

PROGRAMACIÓN DE OPERACIONES EN DOS ETAPAS PARA UN FLOWSHOP MULTIPERÍODO NO TRADICIONAL¹

TWO-STAGE OPERATIONS SCHEDULING FOR A NON-TRADITIONAL MULTI-PERIOD FLOW SHOP

PROGRAMAÇÃO DE OPERAÇÕES EM DUAS ETAPAS PARA UM FLOW SHOP MULTIPERÍODO NÃO TRADICIONAL

Juan Pablo Orejuela-Cabrera

M.Sc. en Ingeniería Industrial. Universidad del Valle, Colombia. Colombia. Investigador principal. Grupo de Investigación: Logística y Producción.

Profesor tiempo completo. Escuela de Ingeniería Industrial, Universidad del Valle, Cali - Colombia

juan.orejuela@correounivalle.edu.co

RESUMEN

En el presente documento se desarrolla una metodología de dos etapas para programar las operaciones en un flowshop multiperíodo, en éste se tienen trabajos que aunque que se deben terminar en la misma ventana de tiempo, unos son de entrega inmediata y deben programarse en el momento más temprano de la ventana, y otros en el momento más tardío, de tal modo que se minimice su tiempo de permanencia en el sistema. Se plantea una estrategia de descomposición temporal para el problema multiperíodo, en la que se combina la programación estructurada con la programación lineal, de tal modo que para cada periodo se corren dos fases compuestas de dos modelos matemáticos que programan los trabajos según su prioridad. La metodología planteada se valida en un problema de programación de trabajos en la industria de alimentos concentrados, obteniéndose como resultado un scheduling para cada periodo que satisface los requerimientos de los productos de entrega inmediata y los de entrega en el momento más tardío. Del tal modo que se minimiza para los primeros el inventario de producto en proceso y para los segundos el tiempo de no permanencia en el sistema.

PALABRAS CLAVE

Flowshop multiperíodo, modelación matemática, programación de operaciones, planeación jerárquica, *Scheduling*.

ABSTRACT

This documents presents a detailed review of a two-stage approach to scheduling operations in a multi-period flow shop where various tasks have to be completed in the same time window. Some tasks are delivered immediately and must be scheduled as early as possible within the time window. Others are carried out in the latest stage in order to minimize their time of permanence in the system. It proposes a time decomposition strategy to address the multi-period issue, combining structured programming and linear programming in such a way that, for each period, two stages are run consisting of two mathematical models that schedule tasks based on their priority. The proposed approach was validated through the resolution of the problem of scheduling work for a feed manufacturing company where a schedule was prepared for each time period, meeting the requirements for immediate delivery products and those which can be delivered at a later point in time. The schedule minimizes the work-in-progress inventory of the first and the time of non-permanence in the system of the latter.

KEYWORDS

Multi-period flow shop, mathematical modeling, operations scheduling, hierarchical planning, scheduling.

RESUMO

No presente documento, desenvolve-se uma metodologia de duas etapas para programar as operações em um flow shop multiperíodo, nele há trabalhos que embora devem terminar na mesma janela de tempo, uns são de entrega imediata e devem programar-se no momento mais cedo da janela e outros no momento mais tardio, de tal modo que se minimize seu tempo de permanência no sistema. Apresenta-se uma estratégia de descomposição temporal para o problema multiperíodo, na qual se combina a programação estruturada com a programação linear, de tal modo que para cada período correm duas fases compostas de dois modelos matemáticos que programam os trabalhos conforme sua prioridade.

A metodologia apresentada valida-se em um problema de programação de trabalhos na indústria de alimentos concentrados, obtendo-se como resultado uma programação para cada período de tempo que satisfaça os requerimentos dos produtos de entrega imediata e os de entrega no momento mais tardio. Del tal modo que se minimiza para os primeiros o inventário de produto no processo e para os segundos o tempo de não permanência no sistema.

PALAVRAS-CHAVE

Flowshop multiperíodo, modelação matemática, programação de operações, Planejamento hierárquico, programação.

Introducción

La planeación y el control de las operaciones es un proceso que en términos generales busca definir la mejor manera de asignar los recursos para atender los requerimientos de demanda, de tal modo que se cumplan los objetivos de la compañía (Noori y Radford, 1997). Esto implica que gran parte del éxito de la organización depende del manejo que tenga este proceso.

Sin embargo, a pesar de los grandes esfuerzos que se hacen por realizar bien el proceso, es común que se presenten múltiples dificultades que se ven reflejadas en algunos problemas tales como, incumplimiento de los clientes, sobrecostos en la producción, faltantes o excesos de inventarios y capacidad (Vollmann, Berry y Whybark, 1995), lo cual aleja a las organizaciones del cumplimiento de sus metas de largo, mediano y corto plazo.

Las dificultades y problemas anteriores son atribuibles al hecho de que el problema de planear y controlar las operaciones es algo muy complejo, ya que involucra al mismo tiempo diferentes horizontes de planeación, diferentes tomadores de decisiones, diferentes tipos de recursos y diferentes tipos de requerimientos (Graves, 2002).

A lo anterior se suma que cada ambiente de fabricación tiene sus propias particularidades: los métodos de producción, la configuración del sistema productivo, la variedad de productos y las partes que los componen, los tipos de procesos, las características de la demanda, ya sea determinística o aleatoria, el comportamiento

y robustez del sistema productivo y demás elementos que agregan complejidad al problema (Harhalakis, Nagi y Proth, 1993).

Uno de los enfoques planteados para resolver el problema de la planeación y el control de la producción es el enfoque jerárquico (Motoa, Osorio y Orejuela, 2007), que con el propósito de darle tratamiento a la complejidad, busca descomponer el problema de planeación en subproblemas subordinados, de tal modo que éstos tengan una menor complejidad que la del problema original, lo que garantiza que al resolver todos los subproblemas quede solucionado el problema original. Otras aplicaciones de este enfoque se pueden ver en (Lindholm, Lisdestam y Quttineh, 2014), (He; Shang, y Li, 2014), (Tian, Willems y Kemp, 2011), (Aghezzaf, Sitompul y Van den Broecke, 2011), (Torabi, Ebadian y Tanha, 2010) ,(Reza y Ghaderi, 2010) entre otros.

Generalmente este proceso de descomposición toma en consideración la estructura jerárquica de la compañía y el horizonte de decisión. En ese sentido se pueden identificar en la bibliografía diversos subproblemas para el problema de planeación y control de la producción, sin embargo muchos autores (Bitran, Haas y Hax 1982), (Chen y Ji, 2007), (Saad, 1990) coinciden en los siguientes: planeación a largo, a mediano y a corto plazo, o planeación estratégica, táctica y operativa, respectivamente.

Cada una de estas fases o subproblemas guardan entre sí relaciones de jerarquía, flujos de información y retroalimentación que garantizan que la solución de cada uno de ellos implique la solución del problema global (Tischer y Carrion 2003).

La planeación operativa es subordinada de la planeación táctica, por tanto está sujeta a cumplir con los requerimientos que ésta le imponga (instrucciones o restricciones). A su vez, la planeación táctica está obligada a enviar instrucciones que sean factibles en lo operativo y para ello debe anticipar los impactos del requerimiento mediante la agregación del problema operativo. Y en caso de que la anticipación no sea suficiente para garantizar factibilidad, el nivel operativo puede reaccionar mediante una retroalimentación al nivel táctico. Esta misma relación se da entre la planeación estratégica y la táctica, siendo la última subordinada de la primera.

