

# Asignación de Capacidades de Almacenamiento de Inventario en Proceso en una Línea de Producción

## Allocation of Intermediate Buffer Storage Capacities in a Production Line

Mario C. Vélez-Gallego ([marvelez@eafit.edu.co](mailto:marvelez@eafit.edu.co))  
Departamento de Ingeniería de Producción, Universidad EAFIT  
Carrera 49 7 sur 50, Medellín, Colombia

Jhull B. Jaramillo-Jiménez ([jjaram61@eafit.edu.co](mailto:jjaram61@eafit.edu.co))  
Departamento de Ingeniería de Producción, Universidad EAFIT  
Carrera 49 7 sur 50, Medellín, Colombia

### Resumen

*En esta propuesta de investigación se considera una línea de producción con demanda aleatoria y fallas aleatorias en cada centro de trabajo. Cada vez que una estación de trabajo falla, se realiza un mantenimiento correctivo de duración aleatoria. Para mitigar el efecto de las fallas aleatorias en el desempeño del sistema, se utilizan zonas de almacenamiento de inventario (buffers) entre centros de trabajo, cada zona está asociada a un costo de mantenimiento de inventario el cual incrementa con el estado del proceso. El objetivo de la investigación es determinar la capacidad de cada buffer de forma que el costo total promedio por unidad de tiempo se minimice mientras se mantiene el nivel promedio de servicio de la línea por encima de un valor mínimo. Para resolver el problema se propone un heurístico de simulación-optimización que permite encontrar una solución factible a un costo computacional razonable.*

**Palabras Clave:** Gestión de inventarios, fallas en máquinas, simulación

### Abstract

*In this research proposal we consider a production line with random demand and subject to random failures at each workstation. Every time a workstation fails, a corrective maintenance activity is triggered to repair the workstation. In order to palliate the effect of the random failures in the performance of the system, intermediate buffers are placed in-between workstations. An inventory holding cost that increases with the stage of the process is associated to each buffer. The research objective in this work is to allocate capacity to each intermediate buffer in the line so that the average cost per time unit is minimized while the average service level is kept above a minimum value. In this paper we assume that unsatisfied demand is lost and the service level is defined as the long term proportion of satisfied demand. A greedy simulation-based heuristic is presented to find a feasible solution to the problem in a reasonable computational time.*

**Keywords:** Inventory management, machine failures, simulation

## 1. Introducción

Como se establece en Rezg, Dellagi y Chelbi (2008) al buscar minimizar el impacto de los paros en la producción, debido a actividades de reparación, se deben tomar decisiones para controlar el inventario basadas no solo en la demanda esperada y en el nivel de servicio deseado, sino también en la disponibilidad de las máquinas. Hoy en día, la literatura académica ha abordado en su mayoría sistemas de una sola máquina, y un poco los sistemas

