

## Consolidación de órdenes en la solución simultánea del batching y el scheduling en plantas batch multiproducto

Sergio Ackermann<sup>1</sup>, Yanina Fumero<sup>1</sup>, Jorge Marcelo Montagna<sup>1</sup>

<sup>1</sup> INGAR, Instituto de desarrollo y diseño (CONICET-UTN),  
Avellaneda 3657 (3000) Santa Fe, Argentina  
{sackermann, yfumero, montana}@santafe-conicet.gov.ar

**Resumen.** La resolución simultánea de los problemas de batching y scheduling en plantas batch multiproducto y multietapa con múltiples unidades de diferentes tamaños operando en paralelo en cada etapa continúa presentando desafíos, tanto en lograr modelos computacionalmente eficientes como en considerar todos los posibles escenarios de producción. En este tipo de plantas, la gran mayoría de los trabajos publicados que consideran múltiples órdenes por producto, tratan a las mismas de forma individual. Este procedimiento genera soluciones que pueden mejorarse si todas las órdenes de un mismo producto son consolidadas en una demanda total del mismo. Al consolidar las órdenes, en general, se requiere procesar menos batches y se utiliza la capacidad de la planta de forma más eficiente. Este trabajo analiza las ventajas y consecuencias de adoptar la consolidación de órdenes cuando el esquema de producción y las características del producto lo permitan. Se presenta una formulación de tiempos discretos mixta-entera lineal (MILP) para la resolución simultánea de los problemas de batching y scheduling en una planta batch multiproducto multietapa donde se admite la consolidación de órdenes de un mismo producto. La planta cuenta con unidades diferentes operando en paralelo en cada etapa y no se dispone de tanques de almacenamiento entre etapas, por lo que se adopta la política de transferencia de espera cero (ZW). Se presenta un ejemplo para evaluar la capacidad del modelo para encontrar una solución óptima como así también su performance computacional.

### 1 Introducción

La operación eficiente de un sistema de producción es clave para el éxito de una industria. En particular, las plantas batch tienen una importancia significativa en la industria de procesos, debido a su flexibilidad para producir diferentes productos y cantidades, muchas veces siguiendo recetas específicas del cliente. Por lo tanto, debido a esa constante variación resulta fundamental el scheduling de la producción para planificar la operación en el corto plazo y lograr una utilización eficiente de los recursos. En las últimas décadas se han presentado numerosos trabajos abordando este tema, acerca de los cuales pueden encontrarse exhaustivas revisiones ([1], [2], [3], [4]).

En el contexto de la producción secuencial, consistente en varias etapas con múltiples unidades de diferente tamaño en paralelo en cada etapa, donde cada batch debe

seguir la misma secuencia a través de todas las etapas, el problema de planeamiento de la producción a corto plazo históricamente ha sido resuelto en un procedimiento jerárquico consistente en dos etapas: primero, se resolvía el problema de batching consistente en determinar el número y tamaño de batches a procesar, y entonces, esos valores eran utilizados como datos para resolver el scheduling, en la segunda etapa, consistente en la asignación de los batches a las unidades donde serán procesados y su secuenciamiento, determinando sus tiempos iniciales y finales de procesamiento. Este procedimiento conduce a soluciones que no son óptimas, pero permite lidiar con problemas reales en un tiempo computacional razonable. En tiempos más recientes, y con la disponibilidad de computadoras más potentes, dentro del ámbito de la programación matemática se han desarrollado formulaciones que, con diferentes características y limitaciones, proponen la optimización simultánea de los problemas de batching y scheduling. Inicialmente todos ellos utilizaron representaciones continuas del tiempo ([5], [6], [7], [8], [9], [10], [11]), sin embargo, las representaciones de tiempo discretas han sido ampliamente utilizadas en modelados propuestos en los últimos tiempos ([12], [13],[14]).

Recientemente, Ackermann y col. [15] han desarrollado una nueva representación del problema, introduciendo el concepto de camino de producción. En esta propuesta, los autores han formulado un modelo MILP de tiempos discretos para la resolución simultánea del batching y el scheduling en el contexto de plantas batch multiproducto multietapa con unidades diferentes en paralelo, operando con política de transferencia de espera cero (ZW).

En general, la mayoría de los trabajos anteriores consideran órdenes mono-producto con una fecha de vencimiento conocida. Si un cliente requiere varios productos en una misma fecha, este pedido se descompone en varias órdenes con la misma fecha de vencimiento. Bajo esta condición, los trabajos mencionados realizan el batching, consistente en determinar el número y tamaño de los batches, para cada orden individualmente. Usualmente, en este tipo de plantas, los batches no se mezclan o dividen, incluso en el caso en que se dispone de almacenamiento intermedio. Por lo tanto, la integridad de los batches se preserva a través de la secuencia de procesamiento completa. El primer trabajo que menciona la importancia de tratar de manera especial el caso en que se cuenta con múltiples órdenes de un mismo producto con diferentes fechas de vencimiento es presentado por Lim y Karimi [5] para una planta de una única etapa con unidades diferentes en paralelo. Ellos proponen que un batch se puede utilizar para satisfacer múltiples órdenes de un mismo producto y consideran la demanda total de cada producto como la suma de las órdenes de ese producto. Esta suposición cambia la estructura del problema ya que ahora no es necesario determinar los batches a partir de órdenes monoproducto, sino teniendo en cuenta la demanda total de cada producto. Entonces, esta demanda debe ser dividida en batches utilizando algún procedimiento de batching. La formulación presentada por los autores asegura que cada orden sea completada, pero no garantiza cumplir con las fechas de vencimiento. La demora en abastecer una orden es llamada tardanza y la función objetivo consiste en minimizar la suma ponderada de la tardanza de todas las órdenes.

