- d_j : fecha de entrega del trabajo *j*
- s_{jl}^{t} : tiempo de preparación entre el trabajo *i* y el trabajo *j* en la etapa *t*
- ch_{ij}^{t} : tiempo de preparación del trabajo *j* si el trabajo *j* es asignado a la máquina *i* en la primera posición de determinada etapa *t*
- Op_j^t : tiempo de procesamiento estándar del trabajo *j* en la etapa *t*
- v_{ij}^{t} : velocidad relativa de la máquina *i* en la etapa *t* para el trabajo *j*
- a_i^t : disponibilidad de tiempo en la máquina *i* en la etapa *t*
- Or_j^t : tiempo de operación del trabajo *j* en la etapa *t*

Como variables intermedias de interés se tienen:

- U_i : igual a 1 si el trabajo *j* se atrasa como resultado de las decisiones de secuenciación adoptadas y 0, en otros casos.
- C_j^t : tiempo de procesamiento del trabajo *j* en la etapa *t*
- T_j**:** tardanza del trabajo *j* como resultado de las decisiones adoptadas

Análisis interno de la tarea.

Consiste en la determinación del modelo matemático y del procesamiento de cálculo requerido para obtener las salidas como función de las entradas.

En un problema de taller de flujo flexible, son procesadas un conjunto de *n* órdenes de trabajo independientes de *j* clientes, $j \in \{1, 2,\ldots, n\}$ con fechas de entrega, $d_1,\ldots, d_n$. Normalmente las órdenes de los clientes no llegan simultáneamente en los problemas reales, por lo que se asume que tienen tiempos de realización no negativos, $r_{1,\ldots}, r_n$. Cada trabajo *j* consiste en una cadena de *k* ($k \geq 2$) operaciones diferentes, $O_j^1,\cdots,O_j^k$ las cuales tienen que ser procesadas en ese orden. Cada trabajo *j* tiene un tiempo unitario fijo Op_j^t para cada etapa *t*, $t \in \{1, 2,\ldots, k\}$. Además, para cada etapa *t*, se tiene un conjunto de m_i^t máquinas paralelas no relacionadas, $i \in \{1,2,\ldots,m^t\}$ donde alguna de las etapas de la producción tiene exactamente una sola máquina, pero al menos una de las etapas de la producción puede tener múltiples máquinas. La máquina *i* en la etapa *t* puede procesar el trabajo *j* a la velocidad relativa v_{ij}^t.

En este problema se asumen herramientas para la programación estática y determinística, las que presuponen restricciones que aseguren los siguientes requisitos:

No se permite la interrupción de la operación una vez comenzada:

- Las operaciones de un trabajo tienen que ser realizadas secuencialmente, sin solapamiento entre las etapas y no se permiten trabajos partidos.
- Los tiempos de preparación que se utilizan se clasifican en dos tipos: tiempo de preparación dependiente de la máquina y tiempo de

preparación dependiente de la secuencia. El tiempo de preparación dependiente de la máquina es de acuerdo con el tiempo de preparación para la máquina a la cual le ha sido asignado el trabajo, esto se asume que ocurra solamente cuando el trabajo es asignado a la máquina en la primera posición de una etapa determinada.

- El tiempo de preparación dependiente de la máquina le concierne al trabajo en la primera secuencia de cualquier etapa, mientras que el tiempo de preparación dependiente de la secuencia es el considerado entre los restantes trabajos.
- El tiempo de preparación S_{lj}^{t} entre el trabajo *l* y el trabajo *j* en la etapa *t*, donde el trabajo *l* es procesado directamente antes del trabajo *j* en la misma máquina, puede tener diferentes valores dependientes del trabajo que haya sido completado y del trabajo que será procesado (dependencia de la secuencia). Todos los tiempos de preparación son conocidos y constantes.

Se imponen adicionalmente las siguientes restricciones:

1. Ninguna máquina puede procesar más de un trabajo al mismo tiempo.
2. Ningún trabajo puede ser procesado por más de una máquina al mismo tiempo.
3. La fecha de trabajo a_i^t (o disponibilidad) de la máquina para cada etapa no es necesariamente igual a cero; sin embargo, las máquinas están siempre disponibles para ese tiempo de disponibilidad.