más complejos como *flow shop* y *job shop*. En la práctica, los profesionales en producción y mantenimiento tienen que lidiar con sistemas mucho más complejos para lo cual no hay modelos disponibles ni herramientas que soporten la toma de decisiones complejas que deben tomar. El objetivo de esta investigación es desarrollar un modelo más general que ayude a los profesionales en la determinación de las capacidades para zonas de almacenamiento (*buffers*) intermedias con parámetros de producción más generales, al no tener restricciones impuestas respecto a la distribución de las variables aleatorias involucradas. En este artículo se considera un sistema de producción organizado como un sistema de ensamble con capacidad de almacenamiento finita entre centros de trabajo e infinita en ambos extremos de la línea. Los centros de trabajo están sujetos a fallas aleatorias que implican la realización de un mantenimiento correctivo, que a su vez tiene duración aleatoria. En cada *buffer* hay un costo de mantener el inventario que incrementa monótonamente con el estado del proceso, es decir, entre más atrás la etapa en la línea productiva menor costo de mantener inventario. El sistema de producción estudiado es operado bajo una estrategia *push* y está sujeta a demanda variable. Si cuando llega una orden de un cliente no hay inventario de producto terminado, la demanda asociada con esa orden es considerada como venta perdida. El problema que se desea abordar en esta investigación es la determinación de los tamaños de los *buffers* entre centros de trabajo que minimice el costo total relevante mientras la probabilidad de desabastecimiento se mantiene por debajo de un valor mínimo dado. En este trabajo excluimos del análisis los *buffers* en ambos extremos del sistema de producción (p.e. *buffers* de materia prima y producto terminado). Además se asume que éstos son de capacidad infinita y que la materia prima siempre está disponible para ser procesada. Más formalmente, sea  $\Pi = \{M, D, B\}$  un sistema de producción arbitrario (p.e. una línea de ensamble) compuesta por un conjunto de máquinas (p.e.  $M$ ), algunos procesos estocásticos  $D$  que representan la demanda de producto terminado del *buffer* final de producto terminado en el sistema (p.e. producto terminado), y un conjunto de *buffers* intermedios (p.e.  $B$ ). Se asume que las funciones de probabilidad de densidad de los tiempos de proceso, tiempos entre fallas y tiempos de reparación son conocidas para cada máquina  $m \in M$ . También se asume que los procesos estocásticos  $D$  que representan la demanda en el sistema son conocidos. Sean  $N$  y  $H$  dos vectores que contienen los tamaños y los costos de mantenimiento unitario asociados con el *buffer*  $B$ , y sea  $N_i$  y  $H_i$  los tamaños y el costo unitario de mantenimiento del *buffer*  $i \in B$  respectivamente. Como se asume que el vector  $H$  es conocido previamente, las variables de decisión son los tamaños de los *buffers* (p.e. vector  $N$ ), y sea  $\theta(N)$  el nivel de servicio promedio observado en el sistema, definido en términos de una proporción de las unidades demandadas que son satisfechas por el inventario de producto terminado si el sistema opera con tamaños de *buffers*  $N$ . El objetivo de esta investigación es encontrar un vector  $N^*$  que minimice  $C(N^*)$  mientras se mantiene el  $\theta(N^*)$  por encima de un valor mínimo preestablecido  $\Phi$ . En el resto de este artículo, el vector  $N$  será tratado como la solución al problema bajo estudio.

## 2. Revisión de la literatura

Considerar simultáneamente el control de inventario y la confiabilidad de máquinas ha llegado a ser un área importante de la investigación en los últimos años. Enfoques recientes para una sola máquina incluye Chelbi y Ait-Kadi (2004), Gharbi, Kenne y Beit (2007), Rezg, Chelbi y Xie (2005), Rezg, Dellagi y Chelbi (2008); donde se desarrollaron modelos matemáticos bajo los supuestos de ventas perdidas y retrasos. En algunas referencias los autores proponen modelos analíticos para encontrar los tamaños de *buffers* y la frecuencia de realización de mantenimientos preventivos de forma que el costo promedio total por unidad

producida sea mínimo. Otros enfoques más generales extienden el problema a una línea de producción con inventarios intermedios como Demir, Tunali y Løkketangen (2010), Dolgui, Eremeev, Kolokolov y Sigaev (2002), Nahas, Ait-Kadi y Nourelfath (2006), Nourelfath, Nahas y Ait-Kadi (2005). En Dolgi et al. (2002) los autores proponen un algoritmo genético (GA) para asignar capacidades de *buffer* en una línea de producción bajo los supuestos de tiempos de procesamiento determinísticos y tiempos de falla y de reparación exponencialmente distribuidos. Allí los autores proponen una aproximación agregada para evaluar asignaciones tentativas de capacidades de *buffers* y un GA para minimizar el costo promedio de inventario de acuerdo al estado del producto. En Nourelfath et al. (2005) los autores presentan una variación del problema en el cual para cada centro de trabajo en la línea de producción hay varias versiones disponibles en el mercado, cada una con una tasa de producción, disponibilidad y precio diferente, y el objetivo que propone es maximizar el *throughput* del sistema de producción manteniendo el costo total bajo cierto valor dado. En Nahas et al. (2006) un heurístico de búsqueda local es usado para maximizar el *throughput* de una línea de producción al aproximar el flujo de partes discretas en una línea de producción a un flujo continuo. Más recientemente una solución aproximada de búsqueda Tabú fue propuesta en Demir et al. (2010) para resolver el mismo problema que en Nahas et al. (2006) presentando mejores resultados con respecto al costo computacional.