Sobre la base de los elementos expuestos de la planeación jerárquica de la producción, en la presente investigación se desarrolla una metodología para abordar el subproblema de la planeación operativa en un ambiente tipo flowshop, en que ya se han realizado los subproblemas de planeación estratégica y táctica. De manera puntual, el plan táctico, que subordina el problema a tratar, consiste en un plan de producción en el que se han definido las cantidades a fabricar de cada producto, en cada momento de tiempo t , para atender la demanda actual o futura en el momento de tiempo s .

Lo anterior supone entonces que se conoce qué cantidades de productos se deben fabricar en t , para entregar en el mismo periodo y cuáles se deben fabricar en t , para entregar en periodos futuros. Esto implica que en todo periodo se tengan dos conjuntos de órdenes que se deben programar en la misma ventana de tiempo, las de entrega inmediata, por tanto de mayor prioridad, y las de entrega futura con menor prioridad. Para la solución de este problema, en la presente investigación se emplea una estrategia de descomposición temporal, en la que mediante un algoritmo de programación estructurado, en cada ventana de tiempo se corren dos modelos: uno para los productos de entrega inmediata y otros para los de entrega futura.

Para abordar lo anterior el documento se ha estructurado de la siguiente forma: en el numeral 1 se presenta la caracterización del ambiente y el problema abordado. En el numeral 2 se presenta el desarrollo metodológico y cada una de sus fases, en el numeral 3 se presenta el caso de estudio y los principales resultados, finalmente se consideran las conclusiones y la bibliografía.

1. Ambientes de fabricación y problema planteado

El *Scheduling* o planeación a corto plazo especifica la organización de la producción de acuerdo con las órdenes emitidas en el plan táctico, considerando la disponibilidad de recursos y restricciones, controlando las operaciones y ajustando la capacidad del sistema para asegurar que se cumplan los requerimientos del plan táctico (Morton y Pentico, 1993) y (Sipper y Bulfin, 1998) Esto implica las siguientes actividades: Asignación de trabajos a las máquinas, determinación de la secuencia de realización de las actividades y programación de las fechas de comienzo y finalización de las operaciones.

Los problemas de *Scheduling* son muy comunes, y no solamente están inscritos dentro de la manufactura, sino que están presentes en diferentes escenarios. Por ejemplo, en la programación de proyectos, la de tripulaciones, la de un grupo de aviones esperando pista para aterrizar y muchas otras posibilidades más. (Conway, Maxwell y Miller, 1967) Algunos de los casos son resueltos aproximadamente, de acuerdo con el momento en que se presentan las actividades (FIFO), razonamiento lógico en algunos ambientes pero no en todos. Por lo general en el resto de los casos, que son la mayoría, para resolverlos se debe determinar la mejor secuencia de trabajos y para ello se podrían enumerar todas las secuencias posibles y elegir aquella que optimiza alguna medida de desempeño.

Son numerosas las investigaciones planteadas alrededor del problema de *Scheduling* pero todavía este problema sigue y seguirá presentando desafíos de investigación por la gran complejidad que puede llegar a presentar (Smith, 2003), (Montenegro, 2005), (Márquez, 2012) (Álzate, 2010) (Osorio, Castrillón, Toro y Orejuela, 2008), (Li e Ierapetritou, 2010), (Szoke, 2011), (Sinnen, 2014), (Cheng, Tadikamalla, Shang y Zhang, 2013), (Zhong, Ou y Wang, 2014), (Wan y Zhang, 2014), (Schwartz y Rivera, 2010) y (Tang, Lui y Sun, 2014) entre otras. Es por ello que para facilitar su comprensión e identificación se plantea que un problema de *Scheduling* se describe por una tripleta $\alpha / \beta / \gamma$. En el campo α se describe el ambiente de máquinas y contiene una entrada. El campo β provee detalles de las características de procesamiento y restricciones; puede contener múltiples entradas o ninguna. El campo γ describe el objetivo que se minimizará y contiene generalmente una entrada. A

continuación, con el uso de la descripción planteada se caracteriza el problema de *Scheduling* abordado en la presente investigación.

- Campo α : Flow shop (F_m). Se tienen “ m ” máquinas en serie y cada trabajo debe ser procesado en cada una de las “ m ” máquinas. Los trabajos siguen la misma ruta. Una vez terminado el procesamiento de un trabajo en una máquina pasa a la cola de la máquina siguiente, de tal modo que ningún trabajo pasa a otro que esté en la fila.
- Campo β : Fechas de liberación (r_j). Todo trabajo j no puede iniciar su proceso antes de la fecha r_j . Estas fechas son tomadas del nivel táctico que indica en qué ventana de tiempo se debe fabricar un producto y en qué ventana de tiempo debe ser entregado, con lo que se tienen múltiples trabajos por ventana de tiempo de fabricación a ser entregados en diferentes momentos.
- Campo γ : Los trabajos de entrega inmediata deben ser programados de tal modo que se minimice el tiempo de terminación total ponderado (wF) y los trabajos de entrega posterior a la ventana de tiempo se deben programar de tal modo que estén el menor tiempo posible como inventario de producto final. Para ello se propone, como medida de desempeño, el complemento del tiempo de terminación total ponderado.

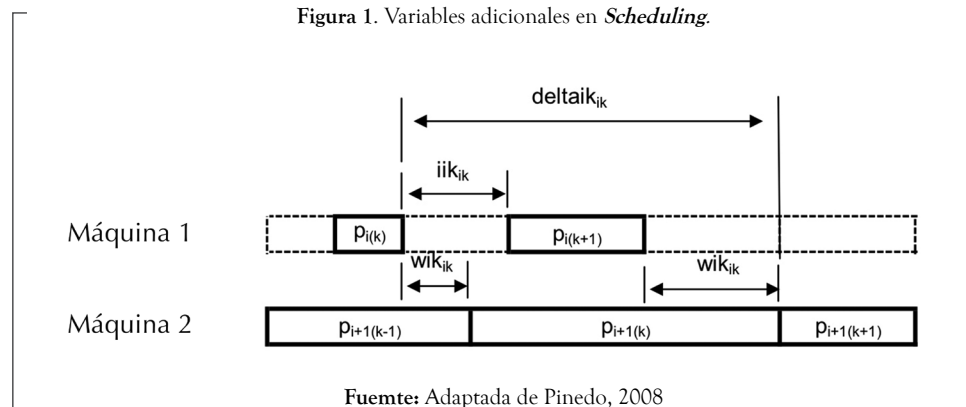
A continuación se muestran los principales elementos involucrados en la representación del ambiente de fabricación, en este el horizonte de planeación está constituido por un conjunto de T períodos semanales consecutivos, $TIEMPO = \{1 \dots T\}$, cada período de tiempo $t \in TIEMPO$, tiene una duración de 48 horas y las horas son las ventanas de tiempo elementales empleadas, el horizonte de planeación en horas está dado por $48 \times T$

Trabajos: El plan de producción lo conforman dos tipos de trabajo, los de prioridad uno y los de prioridad dos, y cada grupo se programa por separado, siendo los trabajos de prioridad uno los de consumo inmediato y por consiguiente se deben programar primero. Los de prioridad dos se almacenarán en inventario y se deben programar en el momento más tardío posible. Cada trabajo está conformado por un subconjunto de M operaciones, una para cada máquina, y cada operación tiene su respectivo tiempo de proceso, que se representa como $TIME_{jm}$ que indica el tiempo de proceso de la operación m del trabajo j .