Se definen:

- El tiempo total de proceso del trabajo *j,* se denota por C_j, entonces el tiempo de proceso total es $C_{\max} = \max_{j\in\{1,\cdots,n\}}\{C_j\}$
- U_i: toma el valor 1 si el tiempo de realización del trabajo *j* es menor que el tiempo de procesamiento C_i del trabajo *j*, y 0 en otros casos.
- El número total de trabajos atrasados se determina por $A_T = \sum_{j=1}^{n} U_j$

El procedimiento de programación a utilizar incluye la respuesta a una de las demandas de los clientes, la que refleja el nivel de prestación de servicio, y a uno de los objetivos del programador, constituido por la minimización del inventario en curso. El primer objetivo se modela, en este caso, mediante la minimización de la cantidad de trabajos atrasados y, en el segundo, a través de la minimización de la suma de los tiempos de finalización de la fabricación de las piezas.

El objetivo es encontrar la secuenciación de horarios y la ocupación de cada máquina con cada trabajo en cada etapa que minimice una aproximación a una función de utilidad que incluya C_{max} y A_T. Por lo tanto, en calidad de aproximación a la función de utilidad se utiliza, en este trabajo, la distancia ponderada de Tchebycheff entre los valores deseados de los indicadores de eficiencia señalados y el espacio de existencia de los criterios normalizados C_{max} y A_T.

El cálculo de algunas variables de interés se aclara directamente en el modelo matemático siguiente, el que incluye los resultados del Análisis Externo e Interno expuestos.

4. Modelo matemático de la tarea.

La preparación de decisiones de la tarea planteada consiste en la generación de opciones de solución al modelo matemático siguiente.

Notaciones:

t índice de las etapas, $t \in \{1,2,\ldots,k\}$

i índice de las máquinas, $i \in \{1,2,\ldots,m^t\}$

j , *l* índices de los trabajos, $j, l \in \{1,2,\ldots,n\}$

r_j, m^t, d_j, s_{ij}^t, ch_{ij}^t, Op_j^t, v_{ij}^t, a_i^t, X_{ijl}^t, Or_j^t, C_j^t, C_{max}, U_j, T_j, A_T se definen en correspondencia con el Análisis Externo.

En correspondencia con el Análisis Externo e Interno se presenta la siguiente estructura para el modelo matemático general:

$$\textbf{\textit{minimizar}}\ \max\left\{\lambda\frac{\left|\mathrm{C}_{\max}-C_{\max}^{deseado}\right|}{C_{\max}^{deseado}},(1-\lambda)\frac{\left|A_T-A_T^{deseado}\right|}{A_T^{deseado}}\right\} \tag{1}$$

sujeto a:

$$\sum_{i=1}^{m^t}\sum_{j=0}^{n}X_{ijl}^{t}=1 \qquad \forall t,l \tag{2}$$

$$\sum_{i=1}^{m^t}\sum_{l=1}^{n+1}X_{ijl}^{t}=1 \qquad \forall t,j \tag{3}$$

$$\sum_{l=1}^{n+1}X_{i0l}^{t}=1 \qquad \forall t,i \tag{4}$$

$$\sum_{j=0}^{n}X_{ij(n+1)}^{t}=1 \qquad \forall t,i \tag{5}$$

$$X_{ijj}^{t}=0 \qquad \forall t,i,j \tag{6}$$

$$\sum_{j=0}^{n}X_{ijl}^{t}=\sum_{j=1}^{n+1}X_{ilj}^{t} \qquad \forall t,i,j \tag{7}$$

$$X_{ijl}^{t}\in\{0,1\} \qquad \forall t,i,j,l,j=0,l=n+1 \tag{8}$$

$$Or_j^t=\sum_{i=1}^{m^t}\sum_{l=1}^{n+1}\frac{Op_j^t}{v_{ij}^t}X_{ijl}^{t} \qquad \forall t,j \tag{9}$$

$$C_l^t-C_j^t\geq s_{jl}^t+Or_l^t+((\sum_{i=1}^{m^t}X_{ijl}^{t})-1)B \qquad \forall t,j,l,j\neq 1 \tag{10}$$

$$C_j^t\geq 0 \qquad \forall t,j \tag{11}$$

$$C_l^t-C_l^{t-1}\geq\sum_{i=1}^{m^t}\sum_{j=1}^{n}X_{ijl}^{t}\,s_{jl}^{t}+\sum_{i=1}^{m^t}ch_{il}^{t}\,X_{i0l}^{t}+Or_l^t \qquad \forall t,l \tag{12}$$