Siguiendo la misma idea, donde todas las órdenes de un mismo producto son consolidadas en una única demanda de ese producto, Marchetti y col. ([9], [10]) desarrollaron modelos para plantas de una única y múltiples etapas respectivamente. Para

resolver el problema, ellos dirigieron su trabajo hacia la generación de clusters de batches, en lugar de tratarlos de forma individual. En este sentido, se cambia el foco del problema, puesto que antes era necesario realizar el batching para un gran número de órdenes monoprodueto mientras que ahora es necesario hacerlo para la demanda total de cada producto. Por lo tanto, debe generarse un número menor de soluciones de batching, pero para cantidades mayores de producto.

Cuando se considera la consolidación de órdenes, se producen varios cambios que deben ser tenidos en cuenta. Antes, cada batch era tratado de forma individual, estaba asociado a una única orden y por lo tanto tenía una fecha de vencimiento y sólo se encontraba relacionado con los otros batches de la misma orden. Ahora, con la nueva formulación, todos los batches de un mismo producto se encuentran relacionados y los mismos no tienen fecha de vencimiento ya que no están asociados a una orden particular, por lo que es necesario garantizar de alguna manera que todos los requerimientos de los clientes sean satisfechos en las fechas de vencimiento correspondientes.

La consolidación de órdenes, que no ha sido estudiada en profundidad en trabajos previos, tiene en principio dos grandes ventajas: en primer lugar, permite un mayor uso de la capacidad de la planta, puesto que se requiere un número menor de batches para satisfacer toda la demanda; en segundo lugar, permite tratar el caso de órdenes muy pequeñas que no cumplen con la ocupación mínima requerida en los equipos, al consolidarlas en una demanda mayor. Esta segunda cuestión no ha sido tratada en detalle en los trabajos de la literatura revisada para esta presentación, ya que o bien consideran que no es factible procesar órdenes cuyo tamaño se encuentre por debajo de los mínimos admisibles en las unidades de la planta o permiten procesar excedentes de producto final respecto a la demanda.

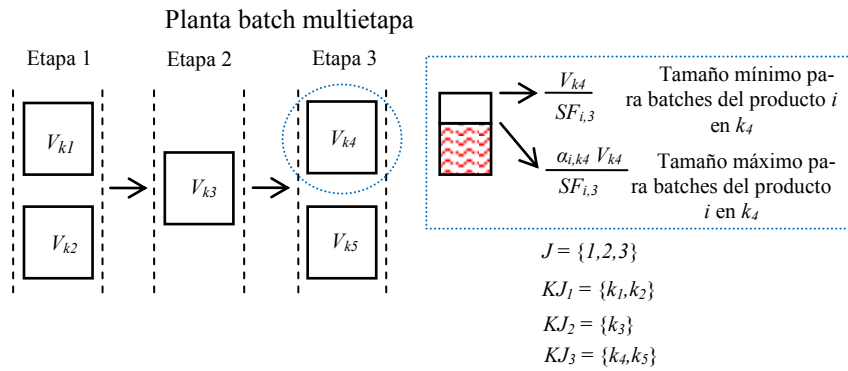
Este trabajo presenta una formulación de programación mixta-entera lineal (MILP) basada en tiempos discretos para la resolución simultánea del batching y el scheduling en plantas batch multiproducto multietapa que deben satisfacer una demanda consistente de múltiples órdenes de cada producto. Cada orden es caracterizada por su tamaño y fecha de vencimiento. En este modelo, las órdenes de un mismo producto se consolidan en una demanda unificada de ese producto, que es igual a la suma del requerimiento de todas esas órdenes. Cada orden debe satisfacerse antes de su fecha de vencimiento, lo cual es uno de los principales desafíos, desde el punto de vista del modelado, cuando se consolidan órdenes. La planta multietapa tiene varias unidades de diferentes tamaños operando en paralelo en cada etapa y no se permite almacenamiento intermedio. Para modelar las decisiones de batching y scheduling, se adopta la representación basada en caminos introducida por Ackermann y col. [15]. El modelo debe determinar el número y tamaño de los batches procesados de cada producto, el camino de producción al que se asigna cada batch y su tiempo inicial de procesamiento. El objetivo del problema es minimizar el makespan, esto es, minimizar el tiempo total requerido para procesar todas las órdenes.

## 2 Formulación del problema

Una planta batch multietapa multiproducto capaz de procesar un conjunto de productos  $i \in I$  debe satisfacer la demanda de un conjunto de órdenes monoprodueto  $d \in$

*D*. Cada orden contiene un único producto  $i \in I$ , y el conjunto  $D_i$  corresponde a todas las órdenes del producto  $i$  ( $D_i \subseteq D$ ). El tamaño  $Q_d$  de la orden  $d$ , como así también su fecha de vencimiento  $DD_d$ , son datos conocidos. Para simplificar la notación, las órdenes de un mismo producto son listadas en orden creciente de acuerdo a su fecha de vencimiento. También se consideran fechas de emisión  $RD_i$  para cada producto, antes de la cual no se puede comenzar a procesar ningún batch de ese producto.