Máquinas: Son las encargadas de las operaciones de los trabajos y al ser el ambiente de fabricación un flowshop, es posible que en una máquina una vez procesado el k -ésimo trabajo no se inicie de inmediato el procesamiento del trabajo en la posición $(k+1)$ -ésima, esto porque la máquina anterior todavía no lo ha desocupado. El tiempo ocioso que transcurre en una máquina entre la finalización del procesamiento del trabajo k -ésimo y el inicio del procesamiento del trabajo $(k+1)$ -ésimo se representará como iik_{ik} . (Ver Figura 1). El tiempo que espera el trabajo en la k -ésima posición desde que termina su proceso en la máquina i e inicia su proceso en la máquina $(i+1)$ se representará por la variable wik_{ik} . (Ver Figura 1).

En el nivel táctico se establece para cada producto y para cada momento de tiempo las cantidades que serán fabricadas para ser consumidas en el mismo período, como las que serán fabricadas para consumos en períodos posteriores. Estas cantidades se indican como $x1_{ptu}$ que representa la cantidad a fabricar del producto “ p ” en el período t para atender la demanda en el período u . lo cual representa una instrucción para el nivel operativo que debe desarrollar un programa de producción que atienda tales requerimientos en todos

Figura 1. Variables adicionales en *Scheduling*.



los periodos. El problema de programación resultante es un *Scheduling* con fechas de liberación diferentes de cero, con un horizonte de planeación igual al del plan táctico.

El problema de *Scheduling* con fecha de liberación diferente de cero, es considerado extremadamente no polinomial, duro para los ambientes flowshop (Pinedo, 2008). Dado que el problema cubre todo el horizonte de planeación y todos los productos, el número de trabajos a programar es alto, lo cual adiciona más complejidad al problema. Por otro lado, el *Scheduling* deseado presenta requerimientos no tradicionales, ya que se busca que los trabajos con prioridad uno se programen en su momento más temprano y los de prioridad dos en el más tardío posible.

Tal requerimiento no tradicional se justifica en la medida que los trabajos de prioridad uno atenderán la demanda del mismo período y, por otro lado, los trabajos de prioridad dos o los que atenderán demandas futuras se almacenarán e inicialmente ocupan espacio en la línea de producción y posteriormente en la bodega de producto terminado, por lo que se busca que la fabricación de los productos sea lo más cercana a su consumo y además se trata de evitar al máximo el inventario en la línea de producción y en la bodega. Para abordar los elementos descritos del problema y la particularidad de sus requerimientos a continuación se presenta la propuesta metodológica.

2. Desarrollo metodológico

Para abordar la complejidad computacional del problema se propone realizar una estrategia de descomposición temporal (Pinedo, 2008), la cual consiste en resolver problemas aislados de *Scheduling* para cada período en el horizonte de planeación. Lo anterior no solo reduce la complejidad asociada al

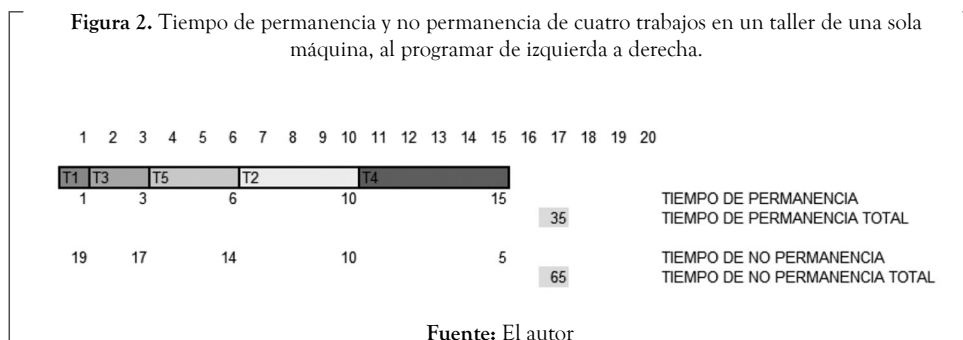
número de trabajos en *Scheduling*, sino que permite convertir el problema T sub problemas. Sin embargo, a pesar de que la descomposición reduce el número de trabajos y genera problemas menos complejos, los requerimientos no tradicionales del problema generan que en cada problema de tiempo se tengan dos grupos de trabajo con diferentes tipos de prioridades, en ese sentido también se propone separar la programación del nivel detallado en dos fases, según la prioridad de los trabajos. Se describe aquí cada una de las fases y la forma en que interactúan.

FASE 1

Programación de los trabajos con prioridad uno: La estructura de dos fases empieza con la programación de los trabajos de consumo inmediato. En esta fase se busca resolver un problema de *Scheduling* con trabajos que tienen todos el mismo tipo de prioridad, la misma fecha (cero) de liberación y la misma fecha de terminación. Este problema es menos complejo y para el número de trabajos de cada ventana de tiempo se puede resolver mediante un modelo de programación lineal entero-mixto.

La función objetivo para esta fase es la minimización del costo de mantenimiento del inventario, la cual se establece como el tiempo de flujo total tradicional en los problemas de *Scheduling* multiplicado por el costo de tenencia por hora por cada unidad en el taller. (Ver Figura 2).

El tiempo de permanencia de un trabajo en el taller está dado por la diferencia entre el tiempo de liberación, que es cero, y el tiempo de terminación del trabajo. El tiempo de no permanencia o complementario está dado por la diferencia entre el horizonte de planeación o fecha de entrega y la fecha de terminación del trabajo. Para el trabajo 1 (T1) en la gráfica anterior su tiempo de terminación es 1, por consiguiente el tiempo



de permanencia es 1 y su tiempo de no permanencia dado un horizonte de 20 es 19.

FASE 2

Programación de los trabajos con prioridad dos: Una vez realizado el programa de producción para los trabajos de prioridad uno, se procede a hacer lo mismo con los trabajos que serán empleados para atender las demandas futuras. De la fase 1, se tienen las fechas en que las máquinas quedan disponibles, además se conoce cuál es la capacidad de cada máquina en la ventana de tiempo. Para realizar la programación de los trabajos de prioridad dos se encuentra la diferencia entre las anteriores y con ello se obtiene el horizonte de programación para los trabajos de prioridad dos.

El problema de *Scheduling* en la fase dos se resuelve con la misma estructura de restricciones de modelo de la fase uno, pero la representación y la lectura de los resultados se hacen diferentes, ya que se busca programar los trabajos en su momento más tarde posible del horizonte de planeación, es decir, programación hacia atrás, los resultados obtenidos se implementan de derecha a izquierda, el trabajo que sale en el *Scheduling* a programar de primero de izquierda a derecha, se programa de último pegado al final del horizonte de planeación.