$$C_j^0=r_j \qquad \forall j \tag{13}$$

$$C_j^t\geq\sum_{i=1}^{m^t}a_i^t\,X_{i0j}^{t}+\sum_{i=1}^{m^t}ch_{il}^{t}\,X_{i0l}^{t}+Or_j^t \qquad \forall t,j \tag{14}$$

$$C_{\max}\geq C_j^k \qquad \forall j \tag{15}$$

$$T_j\geq C_j^k-d_j \qquad \forall j \tag{16}$$

$$T_j\geq 0 \qquad \forall j \tag{17}$$

$$U_j\leq BT_j \qquad \forall j \tag{18}$$

$$BU_j\geq T_j \qquad \forall j \tag{19}$$

$$A_T=\sum_{j=1}^{n}U_j \qquad \forall j \tag{20}$$

$$U_j\in\{0,1\} \qquad \forall j \tag{21}$$

La expresión (1) describe la función objetivo, consistente en la minimización de la distancia ponderada de Tchebycheff entre un nivel de aspiración del decisor, expresado como valores deseados de los dos objetivos normalizados indicados anteriormente y el espacio de existencia de los criterios. La normalización de los criterios se realiza mediante la división de los valores absolutos de las diferencias de cada criterio con respecto a su valor deseado entre el valor deseado y tiene como propósito que ambos criterios tengan el mismo orden de magnitud durante la búsqueda del mejor compromiso entre ellos. Se introduce que $X_{i0l}^{t} = 1$ si el trabajo *j* es ubicado como primer trabajo en la máquina *i* en la etapa *t*, y $X_{ij(n+1)}^{t} = 1$ si el trabajo *j* es ubicado como último trabajo en la máquina *i* en la etapa *t*. Las restricciones (2) – (8) permiten asegurar que la programación parcial en cada máquina en cada etapa sea factible, así mismo las restricciones (2) y (3) aseguran que solamente un trabajo sea asignado a cada posición en la secuencia de cada etapa. Las restricciones (4) y (5) permiten asegurar que solamente un trabajo puede ser asignado en la primera y la última posición en cada máquina y en cada etapa. La restricción (6) asegura que después de que el trabajo ha finalizado en alguna etapa, no puede ser reprocesado en la misma etapa. La restricción (7) obliga a la construcción de secuencias consistentes en cada etapa. La restricción (8) especifica las variables de decisión como variables binarias.

Las restricciones (10) – (14) permiten hallar el tiempo de terminación de cada trabajo. La restricción (10) es el conjunto de las restricciones disyuntivas, lo que indica que, si los trabajos *j* y *l* son programados en la misma máquina en una etapa particular con el trabajo *j* programado antes del trabajo *l*, entonces el trabajo *j* debe terminar el proceso antes de que el trabajo *l* pueda comenzar. Esta restricción obliga al trabajo *l* a seguir al trabajo *j* con por lo menos el tiempo de procesamiento del trabajo *l* más el tiempo de preparación de *j* para *l* si el trabajo *l* es programado inmediatamente después del trabajo *j*. El valor de *B* se fija a una constante muy grande, por ejemplo, mayor que la suma de todos los tiempos de procesamiento de los trabajos y de los tiempos de preparación. La restricción (11) asegura que el tiempo de completamiento de cada trabajo en cada etapa sea un valor no negativo.

La restricción (12) especifica las restricciones de precedencia conjuntiva para los trabajos, en la que se plantea que un trabajo no puede comenzar su procesamiento en la etapa *t* + 1 antes de que acabe en la etapa *t*. La restricción (13) se aplica solamente a la primera etapa, lo que quiere decir que un trabajo no puede comenzar su procesamiento en la primera etapa antes de su fecha de realización. La restricción (14) se aplica solamente a los trabajos que son asignados a la primera secuencia en cada máquina, esto es, el trabajo no puede empezar su procesamiento antes de que la máquina esté disponible.

La restricción (15) se vincula con el tiempo total de proceso. Las restricciones (16) y (17) determinan el valor correcto de la tardanza (T_j). La restricción (16) determina el valor correcto de la demora (L_j) y (17) específica solamente la parte positiva de la demora como la tardanza.

El conjunto de restricciones (18) – (21) vincula el número de trabajos atrasados, esto es, si la tardanza es mayor que cero, el trabajo está atrasado, en otro caso, este no está atrasado.

El modelo es exacto pero el tiempo de solución crece exponencialmente pues se trata de un problema NP-completo, por lo que se han propuesto métodos heurísticos que aseguran la obtención de soluciones aceptablemente próximas a la óptima.