La estructura de la planta es completamente conocida. En la Figura 1 se muestra un esquema de la misma. Cada etapa  $j \in J$  está compuesta por unidades de diferentes tamaños operando en paralelo y fuera de fase, denotadas por  $k \in K_j$ . La unidad  $k$ , de tamaño  $V_k$ , tiene un porcentaje mínimo de ocupación  $\alpha_{ik}$  para procesar un batch de product  $i$  y el tiempo requerido para procesar cada batch de ese producto es  $t_{ik}$ . Finalmente, se considera un factor de tamaño  $SF_{ij}$ , que indica el tamaño requerido en la etapa  $j$  para producir una unidad de masa del producto  $i$ .



**Fig. 1.** Estructura de una planta batch multietapa.

### 2.1 Consolidación de órdenes y sus ventajas

En este trabajo, en lugar de tratar cada orden de forma individual, todas las órdenes de producto  $i$  se consolidan en una única demanda de ese producto, llamada  $QT_i$ , que está dada por:

$$QT_i = \sum_{d \in D_i} Q_d \quad \forall i \in I \tag{1}$$

La planta procesa batches de producto  $i$  en lugar de batches de una orden en particular. Una vez que el procesamiento de un batch concluye, éste se puede utilizar para satisfacer una orden o ser dividido para cumplir con varias órdenes.

Para mostrar las ventajas de la consolidación de órdenes, se analiza un ejemplo sencillo. Se consideran una planta con una única unidad que procesa batches de entre 50 y 100 kilos de cierto producto y dos escenarios de demanda de este producto:

- a) Deben procesarse dos órdenes, ambas de 150 kilos. Si se tratan ambas órdenes de forma individual, se deben procesar dos batches para cada una de ellas, por lo tanto se requieren 4 batches y la ocupación promedio de la capacidad de la unidad es del 75%. Por el contrario, consolidando las dos órdenes, la demanda total es de 300 kilos de producto, que puede ser cumplida usando 3 batches y ocupando el 100% de la capacidad de la unidad;
- b) Deben procesarse dos órdenes de 40 kilos cada una. En este caso no es factible procesar cada orden de forma individual puesto que la capacidad mínima de la unidad es de 50 kilos. Sin embargo, consolidando las dos órdenes es posible procesar un único batch de 80 kilos y usarlo para satisfacer ambas órdenes.

En conclusión, consolidando órdenes de un mismo producto, el número de batches que necesitan ser procesados puede reducirse y, en consecuencia, el tiempo requerido para completar todas las órdenes puede disminuir. Además, con este enfoque, se mejora la utilización de la capacidad productiva de la planta y puede solucionarse el problema del procesamiento de órdenes que no son factibles debido a su tamaño.

## 2.2 Cotas para el número de batches

Cada batch se mantiene íntegro mientras se procesa, sin dividirse o mezclarse con otros batches. Los batches de todos los productos siguen la misma secuencia a través de todas las etapas de la planta. No se permite almacenamiento intermedio: cuando un batch finaliza su procesamiento en una etapa es inmediatamente transferido a la etapa siguiente, esto es, se adopta la política de transferencia de espera cero (ZW).

Para resolver el problema de batching, se adopta la representación basada en caminos de producción (o simplemente “caminos”) introducida por Ackermann y col. [15]. Un camino  $r$  es una  $|J|$ -upla, es decir, una lista ordenada de  $|J|$  elementos  $(k_1, k_2, \dots, k_{|J|})$  donde  $k_1 \in KJ_1, k_2 \in KJ_2, \dots, k_{|J|} \in KJ_{|J|}$ , esto es una secuencia compuesta por una unidad de cada etapa (Figura 2).

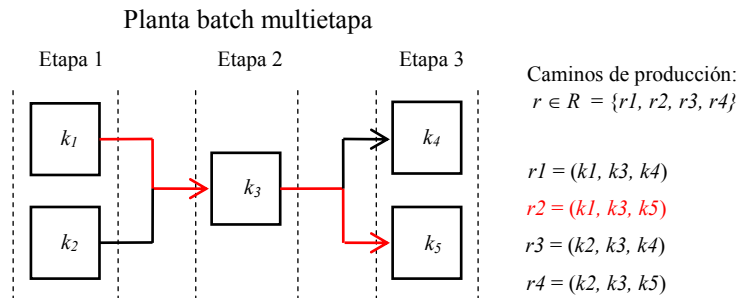


Fig. 2. Definición de los caminos de producción

Sea  $R$  el conjunto de todos los posibles caminos en la planta y  $KR_r$  el conjunto de unidades que forman el camino  $r \in R$ . Entonces, el mínimo y máximo tamaño posible para un batch del producto  $i$  en el camino  $r$  son, respectivamente:

$$B_{ir}^{min} = \max_{j \in J} \{B_{irj}^{min}\} \quad \forall i \in I, r \in R \quad (2)$$

$$B_{ir}^{max} = \min_{j \in J} \{B_{irj}^{max}\} \quad \forall i \in I, r \in R \quad (3)$$

donde el mínimo y máximo tamaño admisible para un batch en cada etapa del camino están definidos por:

$$B_{irj}^{min} = \alpha_{ik} \frac{V_k}{SF_{ij}} \quad \forall i \in I, r \in R, j \in J, k \in KR_r, k \in KJ_j \quad (4)$$

$$B_{irj}^{max} = \frac{V_k}{SF_{ij}} \quad \forall i \in I, r \in R, j \in J, k \in KR_r, k \in KJ_j \quad (5)$$

Notar que los parámetros anteriores están bien definidos pues existe una única unidad  $k$  que pertenece a la etapa  $j$  y al camino  $r$ .