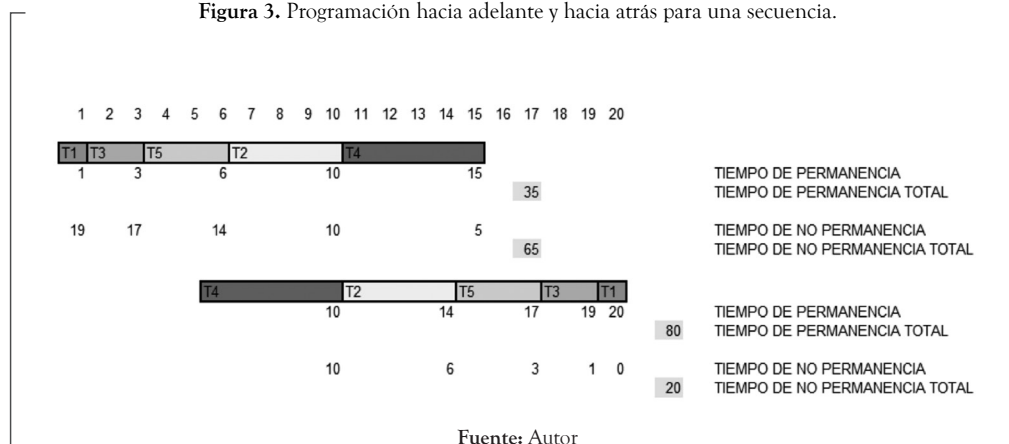
En la Figura 2 se representa un programa óptimo para el tiempo de flujo total en una sola máquina, con su respectiva programación hacia adelante, de izquierda a derecha. En caso de que ese programa fuera óptimo en la fase 2 su implementación se realizaría de derecha a izquierda en programación hacia atrás, lo cual se puede ver en la Figura 3.

Emplear la secuencia óptima de la programación hacia adelante y emplearla como solución al implementarla hacia atrás, permite minimizar el tiempo de no permanencia de los trabajos en el taller, es decir, minimizar el tiempo de permanencia en una secuencia de programación hacia adelante es equivalente a minimizar el tiempo de no permanencia en una secuencia de programación hacia atrás. Como en la fase dos se trata de minimizar el inventario de producto terminado en la línea de producción, esto es equivalente a minimizar el tiempo de no permanencia, la función objetivo se formula como el costo de inventario de los trabajos terminados en la línea de producción.

En la Figura 3 el tiempo de permanencia total de los trabajos programados hacia adelante es de 35 unidades de tiempo y el tiempo de no permanencia total de los trabajos programados hacia atrás es de 20 unidades de tiempo, Hay una relación entre el tiempo de permanencia de los trabajos programados hacia adelante y el tiempo de no permanencia de los trabajos programados hacia atrás, la cual se plantea en el siguiente párrafo.

El tiempo de permanencia de una secuencia de trabajos programados hacia adelante, TPSAD, menos el tiempo de proceso de tales trabajos, TP, es igual al tiempo de no permanencia de los mismos trabajos programados con la misma secuencia pero hacia atrás, TNPSAT. Tal relación se puede expresar así: $TNPSAT = TPSAD - TP$. Por ejemplo, en la Figura 3, la suma de los tiempos de proceso de los trabajos es 15 y al restar los 35 de tiempo de permanencia del programa hacia adelante, da 20, que es el tiempo de no permanencia del programa hacia atrás.

Figura 3. Programación hacia adelante y hacia atrás para una secuencia.



Si bien es cierto las observaciones anteriores anteriormente funcionan para el caso de una única máquina, se hace la extensión al ambiente de fabricación, por tratarse de flowshop permutación, ya que el principio dominante en la secuencia será el mismo, Además, se asume que los trabajos de una misma fase no presentan grandes diferencias entre sus tiempos de proceso -por las decisiones del nivel táctico-, lo cual también mejora la aproximación.

Como resultados finales de las dos fases salen un par de programas hacia adelante y hacia atrás, que cumplen con los requerimientos no tradicionales del problema de *Scheduling*. Este mismo proceso se debe realizar para cada uno de los períodos en el horizonte de planeación del nivel táctico. Lo anterior consolida la propuesta de descomposición temporal del problema del *Scheduling* inicial. (Ver Figura 4).

MODELO MATEMÁTICO PARA LAS FASES 1 Y 2

Para abordar el problema de *Scheduling*, la propuesta metodológica plantea dos fases, Cada una de ellas tiene asociado un modelo matemático de programación lineal entero mixto, este par de modelos tienen la misma estructura, desde el punto de vista de las restricciones, pero en la función objetivo surgen algunas diferencias dado el significado que tiene una solución en cada modelo. A continuación se presenta el modelo base de las dos fases y, por separado, la función objetivo de cada una de las fases.

CONJUNTOS

- Máquinas: Conjunto de máquinas disponibles en el flowshop. El índice que se empleará es m .
- JOBS: Conjunto de todos los trabajos a programar, el índice que se empleará será j para este conjunto.

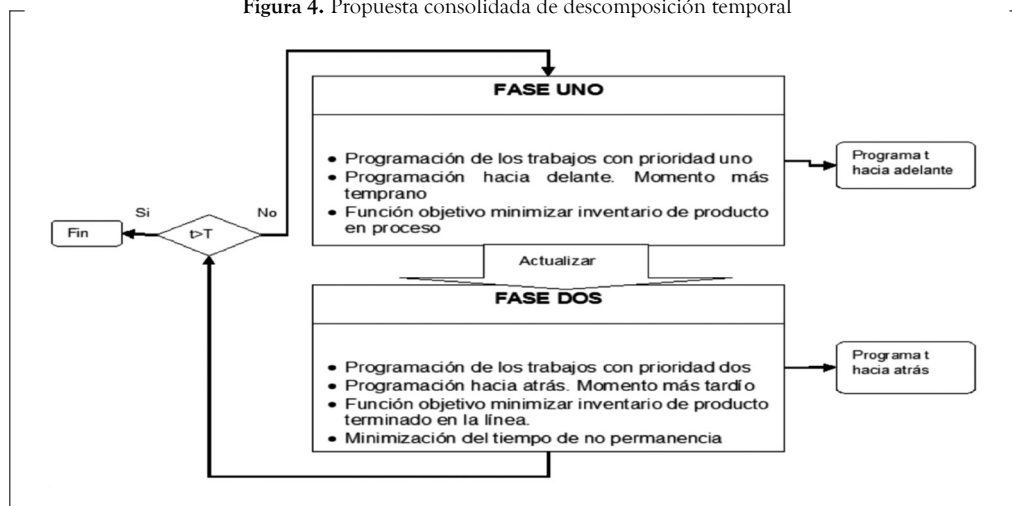
PARÁMETROS

- NBJOBS: Cantidad de trabajos a programar en el flowshop
- M : Cantidad de máquinas en el flowshop.
- FREMAQO: Cantidad inicial de tiempo disponible para la realización del *Scheduling* en la ventana de tiempo.
- CMANWIP1: Costo de tener en inventario de producto en proceso un trabajo de una unidad de tiempo.
- $TIME_{mj}$: Tiempo que se demora el trabajo j en la máquina " m ".
- WEITH: Peso de trabajo asociado a la prioridad, si tiene prioridad uno su peso es 1, lo cual indica que se programa en la fase 1 y si tiene un dos debe programarse en la fase 2.