5. Esquema de solución de la tarea formulada.

Para la solución de la tarea que se formula fueron estudiados numerosos esquemas de solución, entre los que se pueden citar los Algoritmos Genéticos, Redes Neuronales, diferentes algoritmos basados en el método de Ramificación y Acotación, Recocido Simulado, Búsqueda Tabú, entre otros. Finalmente, se adoptó un esquema de solución por aproximaciones sucesivas. En la primera aproximación se realiza, para cada etapa, una partición del conjunto de combinaciones posibles de secuencias de trabajo en *K*

subconjuntos. Para cada partición se genera, un cierto número de opciones de solución prometedoras mediante un algoritmo constructivo, el que busca soluciones que se caracterizan por valores reducidos de la cantidad de trabajos atrasados a partir de una estimación del tiempo mínimo de finalización del trabajo. Como criterio de parada de esta etapa se selecciona la generación de un cierto número (*M*) de secuencias "prometedoras".

El algoritmo de programación que se diseña para secuenciar un conjunto de *n* piezas en un sistema taller de flujo híbrido con *s* etapas se puede examinar como un sistema cerrado en el que entra una secuencia y se obtiene la asignación y secuenciación de las piezas en las máquinas de cada etapa.

Las heurísticas que se implementan se dividen en dos partes: un procedimiento para generar y simular la calidad de operación de un conjunto de opciones de secuencias por etapas, incluyendo la asignación de máquina a cada trabajo de la secuencia. Como algoritmo de generación de secuencias se utiliza el Algoritmo de Búsqueda del Extremo de una Función de un Código Variable [8], el que constituye una aplicación concreta del Método de Integración de Variables. Este algoritmo se aplica a cada una de las *K* particiones del conjunto de permutaciones de trabajos posibles que se realiza para cada etapa. La generación de poblaciones por particiones del conjunto de permutaciones asegura la diversidad de secuencias generadas. Las *r* mejores soluciones generadas en cada etapa son consideradas como conjunto de referencia para la Búsqueda Dispersa.

6. Generación de opciones por el Algoritmo de Búsqueda del Extremo de una Función de un Código Variable.

Los valores iniciales de los códigos variables se generan aleatoriamente, dentro del intervalo de posibles valores del código de solución. La búsqueda de soluciones se realiza de acuerdo al método de Localización del Extremo de una Función de una Variable.

En cada paso de exploración se incluye en la población la mejor solución encontrada, mientras no se cumpla alguno de los criterios de parada adoptados. La actualización de la población se realiza por la comparación del valor de la función objetivo de la mejor entre las dos soluciones generadas en el paso dado (x_h) con el de la peor solución de la población y si este es inferior que aquel, entonces se sustituye esta última solución por x_h. Una vez que es alcanzada la precisión de localización ε prevista se reinicia el proceso de generación de valores aleatorios de x.

En cada iteración se generan 2 códigos variables en el intervalo $x_1 - x_2$. De los 3 subintervalos obtenidos se elimina aquel que contiene el mayor valor de Z y se vuelven a generar otros dos valores dentro del subintervalo obtenido. El proceso se repite mientras la longitud del intervalo $x_A - x_B$ sea superior a una precisión dada. Mientras el tamaño de la población sea inferior a lo establecido se añade a la población, en cada paso, aquella solución con el menor valor de Z entre los 2 códigos generados.

Una vez alcanzado el tamaño requerido de la población se compara el menor valor de Z entre los 2 códigos generados con la peor solución y, en caso de tener menor valor de Z, la solución correspondiente sustituye a la peor de la población.

En el intervalo [1,T] se utiliza el siguiente procedimiento computacional para encontrar el nuevo elemento:

Sean [A,B] = [1, T] y el número entero, llamado Cota de Búsqueda del Mínimo, (CBM), el cual lo define el decisor. $Z_A=f(X_A)$ y $Z_B=f(X_B)$ representan el valor de la función objetivo en los valores A y B respectivamente.