El conjunto  $R$  contiene todos los caminos posibles, sin tener en cuenta el tamaño de las unidades, sus porcentajes mínimos de ocupación o factores de tamaño. Por lo tanto, este conjunto puede contener caminos que no son factibles de utilizar para procesar un producto. Por ejemplo, si  $B_{ir}^{min} > B_{ir}^{max}$ , no es posible procesar ningún batch de producto  $i$  en el camino  $r$ . Entonces, para cada producto  $i$ , se define el conjunto  $F_i$ , que contiene todos los caminos en los que un batch de ese producto puede ser procesado, esto es, los caminos que satisfacen la desigualdad  $B_{ir}^{min} \leq B_{ir}^{max}$ .

El mínimo y máximo tamaño factible para cada batch de producto  $i$  en la planta están dados por:

$$B_i^{min} = \min_{r \in F_i} \{B_{ir}^{min}\} \quad \forall i \in I \quad (6)$$

$$B_i^{max} = \max_{r \in F_i} \{B_{ir}^{max}\} \quad \forall i \in I \quad (7)$$

Una vez conocidos los parámetros anteriores, el posible número de batches a procesar del producto  $i$  está acotado por los siguientes valores:

$$NB_i^{min} = \left\lceil \frac{QT_i}{B_i^{max}} \right\rceil \quad \forall i \in I \quad (8)$$

$$NB_i^{max} = \left\lfloor \frac{QT_i}{B_i^{min}} \right\rfloor \quad \forall i \in I \quad (9)$$

Esto es, el número  $n$  de batches de producto  $i$  a ser procesados se debe seleccionar del conjunto  $N_i$  de enteros positivos que satisfacen  $NB_i^{min} \leq n \leq NB_i^{max}$  para todo  $n \in N_i$ . Esta selección se realiza a través de la variable binaria  $x_{in}$  definida por:

$$x_{in} = \begin{cases} 1 & \text{si el producto } i \text{ se divide en } n \text{ batches} \\ 0 & \text{en otro caso} \end{cases}$$

que debe satisfacer la ecuación:

$$\sum_{n \in N_i} x_{in} = 1 \quad \forall i \in I \quad (10)$$

### 2.3 Asignación de los batches a los caminos

Sea  $IB_{in} = \{b1, b2, \dots, bn\}$  el conjunto que contiene los batches correspondientes al producto  $i$  si se requieren  $n \in N_i$  batches de ese producto para satisfacer su demanda total. Cada batch de  $IB_{in}$  debe ser asignado a un y solo un camino para su procesamiento. La asignación se hace a través de la siguiente variable binaria:

$$z_{inbr} = \begin{cases} 1 & \text{si el producto } i \text{ se divide en } n \text{ batches y el batch } b \text{ se asigna al camino } r \\ 0 & \text{en otro caso} \end{cases}$$

No todas las asignaciones batch-camino pueden ser factibles teniendo en cuenta la demanda del producto  $i$ , el número de batches en que se divide esa demanda y los tamaños admisibles para los batches en cada camino. Sólo es posible procesar el producto  $i$  dividido en  $n$  batches  $b1, b2, \dots, bn \in IB_{in}$  asignados respectivamente a los caminos  $r_{m1}, r_{m2}, \dots, r_{mn} \in F_i$ , si:

$$\sum_{h=1}^n B_{ir_{mh}}^{min} \leq QT_i \leq \sum_{h=1}^n B_{ir_{mh}}^{max} \quad (11)$$

donde un cierto camino se puede utilizar varias veces. Esta expresión asegura que una asignación es factible si y solo si la suma de los tamaños mínimos admisibles para los batches en todos los caminos seleccionados es menor o igual que la demanda total de producto  $i$  y, por otro lado, la suma de los tamaños máximos admisibles para los batches en los mismos caminos es mayor o igual que esa demanda. Si la desigualdad es satisfecha, todas las posibles combinaciones de asignación de los batches a los caminos son factibles y generan soluciones equivalentes. Por lo tanto, la asignación se realiza en orden creciente. De todas las asignaciones batch-camino que satisfacen las condiciones anteriores, se generan los conjuntos  $PR_{inb}$  que contienen los caminos  $r$  donde es factible procesar el batch  $b \in IB_{in}$  si el producto  $i$  es dividido en  $n$  batches.

La introducción de estos conjuntos, determinados a priori a partir de los datos disponibles, reduce ampliamente la naturaleza combinatoria del problema de asignación batch-camino, especialmente cuando los tamaños admisibles en los caminos de la planta difieren unos de otros.