En Massim, Yalaoui, Amodeo, Chatelet y Zebalah (2010) se propone un algoritmo de optimización para determinar un tamaño de *buffers* apropiado para reducir el costo de fabricación mientras se mantiene una tasa de producción deseada. En Zequeira, Prida y Valdés (2004) y Radhoui, Rezg y Chelbi (2009) un modelo matemático es desarrollado para un sistema de fabricación que considera fallas aleatorias y la producción de partes no conformes. El objetivo del modelo es decidir cuando realizar mantenimientos con el objetivo de asegurar un suministro continuo de producto terminado mientras se minimiza el costo total esperado por unidad de tiempo. En Shi y Gershwin (2009) se considera una asignación óptima de *buffers* en un ambiente con tasa de producción determinística (p.e. constante) y con tiempos de falla y reparación geoméricamente distribuidos. Se propone un modelo matemático con restricciones no lineales junto a un algoritmo para resolver el modelo. En Spinellis y Papadopoulos (2000) se presenta un enfoque de recocido simulado (*simulated annealing*) para resolver el problema de asignación de *buffers* en líneas confiables. El objetivo propuesto por estos autores es maximizar el *throughput* promedio de la línea. De forma similar, un modelo de búsqueda Tabú es propuesto en Demir, Tunali y Løkketangen (2012) para el problema de asignación de *buffers* en una línea de producción con máquinas confiables con el objetivo de maximizar el *throughput*. En Zequeira, Valdes y Berenguer (2008) los periodos entre las actividades de mantenimiento preventivo son considerados. En Amiri y Mohtashami (2012) un modelo de simulación fue desarrollado para resolver el problema de asignación de *buffers* en sistemas de producción poco confiables bajo funciones de densidad de probabilidad generales para los tiempos de proceso, tiempo entre fallas y tiempo de reparación. Una propuesta de GA combinado con una técnica de búsqueda lineal fue propuesta con el objetivo de maximizar el *throughput* del sistema.

El problema abordado en esta investigación es similar al considerado en Amiri y Mohtashami (2012). La diferencia entre este trabajo y el planteado en este artículo es que (1) se asume que el costo de mantenimiento de inventario es diferente en cada *buffer* y (2) el objetivo en este trabajo es minimizar el costo total promedio por unidad de tiempo mientras se mantiene el nivel promedio de servicio de la línea por encima de un valor mínimo. No se han encontrado

evidencias, basado en la revisión bibliográfica realizada, de trabajos publicados que consideren el problema bajo los supuestos de (1) costos unitarios diferentes en cada *buffer*, (2) funciones de densidad de probabilidad arbitrarias para los tiempos de procesamiento, tiempos entre fallas y de reparación y (3) configuraciones arbitrarias para la línea de ensamble bajo estudio.

### 3. Metodología

Los procesos formales de investigación enmarcados en el ámbito de la administración de operaciones y en los cuales se usan modelos cuantitativos se pueden clasificar, de acuerdo con la tipología propuesta por Bertrand y Fransoo (2002) en dos grupos: investigación axiomática e investigación empírica. En esta clasificación la investigación axiomática es aquella soportada en un modelo teórico desarrollado por el investigador con el fin de ganar intuición sobre la estructura del problema en estudio. Estos modelos son denominados teóricos ya que aunque su motivación surge de un sistema real, el objetivo del investigador no es contrastar sus resultados con información obtenida del mismo, sino construir una abstracción del problema que pueda ser resuelta por métodos formales y cuya solución brinde información nueva que permita entender y mejorar el desempeño de los sistemas estudiados. En el desarrollo de un modelo axiomático el investigador reconoce el modelo no necesariamente va a representar el comportamiento del sistema real que motivó su desarrollo, ya que por la complejidad del sistema real muchas simplificaciones son necesarias para llegar a un modelo que sea posible resolver. El desarrollo de estos modelos se apoya fundamentalmente en ramas del conocimiento como las matemáticas, la estadística y las ciencias de la computación. Ejemplos clásicos de modelos axiomáticos son los modelos de control de inventarios, teoría de colas y teoría de confiabilidad. De otro lado una investigación se considera empírica si, a diferencia de la axiomática, el investigador tiene por objetivo desarrollar un modelo que describa lo mejor posible el comportamiento del sistema real que motivó su desarrollo, y sus resultados deben ser necesariamente validados con información obtenida del sistema real (i.e. modelo de curva de aprendizaje y de series de tiempo para pronósticos de demanda). En este artículo se describe el desarrollo de una investigación axiomática soportada en un modelo cuantitativo de simulación–optimización. La metodología utilizada para esta investigación se basa en la sugerida por Bertrand y Fransoo (2002) para el caso particular del desarrollo de modelos cuantitativos axiomáticos que involucren modelos de simulación. La metodología utilizada consta de las siguientes etapas: (i) descripción del modelo conceptual, (ii) justificación del método de solución, (iii) descripción de modelo científico, (iv) descripción de método de solución, (v) diseño experimental y (vi) análisis e interpretación de resultados.