Algoritmo Principal para obtener h, X_h y Z_h

Si B-A es menor que CBM entonces

Si B-A = 1 entonces

Si $Z_A < Z_B$ entonces h = A y $Z_h=Z_A$

En caso contrario h = B y $Z_h=Z_B$

En caso contrario

h = Número aleatorio entre A y B y calcular Z_h

En caso contrario

Busca X_1= Número aleatorio entre A y B-1

Busca X_2= número aleatorio entre X_1 y B

Si $Z_{X1} < Z_{X2}$ entonces h = Aplicar el algoritmo en el intervalo [A, X_2] y obtener Z_h

En caso contrario h = Aplicar este algoritmo en el intervalo [X_1, B] y obtener Z_h

Este proceso se repite hasta encontrar que B-A < CBM. En el caso de que además se cumpla que B - A = 1, se toma h, entre A y B, como el que tiene menor valor de la función objetivo. Si B-A > 1, entonces tomar h como un número aleatorio entre A y B.

Este algoritmo tiene como ventaja que conduce a la obtención de elementos con menores valores para la función objetivo, aunque presenta un problema para el caso en que Z_{X1} ó Z_{X2} en alguna etapa intermedia sea menor que Z_{Xh} para algún X_h que se obtenga una vez que finaliza una corrida del Algoritmo Principal. Con el objetivo de eliminar esta posibilidad, se puede realizar un truncamiento, consistente en asumir que Z_{Xh} es Z_{X1} ó Z_{X2} en el caso en que algún valor de Z_{X1} ó Z_{X2}sea menor que Z_{X_p}. Así mismo, se pueden guardar todos los valores de X y Z_X que se calculan durante la ejecución del algoritmo y tomar como h al que tenga menor valor de la función objetivo.

La función ***Z*** adoptada no requiere ser penalizada, pues las restricciones (2) – (7) se implementan de manera natural en el algoritmo mediante la utilización de las matrices de datos que se le solicitan al decisor y por las características del problema de taller de flujo, mientras que la restricción (8) es de asignación.

En el caso de las restricciones (10) – (14) se realizan arreglos desde el punto de vista computacional para poder satisfacer el cumplimiento de estas, lo que a continuación se fundamenta.

Imposición del tiempo de comienzo de los trabajos.

La etapa *i* no puede comenzar el procesamiento del trabajo en la posición *j* hasta que el trabajo anterior *j – 1* no haya finalizado su procesamiento en dicha etapa, además de que el trabajo en la posición *j* tiene que haber completado su tiempo de operación en la etapa anterior.

Al considerar las restricciones (2), (3) y (10), es posible confeccionar una lista de trabajos en la cual se incluya un tiempo adicional para determinado trabajo, pues la máquina posterior aún no ha sido desocupada.

El esfuerzo computacional puede ser obtenido mediante la imposición de la precedencia del trabajo por el cumplimiento de una de las restricciones, sin la posibilidad de ser excedida. Sea t_j^t que representa el tiempo de comienzo del trabajo *j* en la etapa *t*, entonces se tiene que:

$$tc_1^1 = 0 \tag{22}$$

$$tc_{j+1}^1 = tc_j^1 + Or_j^1 \qquad j = 1,\cdots,n-1 \tag{23}$$

$$tc_1^{t+1} = tc_1^t + Or_1^t \qquad t = 1,\cdots,k-1 \tag{24}$$

$$tc_{j+1}^{t+1} = \max\left\{tc_{j+1}^t + Or_{j+1}^t, tc_j^{t+1} + Or_j^{t+1}\right\} \qquad t = 1,\cdots,k-1;\ j = 1,\cdots,n-1 \tag{25}$$

$$tc_1^1 = 0 \tag{26}$$

$$tc_{j+1}^1 = tc_j^1 + Or_j^1 + s_{j,j+1}^1 \qquad j = 1,\cdots,n-1 \tag{27}$$

$$tc_1^{t+1} = tc_1^t + Or_1^t \qquad t = 1,\cdots,k-1 \tag{28}$$

$$tc_{j+1}^{t+1} = \max\left\{tc_{j+1}^t + Or_{j+1}^t, tc_j^{t+1} + Or_j^{t+1} + s_{j,j+1}^1\right\} \qquad t = 1,\cdots,k-1;\ j = 1,\cdots,n-1 \tag{29}$$

La ecuación (22), expresa que con el objetivo de alcanzar la solución óptima, el trabajo 1 tiene tiempo de comienzo igual a 0. Así mismo, el trabajo en la primera etapa solamente es afectado por el tiempo de completamiento del trabajo anterior (23) y el trabajo en la primera posición de la secuencia, en todas las etapas, también solamente es afectado por su tiempo de completamiento en la etapa anterior (24). Para todos los demás casos, se aplica la restricción (25).