Cada batch de un producto es asignado a un único camino para su procesamiento:

$$\sum_{r \in PR_{inb}} z_{inbr} = x_{in} \quad \forall i \in I, n \in N_i, b \in IB_{in} \quad (12)$$

Con el objetivo de evitar soluciones alternativas, la asignación batch-caminos se realiza en orden creciente, esto es, si los batches  $b, b' \in IB_{in}$  con  $b < b'$  son asignados a los caminos  $r, r' \in F_i$  donde  $r < r'$ , entonces  $b$  es asignado a  $r$  y  $b'$  a  $r'$ .

$$\sum_{\substack{r' \in PR_{inb+1} \\ r' < r}} z_{inb+1r'} \leq 1 - z_{inbr} \quad \forall i \in I, n \in N_i, n \geq 2, b, b+1 \in IB_{in}, r \in PR_{inb} \quad (13)$$

#### 2.4 Restricciones temporales y de secuenciamiento

En este modelo se adopta una representación de tiempos discretos, proponiendo una grilla uniforme, donde se usa el índice  $t \in T$  para denotar puntos de tiempo desde 0 (que corresponde a la menor de las fechas de emisión) hasta la última fecha de vencimiento. Como usualmente se hace en este tipo de representaciones, los intervalos de tiempo  $\Delta t$  entre puntos de tiempo consecutivos en la grilla es igual al máximo común divisor entre todos los tiempos de procesamiento.

Teniendo en cuenta el enfoque propuesto, basado en caminos, el modelo determina explícitamente solo el tiempo de inicio de cada batch en el camino donde se procesa. A partir de ese tiempo de inicio y considerando que se adopta la política de transferencia ZW, el tiempo final de procesamiento se puede determinar sumando los tiempos de procesamiento en cada unidad que compone el camino. El tiempo de procesamiento total  $t_{ir}$  de un batch de producto  $i$  en el camino  $r$  está dado por:

$$t_{ir} = \sum_{k \in KR_r} t_{ik} \quad \forall i \in I, r \in F_i \quad (14)$$

Sea  $DD_i = \max_{d \in D_i} \{DD_d\}$  la mayor fecha de vencimiento entre todas las órdenes de producto  $i$ . Entonces, los posibles tiempos de inicio para un batch de ese producto, denotados por  $t \in TIR_{ir}$ , son aquellos que satisfacen  $RD_i \leq t \leq DD_i - t_{ir}$ . Si no existen valores de  $t$  que satisfagan la desigualdad, no es factible procesar el producto.

Se define la variable binaria  $y_{inbrt}$  de la siguiente manera:

$$y_{inbrt} = \begin{cases} 1 & \text{si el producto } i \text{ se divide en } n \text{ batches y el batch } b \text{ comienza a} \\ & \text{procesarse en el camino } r \text{ en el tiempo } t \\ 0 & \text{en otro caso} \end{cases}$$

y la siguiente ecuación asegura que cada batch procesado es asignado a un único tiempo de inicio

$$\sum_{t \in TIR_{ir}} y_{inbrt} = z_{inbr} \quad \forall i \in I, n \in N_i, b \in IB_{in}, r \in PR_{inb} \quad (15)$$



La siguiente ecuación hace que la variable  $y_{inbrt} = 0$ , para todos los  $b, r$  y  $t$  correspondientes, cuando  $x_{in} = 0$ . Esta ecuación es redundante con las ecuaciones (12) y (15), pero su inclusión mejora la performance computacional del modelo.

$$\sum_{b \in IB_{in}} \sum_{r \in PR_{inb}} \sum_{t \in TR_{ir}} y_{inbrt} = x_{in} \quad \forall i \in I, n \in N_i \quad (16)$$

Resta introducir condiciones para evitar solapamientos entre los tiempos de procesamiento de los batches en cada unidad de la planta. Esto se logra imponiendo que, en cada unidad, el punto de tiempo  $t$  sólo es utilizado para procesar como máximo un batch.

Sea  $i$  un producto y  $r \in F_i, r = (k_1, k_2, \dots, k_j, \dots, k_{|J|})$  con  $k_1 \in KJ_1, k_2 \in KJ_2, \dots, k_j \in KJ_j, \dots, k_{|J|} \in KJ_{|J|}$ , un camino factible para procesar un batch de producto  $i$ . Para cada unidad  $k \in KR_r$ , y  $k \in KJ_j$ , esto es, la unidad  $k$  pertenece a la etapa  $j$  del camino  $r$ , se definen los siguientes dos parámetros:

$$TL_{irk} = \sum_{j'=1}^j \sum_{\substack{k' \in KJ_{j'} \\ k' \in KR_r}} t_{ik'} \quad \forall i \in I, j \in J, k \in KJ_j, r \in F_i, r \in RK_k \quad (17)$$

$$TU_{irk} = \sum_{j'=1}^{j-1} \sum_{\substack{k' \in KJ_{j'} \\ k' \in KR_r}} t_{ik'} \quad \forall i \in I, j \in J, j \geq 2, k \in KJ_j, r \in F_i, r \in RK_k \quad (18)$$

donde  $RK_k$  es el conjunto de caminos que contienen a la unidad  $k$ . La ecuación (17) determina el tiempo total de procesamiento de cualquier batch de producto  $i$  en las primeras  $j$  etapas del camino  $r$ , es decir, incluye el tiempo de procesamiento en la unidad  $k$ , mientras que la ecuación (18) representa el tiempo de procesamiento de cualquier batch de producto  $i$  en las primeras  $j-1$  etapas del camino  $r$ , esto es, el tiempo de procesamiento en la unidad  $k$  no está incluido. Si  $j = 1$ , se define  $TU_{irk} = 0$ .