### 4. Solución Propuesta

La solución propuesta es un modelo de simulación basado en un heurístico que parte de una solución inicial  $N_0$  con tamaños de *buffer* arbitrarios que satisfacen la única condición de que estos tamaños son lo suficientemente pequeños para que la solución sea infactible con respecto a  $\theta(N_0)$ . En la iteración  $k$  el heurístico propuesto transforma  $N_k$  en  $N_{k+1}$  al incrementar  $n_{i^*} \in N_k$  en  $\Delta$  unidades. Sea  $N_k\{i, \Delta\}$  una nueva solución obtenida de  $N_k$  al incrementar  $n_i$  en  $\Delta$  unidades de tamaño, entonces  $i^* = \operatorname{argmin}\{m_i | i \in N_k\}$  y  $m_i$  es calculado de acuerdo a la ecuación (1).

$$m_i = \frac{C(N_k\{i, \Delta\}) - C(N_k)}{\theta(N_k\{i, \Delta\}) - \theta(N_k)} \quad (1)$$

Como la solución inicial  $N_0$  es infactible con respecto a  $\theta(N_0)$  (p.e.  $\theta(N_0) < \Phi$ ) y durante la ejecución del algoritmo en cada iteración, uno de los tamaños de un *buffer* es incrementado en  $\Delta$  unidades hasta que en alguna iteración  $k$  el valor de  $\theta(N_k)$  es mayor o igual a  $\Phi$  por primera vez. En este punto el algoritmo finaliza con la solución factible al problema. Para seleccionar el *buffer* al cual se le incrementará la capacidad, un costo marginal por unidad de nivel de servicio es estimado para cada *buffer* en el sistema. Este costo marginal por unidad de nivel de servicio es calculado para cada *buffer* en  $B$  (p.e.  $m_i$ ) y calculado con la ecuación (1). El *buffer* con el menor costo marginal (p.e.  $i^*$ ) es seleccionado y su tamaño es incrementado en  $\Delta$  unidades de tamaño. Este proceso continua hasta alcanzar el nivel de servicio deseado (p.e.  $\Phi$ ). Sea  $x \leftarrow S(N)$  una función que devuelve en  $x$  valor esperado de nivel de servicio si el sistema de producción bajo estudio es operado con los tamaños de *buffers* en  $N$ , y sea  $y \leftarrow C(N)$  una función similar que regresa en  $y$  el valor esperado de costo por unidad de tiempo. El pseudocódigo para la solución propuesta es la siguiente:

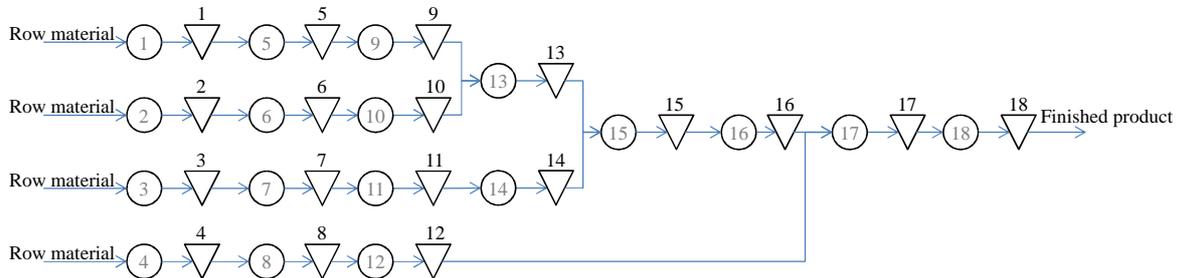
```

N* ← H(M, D, Φ, Δ)
k ← 0
Nk ← {n1, n2, ...}
S0 ← S(N0), C0 ← C(N0)
while Sk < Φ do
    i* ← argmax {mi | i ∈ Nk}
    Nk+1 ← Nk{i*, Δ}
    Sk+1 ← S(Nk+1), Ck+1 ← C(Nk+1)
    k ← k + 1
end
N* ← Nk
end