En el caso de existencia de tiempos de preparación, se necesita considerar un tiempo adicional cuando en la etapa *t* el trabajo *l* es precedido del trabajo *j*. Esto trae como consecuencia la necesidad de realizar algunas modificaciones al conjunto de restricciones (22) – (25), las cuales serían de la (26) a la (29).

La utilización de los conjuntos de restricciones anteriores trae como resultado una reducción considerable del tiempo computacional.

El arreglo de tipo múltiple al inicio que se utiliza consiste en solucionar la tarea formulada en diferentes tiempos, donde el tiempo de ejecución es limitado a una fracción de t_{max}, que es la suma del tiempo máximo de ejecución entre dos pasos consecutivos. Durante todo el tiempo las secuencias diferentes se nutren del primer paso, la evaluación con todas las restricciones del modelo de permutación, solamente si su valor correspondiente de la función objetivo es determinado. Esta operación requiere de poco tiempo, comparada con el segundo paso en el cual se determina la mejoría, a partir de considerar la resecuenciación después de la primera etapa.

La secuencia en la segunda etapa se fija por una de las secuencias obtenidas en la primera etapa. Esto trae como consecuencia, que es posible considerar secuencias de permutaciones que no sean óptimas dentro del conjunto de secuencias de permutación, pero las cuales dan como resultado una relativamente mejor solución final después de que se aplica el segundo paso. Este arreglo se aplica con la concepción de que estas secuencias se generan de manera aleatoria y por esta razón no están sujetas a ningún patrón

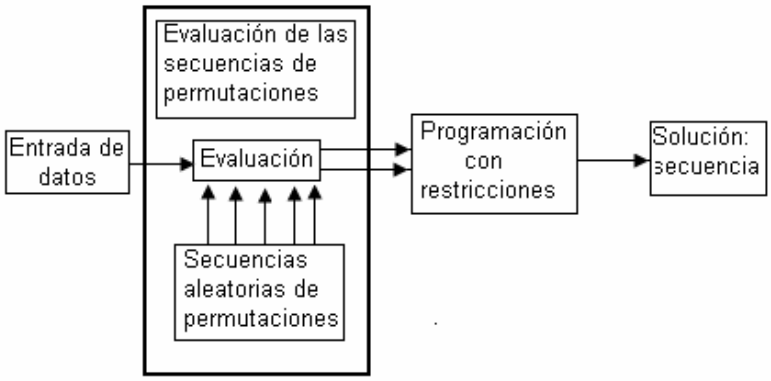

Figura 1. Esquema para las secuencias de permutación generadas aleatoriamente que son evaluadas con el modelo de permutación con todas las restricciones.

En este caso (figura 1), se genera una determinada cantidad de secuencias de permutaciones aleatoriamente, las que se evalúan en el modelo con todas las restricciones. Las secuencias más promisorias se agrupan y son las que se consideran con posibilidad para ser seleccionadas por el decisor para pasar al próximo paso, donde se realiza la conciliación con el objetivo de determinar la secuencia correspondiente. Este proceso se realiza en cada una de las etapas, por lo que al final se obtiene el conjunto de secuencias para la tarea formulada.

Simulación del proceso de carga de las máquinas.

Se utilizan las siguientes reglas básicas de prioridad: una vez cargada una máquina con un trabajo, ese trabajo se finaliza en la etapa antes de cargar esa máquina con un próximo trabajo.

Los trabajos son asignados a las diferentes máquinas, en el orden de la secuencia simulado tan pronto como sea posible a las máquinas. Si se hace posible cargar más de una máquina con el mismo trabajo se elige la máquina que aporta el menor valor del tiempo de holgura estimado para todas las etapas. La holgura del trabajo se calcula como su fecha de entrega menos el tiempo de procesamiento requerido para dicho trabajo, suponiendo que a partir de la etapa dada no existen tiempos de espera por máquina.

La regla Menor Tiempo de Procesamiento (*Shortest Processing Time*, SPT) utilizada en la simulación del paso por cada una de las etapas es una regla simple de prioridad en la cual los trabajos son secuenciados en un orden creciente de los tiempos de procesamiento. En el modelo de una sola máquina con la regla SPT se prueba que la suma de tiempos de la finalización y la demora es reducida al mínimo. El razonamiento es guardar trabajos con tiempos de procesamiento cortos para ser asignados secuenciados sin afectar el equilibrio de la carga de trabajo. Sin embargo, debido a los problemas en un taller de flujo flexible con máquina paralela sin relación, se tiene que adaptar la regla SPT utilizando el tiempo de funcionamiento como se declara anteriormente. Entonces, la mejor solución, entre las generadas en la primera etapa, se selecciona entre las combinaciones de las velocidades relativas de la máquina y los tiempos de preparación.