VARIABLES DE DECISIÓN

- $x_{jk} \geq 0$: Variable binaria que vale uno si el trabajo j se programa en la secuencia k y cero de lo contrario.

Figura 4. Propuesta consolidada de descomposición temporal



Fuente: El autor

- $iik_{ik} \geq 0$: Variable continua, que indica el tiempo ocioso que experimenta la máquina i , desde cuando termina de procesar el trabajo en la posición k -ésima de la secuencia, hasta el momento en que inicia el proceso del trabajo en la posición $(k+1)$ -ésima de la secuencia.
- $wik_{ik} \geq 0$: Variable continua que indica el tiempo que espera el trabajo en la k -ésima posición de la secuencia, desde el momento en que se termina la operación en la máquina i y el inicio de la operación siguiente en la máquina $i+1$.
- $cj_j \geq 0$: Variable continua que me indica el tiempo de terminación del trabajo en la j -ésima posición de la secuencia.
- $fremaq_m \geq 0$: Variable continua que me indica el tiempo en que la máquina "m" queda libre, una vez procesa todos los trabajos.
- $makespan \geq 0$: Variable continua que me indica cuál es el tiempo de terminación del último trabajo.
- Lo anterior es explicable dado que si la máquina i tuvo que esperar para iniciar el proceso del trabajo en la posición $k+1$ de la secuencia, era porque este trabajo no se había terminado en la máquina $i-1$, por tal motivo al terminarse en esa máquina ($i-1$) y al estar ociosa la i , es natural que pase de una a procesar en la máquina i .
- Otro elemento que se debe garantizar es la relación entre la variables físicas de problema y las funcionales, es decir, las variables binarias con el resto de variables y parámetros. Los tres anteriores tópicos se consolidan en la siguiente restricción (Pinedo, 2008),

$$iik_{ik} + \left[\sum_{j \in JOBS} xjk_{j,(k+1)} * TIME_{ij} \right] + wik_{i,(k+1)} = wik_{i,k} + \left[\sum_{j \in JOBS} xjk_{j,k} * TIME_{(i+1),j} \right] + iik_{(i+1),k}$$

$$\forall k \in \{1 \dots NJOBS - 1\}; \forall i \in \{1 \dots M - 1\}$$

RESTRICCIONES

- Se debe garantizar que para cada posición en la secuencia únicamente debe haber un trabajo.

$$\sum_{j \in JOBS} x_{jk} = 1 \quad \forall k \in \{1 \dots NJOBS\}$$

- Se debe garantizar que un trabajo esté una vez en alguna posición de la secuencia.

$$\sum_{k \in \{1 \dots NJOBS\}} x_{jk} = 1 \quad \forall j \in JOBS$$

- Se debe garantizar la relación entre las variables físicas y las funcionales, es decir la relación que existe entre las variables iik_{ik} ; y wik_{ik} , que se evidencia en el siguiente ejemplo, si $iik_{ik} > 0$, es decir la máquina está un tiempo desocupada antes de empezar a procesar el trabajo en la posición $(k+1)$ -ésima en la secuencia, lo anterior implica que ese trabajo, el de la posición $(k+1)$ -ésima inicio su proceso en la máquina i inmediatamente después de que dejó la máquina $i-1$, es decir $w_{(i-1),(k+1)} = 0$,

- En la Figura 1 se muestra una representación gráfica de tal restricción, que formalmente se establece considerando la diferencia entre el tiempo en que el trabajo en la posición $(k+1)$ de la secuencia inicia en la máquina $(i+1)$ y el tiempo en que el trabajo en la posición k de la secuencia termina en la máquina i .
- En la anterior restricción, el tiempo de proceso en la máquina i del trabajo en la posición k de la secuencia pik_{ik} se representa con la siguiente sumatoria:

$$\sum_{j \in JOBS} xjk_{j,(k+1)} * TIME_{ij} = pik_{ik}$$

- El trabajo que se programa de primero una vez termina de hacer la operación en la máquina i pasa de inmediato a la máquina $i+1$, es decir no tiene que esperar por ser el primer trabajo, lo que se controla con la siguiente restricción:

$$wik_{i1} = 0 \quad \forall i \in \{1 \dots M - 1\}$$

- La primera máquina del flowshop una vez termina el trabajo en posición k de la secuencia puede empezar a procesar de inmediato el trabajo de posición $k+1$ en la secuencia, ya que se asume

que los tiempos de liberación son cero, lo cual se garantiza con la siguiente restricción:

$$iik_{1k} = 0 \quad \forall \quad k \in \{1 \dots NJOBS - 1\}$$

- El tiempo de terminación de un trabajo en la posición j de la secuencia en un flowshop, c_j se puede determinar como la suma de los tiempos de proceso en todas las máquinas del trabajo que se programó de primero, más la suma de los tiempos ociosos que tiene la última máquina hasta antes del trabajo en la posición j de la secuencia, más el tiempo de proceso en la última máquina de los trabajos programados antes del trabajo en la posición $j+1$ de la secuencia, sin incluir el primero. Esto se puede ver en la Figura 5.
- En la Figura 5 se puede identificar que el tiempo de terminación del trabajo que se procesa de tercero, es decir, el trabajo T1, es 22, el cual se puede obtener sumando los tiempos de proceso del trabajo que se procesa de primero, trabajo T1, más los tiempos de proceso de los trabajos que están antes del que se procesa de cuarto trabajo en la última máquina, sin incluir el que se procesa de primero (es decir, el trabajo antes de T2), más los tiempos ociosos que tenga la última máquina hasta antes del trabajo que se procesa de tercero.
- Para el ejemplo, solo dos trabajos se procesan antes del trabajo que se procesa de cuarto, que son el trabajo T5 (segundo en la secuencia) con tiempo de proceso igual a 5 unidades y el trabajo T1 (tercero en la secuencia). con tiempo de proceso de 3 unidades. La máquina cuatro solo está ociosa antes del trabajo que se procesa de tercero una vez y el tiempo es de 3 unidades. Los tiempo de proceso

del trabajo que se procesa de primero suman 11 unidades, más las 5 unidades del trabajo T5, más tres unidades del trabajo T1, más las 3 unidades de tiempo ocioso suman 22 unidades que representan el tiempo de terminación del trabajo que se procesa de tercero, o sea el trabajo T1. Por tanto, para el cálculo del tiempo de terminación de un trabajo en la posición s de la secuencia se emplea la siguiente restricción.