```

## 5. Experimentos computacionales

Para probar el desempeño del heurístico propuesto se usó como instancia de prueba de la línea de ensamble propuesta en Amiri y Mohtashami (2012) y representado en la figura 1, donde los círculos representan la máquina y los triángulos representan los *buffers* intermedios.



**Figura 1.** Sistema de producción usado para las pruebas.

El sistema utilizado para probar está compuesto de 17 *buffers* y 18 máquinas. La función de densidad de los tiempos de proceso, tiempos entre fallas y tiempos de reparación son

presentadas en la tabla 1. Las letras E, G y W representan la función de densidad exponencial, gamma y weibull respectivamente.

**Tabla 1.** Descripción de las máquinas en el sistema de producción usado para pruebas

<b>Estación</b>	<b>Tiempo de proceso</b>	<b>Tiempo entre fallas</b>	<b>Tiempo de reparación</b>
1	W(20,10,2)	W(1000,70,20)	G(1000,250,80)
2	W(25,11,4)	W(9000,200,18)	G(1200,250,80)
3	W(30,11,4)	W(9500,200,18)	G(200,250,80)
4	W(20,10,6)	W(9000,200,18)	G(1250,250,80)
5	W(35,4,3)	W(15000,130,20)	G(400,300,52)
6	W(37,9,6)	W(17000,240,90)	G(2354,300,52)
7	W(36,10,6)	W(18000,240,90)	G(2000,325,52)
8	W(40,9,6)	W(1700,240,90)	G(2000,300,52)
9	W(30,11,9)	W(11000,350,18)	G(500,300,52)
10	W(45,16,9)	W(9000,180,40)	G(700,300,52)
11	W(46,12,9)	W(9000,180,40)	G(756,300,52)
12	W(48,11,12)	W(9400,140,70)	G(730,300,52)
13	E(40)	W(12000,200,80)	G(1500,300,52)
14	W(44,17,9)	W(8650,180,40)	G(700,300,52)
15	W(27,12,10)	W(8100,300,25)	G(2659,140,60)
16	E(43)	W(8230,250,30)	G(2768,160,65)
17	W(26,12,10)	W(7700,324,25)	G(2546,140,60)
18	W(33,14,10)	W(7900,400,25)	G(2434,143,60)

De acuerdo al *buffer*, la instancia propuesta en Amiri y Mohtashami (2012) incluye un tamaño de inventario que se usa como solución inicial (p.e.  $N_0$ ). El problema abordado en el artículo, mencionado arriba, no considera diferentes costos de mantener inventario, en esta investigación se asignaron unos costos de mantenimiento por unidad de tiempo arbitrarios para cada *buffer* que aumenta con la etapa del *buffer* en el sistema. Un resumen de la información más relevante es presentado en la tabla 2. Para el proceso estocástico  $D$  que modela la demanda observada al final del sistema productivo, se asumen las llegadas de clientes de acuerdo a un proceso *Poisson* homogéneo de tasa  $\lambda$  y que cada cliente que llega ordena una unidad. Para probar la tasa propuesta se estableció arbitrariamente de  $\lambda=0,003159$  órdenes de clientes por unidad de tiempo.

**Tabla 2.** Descripción de parámetros de *buffers* usados para la prueba

<b>Buffer <math>i</math></b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>10</b>	<b>11</b>	<b>12</b>	<b>13</b>	<b>14</b>	<b>15</b>	<b>16</b>	<b>17</b>
$N_i$	200	180	250	170	200	240	160	210	150	150	220	220	140	140	230	200	550
$H_i$	1	1	1	1	2	2	2	2	3	3	3	3	4	4	5	6	7