La regla Fecha de Entrega Más Temprana (*Earliest Due Date first*, EDD) programa los trabajos de acuerdo a un orden creciente de las fechas de entrega y tiende a minimizar la tardanza máxima.

La regla Mínimo Tiempo de Holgura (*Minimum Slack Time first*, MST) es una variación de la regla EDD, esta se refiere a la holgura restante de cada trabajo, definida como su fecha de entrega menos el tiempo de procesamiento requerido para procesarla.

Así, las restricciones (2) – (28), (11), (15) – (17) quedan satisfechas automáticamente de acuerdo al método de solución adoptado; las restricciones (9), (20) y (21) aportan algoritmos de cálculo que se ejecutan durante la simulación del paso de la secuencia por la etapa de fabricación. Las restricciones (10), (12), (14), (18) y (19) se relacionan con el orden en que los trabajos pueden realizarse en la misma etapa y entre las etapas, por lo que se aseguran por el procedimiento de paso de la secuencia por las etapas.

Paso de una etapa a la siguiente.

Con el fin de mejorar, en lo posible la calidad de las soluciones generadas se elaboró el siguiente algoritmo, basado en el método de Búsqueda Dispersa [9,10].

Entre las *r* mejores soluciones generadas en la etapa anterior se seleccionan 2 de ellas aleatoriamente y se genera un código aleatorio intermedio al correspondiente de ambas soluciones generadas, de acuerdo a la lista de permutaciones originalmente elaborada. Se evalúa la función objetivo y si su valor es inferior al mayor entre las *r* mejores soluciones, entonces se selecciona la nueva solución y se elimina la peor entre las disponibles. Este proceso se repite hasta que se cumpla uno de los criterios de parada seleccionados.

Una vez generadas las *r* soluciones con los menores valores alcanzados de la aproximación a la función de utilidad utilizada, se pasa a la simulación de las condiciones de paso de cada una de estas secuencias por las sucesivas etapas posteriores. La simulación se realiza de igual forma a lo descrito para la primera etapa, rectificando el valor de *Z* para cada una de las secuencias ya generadas.

El refinamiento de las soluciones, es decir, el recálculo de la función de utilidad de las secuencias, la generación de nuevas secuencias "prometedoras" por búsqueda dispersa y el reordenamiento de las soluciones generadas durante la etapa anterior y las nuevas generadas en la etapa. La generación de nuevas soluciones "prometedoras", se realiza mediante:

- La sustitución de los elementos que se han obtenido, por algún elemento, si existe, perteneciente a cualquier vecindad de dicho elemento seleccionada de forma aleatoria, y que posea un valor menor de la función objetivo.
- La utilización de un procedimiento de búsqueda de elementos que pertenezca a las vecindades de los individuos de la población generada con un valor menor de la función objetivo, para sustituir a cada uno de los elementos anteriores.

Entre todas las secuencias procesadas se eligen para la simulación de su paso por la etapa solo las primeras *r*, en el orden de la función de utilidad.

Conocido que la suma del retraso total y el tiempo de finalización total responden a unidades diferentes, los factores de la función se deben normalizar. Para lo cual, se calcula inicialmente, el retraso máximo y tiempo de proceso máximo para una secuencia formada por las piezas ordenadas según orden creciente del número de pieza. Estos valores permiten calcular los valores de los parámetros que forman parte de la función objetivo (1).

Los valores $A_T max$ y *Cmax* se calculan para cada ejemplar una única vez y son independientes del procedimiento de resolución utilizado. Se calculan *Cmax* y $A_T max$ para cada ejemplar y se incorpora la función objetivo dada por la ecuación cuando se tenga que valorar una secuencia dada.

Para obtener una función de calidad que permita ordenar los códigos de acuerdo a los valores de la función objetivo, se utiliza la distancia ponderada de Tchebycheff entre los valores calculados para cada indicador y su valor deseado.

Criterios de parada del procedimiento de optimización.

Como criterios de parada del proceso de generación de opciones de secuencia, se utilizan los establecidos tradicionalmente:

i. Realizar una determinada cantidad de mejoras.
ii. Aplicar el algoritmo hasta que se satisfaga que $\left|Z_{X_P} - Z_{X_h}\right| < \varepsilon$, donde ε es un valor positivo prefijado.
iii. Realizar mejoras hasta alcanzar un determinado por ciento de la población.