Sea  $t$  un punto genérico en la grilla de tiempos propuesta y  $t'$  el tiempo inicial de procesamiento del batch  $b$  de producto  $i$  en el camino  $r$ . Si  $t' = t - TL_{irk}$  el batch  $b$  finaliza su procesamiento en el punto de tiempo  $t$  en la unidad  $k$  y entonces el punto está disponible para procesar otro batch; si  $t' = t - TU_{irk}$ , el batch  $b$  comienza su procesamiento en el punto de tiempo  $t$  en la unidad  $k$ . Por lo tanto, si  $t' \in [t - TL_{irk} + 1, t - TU_{irk}]$  el batch  $b$  se estará procesando en el punto  $t$  en la unidad  $k$ .

Por lo tanto, en el punto de tiempo  $t$  en la unidad  $k$  de la etapa  $j$  se estará procesando como mucho un batch si se satisface la siguiente restricción:

$$\sum_{i \in I} \sum_{n \in N_i} \sum_{b \in IB_{in}} \sum_{\substack{r \in PR_{inb} \\ r \in RK_k}} \sum_{t'=t-TL_{irk}+1}^{t-TU_{irk}} y_{inbrt'} \leq 1 \quad \forall j \in J, k \in KJ_j, t \in T \quad (19)$$

Si dos batches  $b < b'$  del mismo producto son asignados al mismo camino  $r$  entonces, con el objetivo de evitar soluciones redundantes,  $b$  comienza su procesamiento antes que  $b'$ , esto es:

$$\sum_{\substack{t' \in TIR_{ir} \\ t' < t + t_{ik}}} y_{inb+1rt'} \leq 1 - y_{inbri} \quad \forall i \in I, n \in N_i, n \geq 2, b, b+1 \in IB_{in}, \quad (20)$$

$$t \in TIR_{ir}, k \in KJ_1, r \in PR_{inb}, r \in RK_k$$

## 2.5 Restricciones de satisfacción de demanda

La demanda de cada orden debe ser satisfecha antes de su fecha de vencimiento. Sea  $bs_{inbri}$  la variable no negativa que indica el tamaño del batch  $b$  de producto  $i$  si este es dividido en  $n$  batches y es procesado en el camino  $r$  con tiempo inicial  $t$ . Ese tamaño está acotado por los tamaños admisibles para los batches en el camino, como se expresa en la siguiente ecuación:

$$B_{ir}^{min} y_{inbri} \leq bs_{inbri} \leq B_{ir}^{max} y_{inbri} \quad \forall i \in I, n \in N_i, b \in IB_{in}, r \in PR_{inb}, t \in TIR_{ir} \quad (21)$$

Para cada orden  $d \in D_i$ , se introduce el conjunto  $TD_{dr}$  que contiene los puntos de tiempo  $t$  que satisfacen  $RD_i \leq t \leq DD_d - t_{ir}$ , de modo que si el batch  $b$  de producto  $i$  asignado al camino  $r$  comienza su procesamiento en un punto de tiempo  $t \in TD_{dr}$ , este batch completa su procesamiento a tiempo para ser utilizado para satisfacer la orden  $d$ .

Para asegurar que todas las órdenes se puedan completar a tiempo, debe satisfacerse lo siguiente:

$$\sum_{n \in N_i} \sum_{b \in IB_n} \sum_{r \in PR_{nb}} \sum_{t \in TD_{dr}} bs_{inbri} \geq \sum_{d'} Q_{d'} \quad \forall i \in I, d \in D_i, d' \in D_i, d' \leq d \quad (22)$$

Teniendo en cuenta que los elementos de  $D_i$  están ordenados en orden creciente, de acuerdo a sus fechas de vencimiento asociadas, la ecuación (22) garantiza que todo el requerimiento de producto  $i$  acumulado hasta la fecha de vencimiento de  $d$  puede ser satisfecho usando batches que han finalizado su procesamiento antes de esa fecha

Finalmente, el total de producto  $i$  procesado debe ser igual a su demanda:

$$\sum_{n \in N_i} \sum_{b \in IB_n} \sum_{r \in PR_{nb}} \sum_{t \in TIR_{ir}} bs_{inbri} = QT_i \quad \forall i \in I \quad (23)$$

## 2.6 Función objetivo

La función objetivo considerada en este trabajo consiste en minimizar el makespan ( $MK$ ), esto es, minimizar el tiempo total requerido para asegurar la satisfacción de todas las órdenes. La siguiente ecuación hace que  $MK$  sea mayor o igual que el tiempo final de procesamiento de todos los productos:

$$MK \geq (t + t_{ir}) y_{inbri} \quad \forall i \in I, n \in N_i, b \in IB_{in}, r \in PR_{inb}, t \in TIR_{ir} \quad (24)$$

El modelo se formula entonces de la siguiente manera:

$$\text{Minimizar } MK \tag{25}$$

sujeto a las restricciones (10)-(13), (15)-(16) y (19)-(24).

Dado que para satisfacer la demanda hay un número mínimo de batches que debe ser procesado y los tiempos de procesamiento en cada unidad es un dato conocido, es posible proponer una cota inferior a la función objetivo del problema. El valor de la cota inferior se obtiene considerando todas las posibles combinaciones para procesar esas cantidades mínimas de batches en cada etapa, asumiendo que el tiempo ocioso entre pares de batches es cero. De esta manera, se obtiene una buena cota inferior, que mejora notablemente la performance computacional del modelo.