El algoritmo propuesto fue implementado en Visual Basic for Applications® (VBA) y ejecutado en un computador equipado con cuatro procesadores Intel® Core i7 y 6 GB de memoria, con Windows 7 a 64 bits. Como parámetros se uso  $\Phi=0.95$  y  $\Delta=5$  unidades. El tiempo computacional requerido del algoritmo para terminar esta prueba fue de 58542 segundos. La evolución del nivel de servicio del heurístico en la búsqueda evolucionada de una solución factible es descrita en la figura 2. Como se puede observar en la figura, el valor

esperado de nivel de servicio no es exactamente  $\Phi=0.95$  al terminar. Esto se debe a que la prueba de si el sistema ha alcanzado o no un nivel específico de nivel de servicio es una prueba estadística. El algoritmo finaliza cuando no hay evidencia estadística de que el valor de nivel de servicio sea estrictamente inferior a  $\Phi$ . De forma similar, la figura 3 representa la evolución del costo promedio por unidad de tiempo del heurístico en el tiempo.

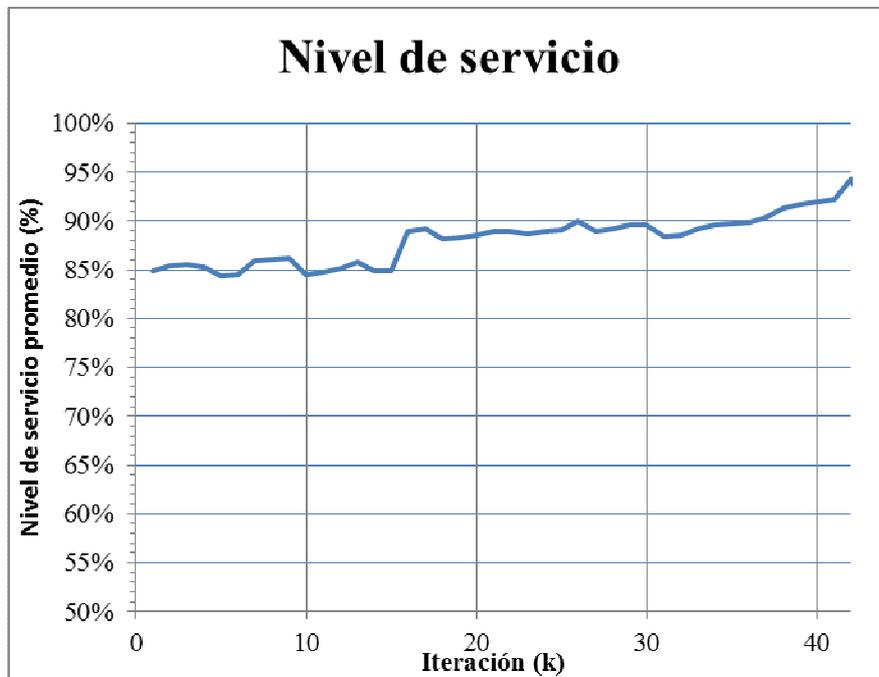


Figura 1. Evolución del valor de nivel de servicio



Figura 2. Evolución del costo promedio por unidad de tiempo

En la tabla 3 se presenta una comparación entre la solución inicial y la final, donde  $N_0$  y  $N_f$  representan vector de tamaños para la solución inicial y final respectivamente, mientras que