Según experimentos numéricos realizados por el autor, la aplicación del algoritmo principal a un por ciento de toda la población, equivale, en la mayoría de los casos al muestreo exhaustivo de la población.

Se selecciona una muestra, constituida por un 20% de los elementos favorables que se obtienen con anterioridad.

Por otra parte, se deben tomar medidas para garantizar la participación de individuos de todo el intervalo posible para las soluciones, pues el método está basado en soluciones aleatorias

Como resultado de la minimización, para diferentes combinaciones de valores de λ, se muestrea el espacio de soluciones eficientes, es decir, soluciones tales que son no peores a las restantes, del conjunto de posibles soluciones, para al menos uno de los indicadores.

En la función objetivo es necesario definir los valores deseados para cada uno de los indicadores. El valor deseado para C_{max} se consideró igual a la suma de los tiempos de operación estándar conjuntamente con los tiempos de preparación inherentes a la secuencia y a las máquinas y para la cantidad de trabajos atrasados es posible asumir cero o uno de acuerdo a su estado.

7. Conclusiones.

En el trabajo se presenta un modelo matemático adecuado para los talleres de producción de piezas, con estructura de taller de flujo, para realizar la conciliación de las secuencias de trabajo en las áreas de producción de piezas, a partir de considerar la tarea de secuenciación de la fabricación como integrante de la tarea mayor de Integración del diseño, la generación de tecnologías, la fabricación y la planeación de la producción de piezas maquinadas.

Así mismo, se introduce una metaheurística para la solución de la tarea formulada, basada en la integración de las técnicas propias del Algoritmo de Búsqueda Aleatoria del Extremo de una Función de un Código Variable, del método de Integración de Variables y del método de

Búsqueda Dispersa con el que se logra reducir considerablemente el número de opciones de solución generadas en el proceso de búsqueda con respecto a otros algoritmos disponibles.

8. Referencias.

1. **BIN, W.** "*CAD. Interaction with CAPP". Manufacturing Systems Design and Analysis*. Brunel University, Uxbridge, UK: Chapman & Hall, 1994.
2. **JWATA, K.** *Knowledge based computer aided process planning in intelligent manufacturing systems.* Milacic, Amsterdam: Ed. V. R., 1986.
3. **MILNER, D. A. y VASILIOU.** *"The integration of CAD and CAM". Computer – Aided Engineering for Manufacture.* London, UK: Kogan Page Ltd, 1994.
4. **MITCHELL, F. H**. CIM *Systems. An introduction to Computer Integrated Manufacturing*. New Jersey: Prentice Hall, 1991.
5. **ÁVILA, R.** *Generación de Tecnología para Máquinas Herramienta con Control Numérico Computarizado.* Tesis en opción al grado científico de Dr. en Ciencias Técnicas, Universidad de Holguín, Holguín. Cuba. 1999.
6. **WILLIAMS, D. J.** *Manufacturing Systems. And introduction to the technologies*. UK: Chapman-Hall, 1994.
7. **ARZOLA, R. J.** *Sistemas de Ingeniería*. La Habana: Editorial Félix Varela, 2000.
8. **ARZOLA, R. J.; SIMEÓN, R. E.,** *et al*. El Método de Integración de Variables: una generalización de los Algoritmos Genéticos. *En Proceeding Intensive Workshop: Optimal Design of Materials and Structures. París, Francia. 2003*.
9. **GLOVER, E.; KELLY, J. P.,** *et al*. "Genetic algorithms and Tabu Search. Hybrids for optimization". *Computers and Operations Research.* 1995.
10. **LAGUNA, M. y MARTÍ, R.** *Scatter Search: Methodology and Implementations in C*. Kluwer Academic Publishers, 2003.

Alberto Fiol-Zulueta[1], José Arbola-Ruíz[1], Adriano da Silva-Carvalho[2]

1. Departamento de Matemática. Facultad de Ingeniería Mecánica
E-mail: tcarrasco@mecanica.ispjae.edu.cu, jararzola@mecanica.ispjae.edu.cu
Instituto Superior Politécnico "José Antonio Echeverría" – CUJAE
Calle 114 #11901 e/119 y 127. Marianao. La Habana. CP 19390. Cuba.

2. Facultad de Ingeniería, Universidad de Oporto. Portugal
E-mail: acs@fe.up.pt