### 3 Ejemplo ilustrativo

Se considera una planta consistente en tres etapas y dos unidades en cada etapa, cuya estructura se muestra en la Figura 3.

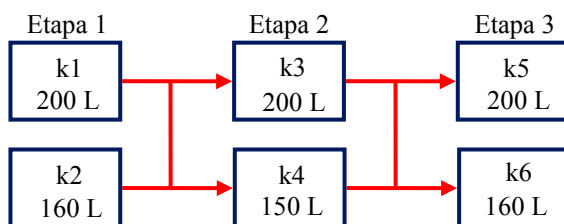


Fig. 3. Estructura de la planta

De acuerdo a la estructura de la planta, es posible definir 8 caminos diferentes. Las unidades que componen cada camino, como así también los tamaños mínimo y máximo de batch en cada uno de ellos, se muestran en la Tabla 1.

Tabla 1. Definición de los caminos y tamaños admisibles para los batches

Camino	Composición del camino			$\forall i, B_{ir}^{min}$ (kg)	$\forall i, B_{ir}^{max}$ (kg)
	Etapa1	Etapa2	Etapa3		
r1	k1	k3	k5	140	200
r2	k1	k3	k6	140	160
r3	k1	k4	k5	140	150
r4	k1	k4	k6	140	150
r5	k2	k3	k5	140	160
r6	k2	k3	k6	140	160
r7	k2	k4	k5	140	150
r8	k2	k4	k6	112	150

En este ejemplo, para simplificar la lectura de los datos y resultados, se considera que el porcentaje mínimo de ocupación es del 70% en todas las unidades para todos

los productos y los factores de tamaño son todos iguales a 1. Por lo tanto, los tamaños mínimos y máximos de batch es posible calcularlos independientemente del producto.

La planta puede procesar un conjunto de cuatro productos {i1, i2, i3, i4}. Debe satisfacerse un conjunto de diez órdenes {d1, d2, ..., d10} cuya relación con los productos, demandas, fechas de emisión y vencimiento se presentan en la Tabla 2.

**Tabla 2.** Demandas, fechas de emisión y vencimiento, y tiempos de procesamiento

Producto	Orden	$Q_d$ (kg)	$QT_i$ (kg)	$RD_i$ (h)	$DD_d$ (h)	Tiempo $t_{ik}$ (h)					
						k1	k2	k3	k4	k5	k6
i1	d1	240			20						
	d2	120	600	0	28	4	3	4	3	4	3
	d3	240			38						
i2	d4	160	590	0	20	4	5	4	5	4	5
	d5	430			28						
i3	d6	240			20						
	d7	130	660	0	28	4	3	5	3	4	3
	d8	290			38						
i4	d9	180	650	0	20	2	3	2	3	3	2
	d10	470			38						

El modelo fue implementado en GAMS versión 24.7.4 en un procesador Intel Core i7, 8 GB de RAM y 3.6 GHz. El problema MILP fue resuelto utilizando el resolvidor CPLEX 12.6.3. El modelo, que cuenta con 19982 restricciones, 7012 variables discretas y 4518 variables continuas, requirió 109 segundos de CPU para alcanzar el óptimo, con gap 0%. El valor óptimo de la función objetivo es de 32 horas.

**Tabla 3.** Batching y scheduling óptimos

Producto	Batch	Tamaño	Camino	Inicio (h)	Finalización (h)
i1	b1	150	r8	3	12
	b2	150	r8	9	18
	b3	150	r8	15	24
	b4	150	r8	18	27
i2	b1	200	r1	2	14
	b2	200	r1	11	23
	b3	190	r1	15	27
i3	b1	200	r1	6	19
	b2	200	r1	19	32
	b3	148	r8	6	15
	b4	112	r8	12	21
i4	b1	200	r1	0	7
	b2	160	r2	26	32
	b3	150	r4	23	30
	b4	140	r8	0	8

La Tabla 3 resume los resultados obtenidos, indicando los batches procesados para cada producto y sus tamaños. La misma tabla muestra la asignación de cada batch al camino donde se procesa, y su tiempo inicial y final de procesamiento.

Es importante remarcar que, si no se hiciera la consolidación de órdenes, por ejemplo en el caso del producto i1, debido al tamaño de las órdenes, se requerirían al menos dos batches para la orden d1, uno para d2 y dos para d3, es decir, al menos 5 batches. Al realizar la consolidación, se procesaron en total 4 batches de producto i1.

Con la solución provista por el modelo es posible satisfacer todas las órdenes antes de su fecha de vencimiento.