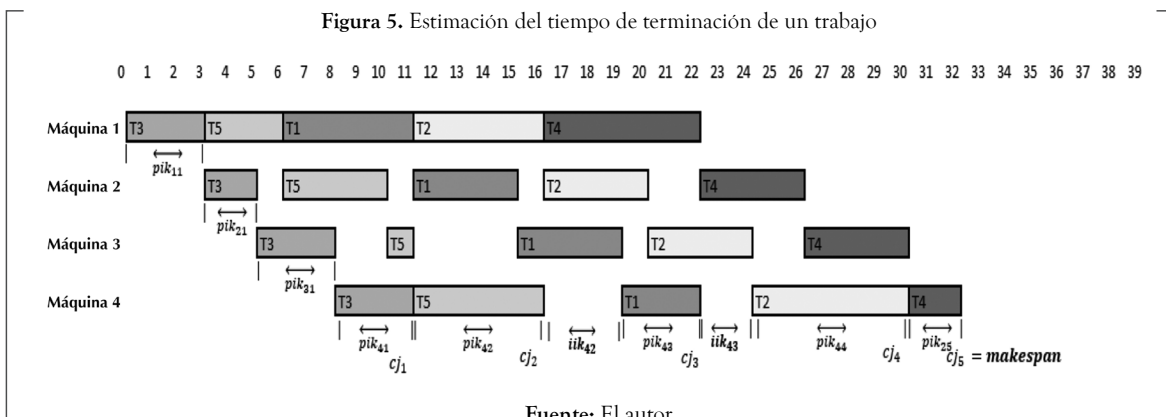
$$c_j^s = \sum_{q \in MAQUINAS} \sum_{j \in JOBS} xjk_{j,1} * TIME_{j,q} + \sum_{k \in \{1, \dots, (s-1)\}} \left[iik_{M,k} + \sum_{j \in JOBS} xjk_{j,(k+1)} TIME_{j,M} \right]$$

$$\forall \quad s \in JOBS$$

- Como elemento de control o medida alternativa de bondad del *Scheduling* se calcula el makespan o el tiempo de terminación del último trabajo en la última máquina, el principio utilizado para el cálculo fue el mismo que se empleó para la determinación del tiempo de terminación de cada trabajo, de hecho el makespan es uno de esos tiempos de terminación, para su determinación propone la siguiente restricción.

$$makespan = \sum_{q \in \{1, \dots, (M-1)\}} \sum_{j \in JOBS} xjk_{j,1} * TIME_{j,q} + \sum_{j \in \{1, \dots, (NJOBS-1)\}} iik_{M,j} + \sum_{j \in JOBS} TIME_{j,M}$$

- Dado que *Scheduling* se realiza en dos fases se requiere saber cuál es el momento en que todas las máquinas procesan su último trabajo, para poder saber de qué capacidad disponen. Para ello se implementa la siguiente restricción, que funciona como la del makespan, pero ahora en esta solo se suman los tiempos de proceso del primer trabajo



Fuente: El autor

hasta antes de la máquina en cuestión, se calculan los tiempos ociosos de tal máquina y la suma de los tiempos de proceso de todos los trabajos en ella

Así :

$$fremaq_q = \sum_{u \in \{1..(q-1)\}} \sum_{j \in JOBS} xjk_{j,u} * TIME_{j,u} + \sum_{j \in \{1..(NBJOBS-1)\}} iik_{q,j} + \sum_{j \in JOBS} TIME_{j,q}$$

$$\forall q \in MAQUINAS$$

FUNCIÓN OBJETIVO EN LA FASE 1

En esta fase se busca minimizar los costos de tenencia de inventario de producto en proceso. Para ello se asume que el costo de tener en inventario en proceso una unidad de tiempo un trabajo es igual para todos los trabajos. Dado el proceso de descomposición temporal, en esta investigación los problemas de *Scheduling* resultantes tienen fechas de liberación iguales a cero, con lo que se puede plantear el costo de tenencia de inventario en proceso, como la sumatoria de los tiempos de terminación de todos los trabajos, multiplicados por el costo de tenencia.

- El costo de tenencia de inventario de producto en proceso se puede plantear de la siguiente manera:

$$\sum_{s \in JOBS} cj_s * CMANWIP1$$

FUNCIÓN OBJETIVO EN LA FASE 2

En la fase 2 se minimiza el costo asociado a la no permanencia de los trabajos en el taller, es decir el costo por unidad de tiempo por estar al final de la línea como producto semiterminado.

- Teniendo en cuenta que la solución del modelo se implementará de derecha a izquierda, el tiempo de no permanencia de todos los trabajos en el taller se aproximará al tiempo de flujo total de la misma secuencia, de izquierda a derecha, menos la suma de tiempos de proceso de los trabajos, como se explicó anteriormente. El costo total de tenencia de trabajos semiterminados al final de la línea será el tiempo de no permanencia de los trabajos en el taller por el costo de tenencia. Esto se representará de la siguiente manera.

$$\left(\sum_{s \in JOBS} (cj_s) - \sum_{m \in \{1..M\}} \sum_{j \in JOBS} TIME_{j,m} \right) * CMANWIP2$$

- La anterior representa el valor de la función objetivo, una vez la secuencia se implemente de derecha a izquierda, pero tal como se explicó en la Figura 3, minimizar el tiempo de permanencia de la secuencia de izquierda a derecha equivale a minimizar el tiempo de no permanencia de la secuencia implementada de derecha a izquierda. Por tanto, la función que optimizará el modelo se presenta a continuación.

$$\sum_{s \in JOBS} (cj_s) * CMANWIP2$$

3. Caso de estudio y análisis de resultados

El desarrollo metodológico planteado se aplica a un problema de programación de trabajos en la industria de alimentos concentrados, en esta industria a partir de un conjunto de varias materias primas se fabrican diferentes productos para la alimentación de animales, resultado de mezcla de las materias primas que deben cumplir con unos requerimientos nutricionales, propios de la dieta de cada animal, según su ciclo de vida y su destinación productiva.

La planta de fabricación compromete diferentes máquinas que son empleadas en el proceso de manufactura y el tiempo de uso de cada una de ellas depende de la selección de materias primas que se haya realizado para la producción de cada uno de los bienes. Para este caso en particular, se manejarán cuatro máquinas: la mezcladora, la dosificadora, el molino y la empacadora.

La mezcladora es la encargada de combinar las materias primas, para que la mezcla sea homogénea, la dosificadora es la encargada de pesar las cantidades de cada materia prima que harán parte de la mezcla, el molino es la máquina encargada de dar el tamaño requerido para el producto, la empacadora es la encargada del vaciado de los productos terminados en costales y el sellado de éstos.

En vista de que se parte de un plan táctico de producción para cada ventana de tiempo, a continuación se presenta la Tabla 1, donde se muestra una sección del plan táctico, en la que se especifican las cantidades a producir, indicándose en las filas los periodos *i* en que se tiene programado producir y en las columnas los periodos *j* de demanda en que serán entregados los productos.

Tabla 1. Sección del plan táctico, unidades en toneladas.

Producto	<i>i \ j</i>	<i>j=1</i>	<i>j=2</i>	<i>j=3</i>
Prepico	<i>i=1</i>	30	30	
	<i>i=2</i>		50	20
	<i>i=3</i>			60
Levante	<i>i=1</i>	25		
	<i>i=2</i>		40	40
	<i>i=3</i>			70
Pre-postura	<i>i=1</i>	20		
	<i>i=2</i>			20
	<i>i=3</i>			
Postura	<i>i=1</i>	40		
	<i>i=2</i>			
	<i>i=3</i>			50
Iniciación	<i>i=1</i>	30		
	<i>i=2</i>		60	
	<i>i=3</i>			60

Fuente: Los autores

A partir de la Tabla 1, y teniendo presente que los tiempos de proceso de los productos depende de la configuración de materias primas, se presenta la Tabla 2 en la que se han seleccionado los trabajos que serán fabricados en la ventana de tiempo 1; para cada uno de ellos se presenta su periodo de fabricación, *i*, su periodo de entrega, *j*, el tamaño del lotes y los tiempos de proceso en cada máquina (vser tabla 2).