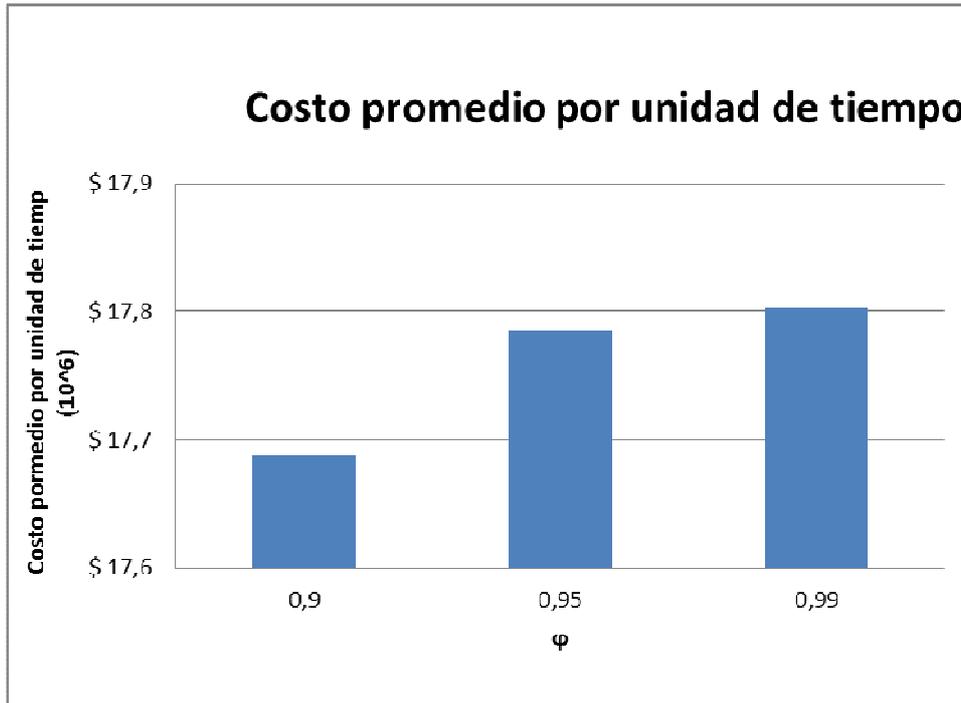
%Gap es el aumento relativo en el tamaño observado de cada *buffer*, calculado con respecto al tamaño inicial.

**Tabla 2.** Solución inicial y final

<b><i>Buffer i</i></b>	<b><i>N<sub>0</sub></i></b>	<b><i>N<sub>F</sub></i></b>	<b><i>%Gap</i></b>
<b>1</b>	200	205	2,50%
<b>2</b>	180	195	8,33%
<b>3</b>	250	255	2,00%
<b>4</b>	170	170	0,00%
<b>5</b>	200	210	5,00%
<b>6</b>	240	250	4,17%
<b>7</b>	160	165	3,13%
<b>8</b>	210	230	9,52%
<b>9</b>	150	165	10,00%
<b>10</b>	150	160	6,67%
<b>11</b>	220	240	9,09%
<b>12</b>	220	220	0,00%
<b>13</b>	140	160	14,29%
<b>14</b>	140	160	14,29%
<b>15</b>	230	255	10,87%
<b>16</b>	200	205	2,50%
<b>17</b>	550	565	2,73%

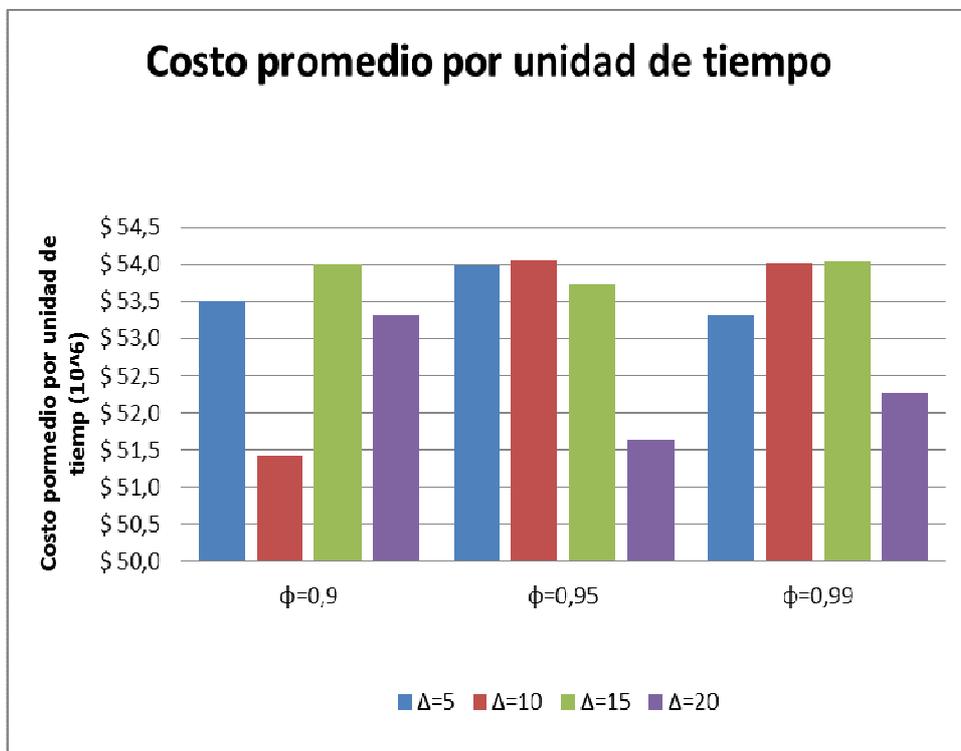
Con el objetivo de observar el comportamiento modelo propuesto, se realizó una simulación variando tres parámetros del modelo  $\lambda$ ,  $\Phi$  y  $\Delta$ . Para  $\lambda$  se utilizaron las proporciones  $0.9\lambda_i$ ,  $1\lambda_i$  y  $1,1\lambda_i$ , donde  $\lambda_i=0,003159$  clientes por unidad de tiempo; como valores de  $\Phi$  se utilizaron 90%, 95% y 99%, y como valores de  $\Delta$  5,10, 15, 20. Los comportamientos observados al variar cada parámetro se muestran a continuación.