La Figura 4 muestra el diagrama de Gantt de la solución óptima

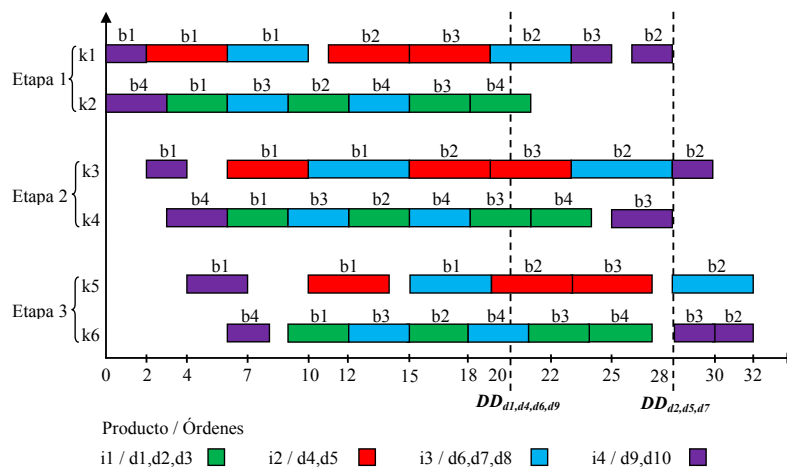


Fig. 4. Diagrama de Gantt de la solución óptima

## 4 Conclusiones

El modelo desarrollado en este trabajo resuelve exitosamente el problema de consolidación de múltiples órdenes de cada producto para plantas batch multiproducto multietapa con unidades diferentes operando en paralelo en cada etapa, que no cuentan con almacenamiento intermedio entre etapas y operan adoptando la política de transferencia de espera cero (ZW). La formulación propuesta determina la demanda total por producto, y resuelve el problema de batching sobre esas cantidades, asegurando que cada orden sea completada antes de su fecha de vencimiento. Bajo las condiciones anteriores, los problemas de batching y scheduling son resueltos simultáneamente, esto es, el modelo provee el número de batches en que cada producto es dividido, el camino de producción donde se procesa cada batch y su tiempo inicial y final de procesamiento, resultando en una operación eficiente de la planta.

El modelo presentado en este trabajo requiere de un esfuerzo computacional significativo cuando se procesa un gran número de batches de cada producto, dado el

carácter combinatorio del problema. Sin embargo, la formulación propuesta resuelve problemas de considerable tamaño en tiempos computacionales razonables, los cuales no son mostrados por cuestiones de espacio, proveyendo soluciones que mejoran las obtenidas por métodos que no realizan consolidación de órdenes, disminuyendo el número de batches a procesar, mejorando la utilización del equipamiento y reduciendo el tiempo necesario para procesar todas las órdenes.

## Referencias

1. Méndez, C. A., Cerdá, J., Grossmann, I. E., Harjunkski, I., Fahl, M.: State-of-the-art review of optimization methods for short-term scheduling of batch processes. *Comput. Chem. Eng.* (2006) 30(6–7), 913–946.
2. Maravelias, C. T.: General framework and modeling approach classification for chemical production scheduling. *AIChE Journal* (2012) 58(6), 1812–1828.
3. Harjunkski, I., Maravelias, C., Bongers, P., Castro, P., Engell, S., Grossmann, I., Hooker, J., Méndez, C., Sand, G., Wassick, J.: Scope for industrial applications of production scheduling models and solution methods. *Comput. Chem. Eng.* (2014) 62, 161–193.
4. Castro, P. M., Grossmann, I. E., Zhang, Q.: Expanding scope and computational challenges in process scheduling. *Comp. Chem. Eng.* (2018) 114, 14–42.
5. Lim, M., Karimi, A. I.: A slot-based formulation for single-stage multiproduct batch plants with multiple orders per product. *Ind. Eng. Chem. Res.* (2003) 42, 1914–1924.
6. Prasad, P., Maravelias, C. T.: Batch selection, assignment and sequencing in multi-stage multi-product processes. *Comput. Chem. Eng.* (2008) 32, 1106–1119.
7. Sundaramoorthy, A., Maravelias, C. T.: Simultaneous batching and scheduling in multistage multiproduct processes. *Ind. Eng. Chem. Res.* (2008) 47, 1546–1555.
8. Sundaramoorthy, A., Maravelias, C. T.: Modeling of storage in batching and scheduling of multistage processes. *Ind. Eng. Chem. Res.* (2008) 47, 6648–6660.
9. Marchetti, P. A., Méndez, C. A., Cerdá, J.: Mixed-Integer Linear Programming Monolithic Formulations for Lot-Sizing and Scheduling of Single-Stage Batch Facilities. *Ind. Eng. Chem. Res.* (2010) 49, 6482–6498.
10. Marchetti, P. A., Méndez, C. A., Cerdá, J.: Simultaneous lot sizing and scheduling of multistage batch processes handling multiple orders per product. *Ind. Eng. Chem. Res.* (2012) 51, 5762–5780.
11. Fumero, Y., Corsano, G., Montagna, J. M.: Simultaneous batching and scheduling of batch plants that operate in a campaign-mode, considering nonidentical parallel units and sequence-dependent changeovers. *Ind. Eng. Chem. Res.* (2014) 53, 17059–17074.
12. Sundaramoorthy, A., Maravelias, C. T., Prasad, P.: Scheduling of multistage batch processes under utility constraints. *Ind. Eng. Chem. Res.* (2009) 48, 6050–6058.
13. Sundaramoorthy, A., Maravelias, C. T.: Computational study of network-based mixed-integer programming approaches for chemical production scheduling. *Ind. Eng. Chem. Res.* (2011) 50(9), 5023–5040.
14. Merchan, A. F., Lee, H., Maravelias, C. T.: Discrete-time mixed-integer programming models and solution methods for production scheduling in multistage facilities. *Comput. Chem. Eng.* (2016) 94, 387–410.
15. Ackermann, S., Fumero, Y., Montagna, J. M.: New problem representation for the simultaneous resolution of batching and scheduling in multiproduct batch plants. *Ind. Eng. Chem. Res.* (2021) 60(6), 2523–2535.