Tabla 2. Características de los trabajos de la primera ventana de tiempo.

	Producto	<i>i</i>	<i>J</i>	Lotes (Ton)	M1 (Hr)	M2 (Hr)	M3 (Hr)	M4 (Hr)
T1	Prepico	1	1	30	5	4	4	3
T2	Levante	1	1	25	5	4	4	6
T3	Pre-postura	1	1	20	3	2	3	3
T4	Postura	1	1	40	6	4	4	2
T5	Iniciación	1	1	30	3	4	1	5
T6	Prepico	1	1	25	5	4	4	3

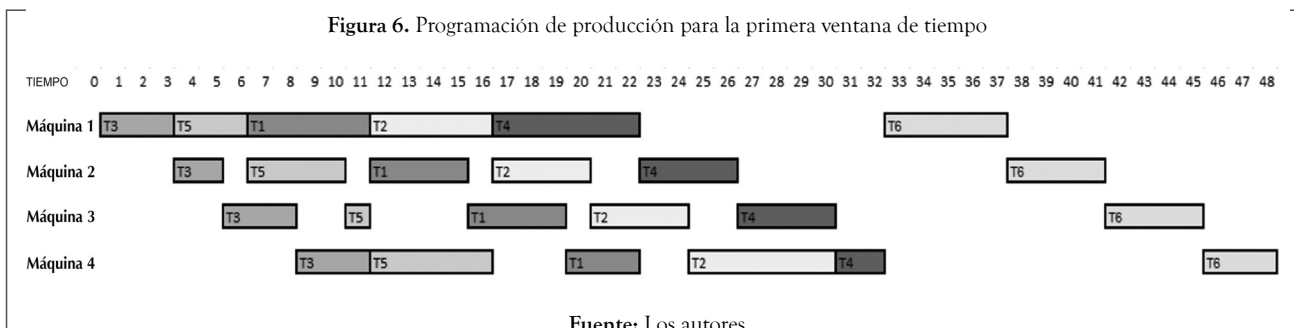
Fuente: Los autores

En Tabla 2, los trabajos T1, T2, T3, T4 y T5 forman el grupo de prioridad uno y deben ser programados en el momento más temprano, y el trabajo T6 es de prioridad dos y debe ser programado en el momento más tardío de la ventana de tiempo.

En la Figura 6, se representa la programación que arroja la metodología para la primera ventana de tiempo, los datos tomados corresponden a los de la Tabla 2.

Los tiempos de terminación de los trabajos T3, T5, T1, T2 y T4, son respectivamente 11, 16, 22, 30 y 32. Partiendo de un valor del costo de tener en inventario en proceso en el sistema igual a 100 unidades monetarias por hora, se obtiene un valor de la función objetivo para la fase 1 de 11100 unidades monetarias. Es importante aclarar que al ser costo de tenencia constante para todos los productos considerarlo o no, la solución resultante sería la misma y solo tendría efecto en el valor de función objetivo, por lo que vale la pena presentar una medida de desempeño con gran interpretación práctica que es el tiempo de flujo promedio, que para este caso será de 22.2 horas, que significa que en promedio cada trabajo permanece en el taller 22.2 horas desde el momento en que entra al sistema hasta cuando sale de la planta, ya que son de entrega inmediata.

Figura 6. Programación de producción para la primera ventana de tiempo



Fuente: Los autores

El trabajo 6, tal como lo especifica el plan táctico, será fabricado el periodo 1 y entregado en el periodo dos, lo cual puede ser explicable por restricciones o por el esquema de costos de ese plan. Este trabajo es único de la fase dos y si se programa en una secuencia de adelanta hacia atrás iniciando en el periodo cero su tiempo de terminación o de permanencia en el sistema es 16 y su complemento el tiempo de no permanencia, es de 32, equivalente a restar el tiempo de permanencia ($TPSAD=16$) a la duración de la ventana de tiempo (48). Su tiempo de no permanencia o complemento, de tiempo de permanencia en la secuencia de derecha a izquierda es de cero ($TNPSAT=0$), tal como se explicó en la Figura 3, $TNPSAT=TPSAD-TP$.

También en la fase dos por asumirse que el costo de tenencia de inventario de producto en proceso en el sistema (COMANWIP2) es constante para todos los productos, la solución será la misma o no se verá afectada ante variaciones en este parámetro, por tanto el valor de cero en la secuencia de adelante para atrás refleja que el trabajo se programa en el momento más tarde posible de la ventana de tiempo.

4. Conclusiones

La propuesta planteada para tratar los requerimientos no tradicionales se apoya en dos modelos de optimización y se parte del supuesto de que los trabajos en un mismo período, tienen tiempos de proceso similares se logra disminuir la complejidad del problema original, encontrándose soluciones factibles que son óptimos locales.

Si bien es cierto que con el enfoque propuesto no se determina la solución óptima del problema original, se realiza un buen acercamiento en términos de la formalización del mismo, según la investigación efectuada no hay evidencias en términos de resolver de manera integrada el problema descrito en esta investigación, pero se tiene claro que solo el problema de *Scheduling* resultante es strongly NP-hard. Lo cual nos indica que cualquier acercamiento, en términos de modelar y resolver el problema con sus características particulares, empleando herramientas de investigación de operaciones, representa una ganancia.

Desde el punto de vista práctico, es importante resaltar que para poder abordar los flujos de información inherentes al enfoque empleado se requieren

herramientas computacionales que faciliten este proceso y lo hagan eficiente, En este caso, se empleó OPL como lenguaje de modelación, Ceplex como solver y OPI Script como herramienta de programación estructurada para garantizar los flujos.

La propuesta considera que los costos de tenencia de cada una de las fases son constantes para todos los productos, una posible extensión del trabajo sería considerar estos costos diferentes para cada producto, aspecto que no fue estimado en esta investigación, ya que genera una no linealidad en la función objetivo.

La formulación del problema de diferentes prioridades en un solo modelo matemático genera una no linealidad al momento de definir la función de desempeño, se recomienda como investigación futura emplear herramientas tipo Constraint satisfaction problems (CSPs) como alternativa para tratar dicha no linealidad.



CONFLICTO DE INTERESES

Los autores declaran no tener ningún conflicto de intereses.

NOTAS

1. Artículo resultado del proyecto de investigación: Planeación y control de la producción en la Industria de concentrados. Desarrollado entre enero de 2011 y diciembre de 2013.

REFERENCIAS BIBLIOGRÁFICAS

1. AGHEZZAF, E-H, SITOMPUL, C. & VAN DEN BROECKE, F.V. A robust hierarchical production planning for a capacitated two-stage production system, *En: Computers & Industrial Engineering*. . Marzo, 2011. Vol 60, Iss. 2, P 361-372.
2. ALZATE SALAZAR, Juan D. Recopilación y sistematización de los temas de complejidad computacional en los modelos de *scheduling*, Cali: Trabajo de grado, Escuela de ingeniería industrial - Universidad del Valle, 2010. P 106.
3. BITRAN, Gabriel R.; HAAS, Elizabeth A. & HAX, Arnoldo C. Hierarchical production Planning: A Two Stage System, *En: Operations Research*. Marzo – abril, 1982. Vol. 30, no. 2, P. 232-251.
4. CHEN, Kejia & JI, Ping. A mixed integer programming Model for advanced planning and scheduling (APS) *En: European Journal*

- of Operational Research. Agosto, 2007. Vol. 181, no. 1. P. 515-522.
5. CHENG, Mingbao, TADIKAMALLA, Pandu R., SHANG, Jennifer & ZHANG, Shaqing. Bicriteria hierarchical optimization of two-machine flow shop scheduling problem with time-dependent deteriorating jobs. *En: European Journal of operational research*. Marzo, 2013. Vol 234, no. 3. P. 650 – 657.
 6. CONWAY, Richard W., MAXWELL, William L. & MILLER, Louis W. *Theory of scheduling*. Massachusetts: Addison-Wesley Pub. Co., 1967. P. 18.
 7. GRAVES, Stephen C. *Manufacturing planning and control*, New York: Oxford University Press, 2002. P. 728 – 746. ISBN 9780195125948.
 8. HARHALAKIS, George; NAGI, R. & PROTH, J. R. Hierarchical modeling approach for production planning. *En: Control Engineering Practice*. Febrero, 1993. Vol 1, no. 1, P. 31 – 36.
 9. HE, N.; ZHANG, D.Z. & LI, Q. Agent-based hierarchical production planning and scheduling in make-to-order manufacturing system. *En: International Journal of Production Economics*. Marzo, 2014. Vol. 149. P.117-130..
 10. LI, Zukui & IERAPETRITOU, Marianthi G. Production planning and scheduling integration through augmented lagrangian optimization, *En: Computers and chemicals engineering*. Junio, 2010. Vol 34, no. 6. P. 996-1006.
 11. LINDHOLM, Ann; LIDESTAM, Helene & QUTTINEH, Nils-Hassan. Hierarchical production Scheduling- A case study at Perstorp. *En: Computer Aided Chemical Engineering*. Junio, 2014. Vol 33. P. 511-516
 12. MÁRQUEZ DELGADO, José Ed. Optimización de la programación (scheduling) en talleres de mecanizado. Madrid: Tesis Doctoral, Universidad politécnica de Madrid, 2012, p 187.
 13. MICHAEL Pinedo. *Scheduling, Theory, Algorithms and Systems*. 3. Ed. New York: Springer, 2008. P. 665. ISBN 978-0-387-78934-7.
 14. MONTENEGRO VILLOTA, Angela L. Recopilación, sistematización y análisis crítico de los desarrollos en scheduling ocurridos en la década de 1993 al 2003. Cali: Trabajo de grado, Escuela de ingeniería industrial - Universidad del Valle, 2005. P 289.
 15. MORTON, Thomas E. & PENTICO, David W. *Heuristic Scheduling Systems: With Applications to Production Systems and Project Management*, New York: A Wiley & Sons, Inc, 1993. P. 351. (Wiley Series in Engineering and Technology Management Colección). ISBN 978-0471578192.
 16. MOTOA GARAVITO, Tulio G.; OSORIO GOMEZ, Juan C. & OREJUELA CABRERA, Juan P. Planificación Jerárquica de la Producción (Hierarchical Production Planning): El Estado del Arte y Presentación de Experiencias. *En: Heurística: Universidad del Valle*. Octubre, 2007. Vol. 14, no. 1. P.35 -47.
 17. NOORI, Hamid. & RADFORD, Russel. Administración de operaciones y producción: Calidad total y respuesta sensible rápida. Traducido por Germán Alberto Villamizar, Bogotá: McGraw-Hill, 1997. P. 600. ISBN 9789586006033.
 18. OSORIO GOMEZ, Juan C.; CASTRILLON MONTENEGRO, Oscar E.; TORO CARDONA, Juliana A. & OREJUELA CABRERA, Juan P. Modelo de programación Jerárquica de la producción en un Job Flexible con interrupciones y tiempos de alistamiento dependientes de la secuencia. *En: Revista Ingeniería e Investigación: Universidad Nacional de Colombia*. Agosto, 2008. Vol. 28, no. 2, P. 72-79.
 19. REZA, Tanha A.& Ghaderi, S.F. Generation Planning in Iranian power plants with fuzzy hierarchical production planning. *Junio, 2010. Vol 51, Iss. 6, P 1230-1241.*
 20. SAAD, Germaine H. Hierarchical production - planning systems: extensions and modifications, *En: The journal of operational research society*. Julio, 1990. Vol. 41, no.7, P. 609-624.
 21. SCHWARTZ, Jay D. & RIVERA, Daniel E. A process control approach to tactical inventory management in production-inventory systems. *En: Int. J. Productions Economics*. Mayo, 2010. Vol. 125, no. 1. P. 111- 124.
 22. SINNEN, Oliver. Reducing the solution space of optimal task scheduling, *En: Computer & operations research*, Marzo, 2014. Vol 43, no.1. P. 201-214.
 23. SIPPER, Daniel. y BULFIN, Robert L. Planeación y Control de la Producción. Traducido por Marcia González Osuna. Ciudad de México: McGraw-Hill, 1998, P. 351. ISBN 9789701019443
 24. SMITH, Stephen F. *Multidisciplinary Scheduling: Theory and Applications*. 1st International Conference, MISTA '03 Nottingham, UK, 13–15. Nottingham: Springer Science+Business Media, Inc., 2003. P. 3 – 17.
 25. SZOKE, Akos. Conceptual scheduling model and optimized release scheduling for agile environments, *En: Information and software technology*. Junio, 2011. Vol 52, no. 6. P. 574-591.
 26. TANG, Yuanjie.; LIU, Rengkui & SUN, Quanxin. Scheduling control model for linear projects based on linear scheduling method and constraint programming. *En: Automation in construction*. Enero, 2014, no. 1. P. 22-37
 27. TIAN, Feng; WILLEMS, Sean P. & KEMPF, Karl G. An iterative approach to item-level tactical production and inventory planning. *En: International Journal of Production Economics*. September, 2011. Vol. 133, Iss. 1. P.439-450.
 28. TISCHER, Irene. & CARRION, Andrés. La planificación jerárquica y su aplicación a la cosecha de caña de azúcar. *En: Ingeniería y competitividad: Universidad del Valle*. Abril 2003. Vol. 4, no. 2 P.42 – 52
 29. TORABI, S.A; EBADIAN, M. & TANHA, R. Fuzzy hierarchical production planning (with a case study). *En: Fuzzy Sets and Systems*. Junio, 2010. Vol. 161, Iss. 11. P. 1511-1529
 30. VOLLMANN, Thomas E.; BERRY William L. & WHYBARK D. Clay. *Sistemas de Planificación y Control de la Fabricación*. 3. Ed. Madrid: McGraw-Hill, 1995. P. 450. ISBN 84-8086-195-9.
 31. WAN, Long y ZHANG, An. Coordinated scheduling on parallel machines with batch delivery. *En: Int. J. Production Economics*, Abril, 2014. Vol. 150, no.1. P 199-203.
 32. ZHONG, Xueling, OU, Jinwen. & WANG, Guoqing. Order acceptance and scheduling with machine availability constraints, *En: European journal of operational research*. Febrero, 2014. Vol. 232, no. 3. P. 435-441.