Al aumentar el nivel de servicio deseado ( $\Phi$ ), se observa que a mayor  $\Phi$  mayor costo promedio por unidad de tiempo. Este comportamiento se muestra en la figura 4 y además se observa que este incremento no tiene una proporción constante.



**Figura 4.** Costo promedio por unidad de tiempo, variando  $\Phi$

Al variar el incremento realizado ( $\Delta$ ), como se observa en la figura 5 y 6 no hay comportamiento general en el costo promedio por unidad de tiempo. Lo que sí se puede observar es que con un  $\Delta$  menor la variación del costo entre iteraciones es menor, esto se observa en la figura 7.



**Figura 5.** Costo promedio por unidad de tiempo, variando  $\Delta$  y  $\Phi$

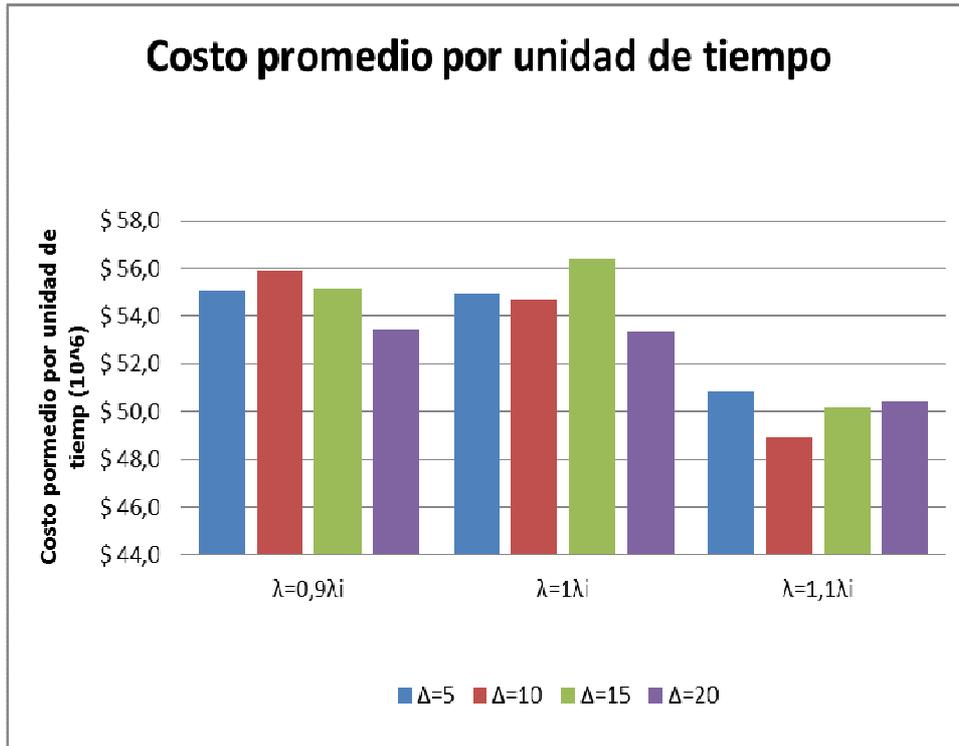


Figura 6. Costo promedio por unidad de tiempo, variando  $\Delta$  y  $\lambda$

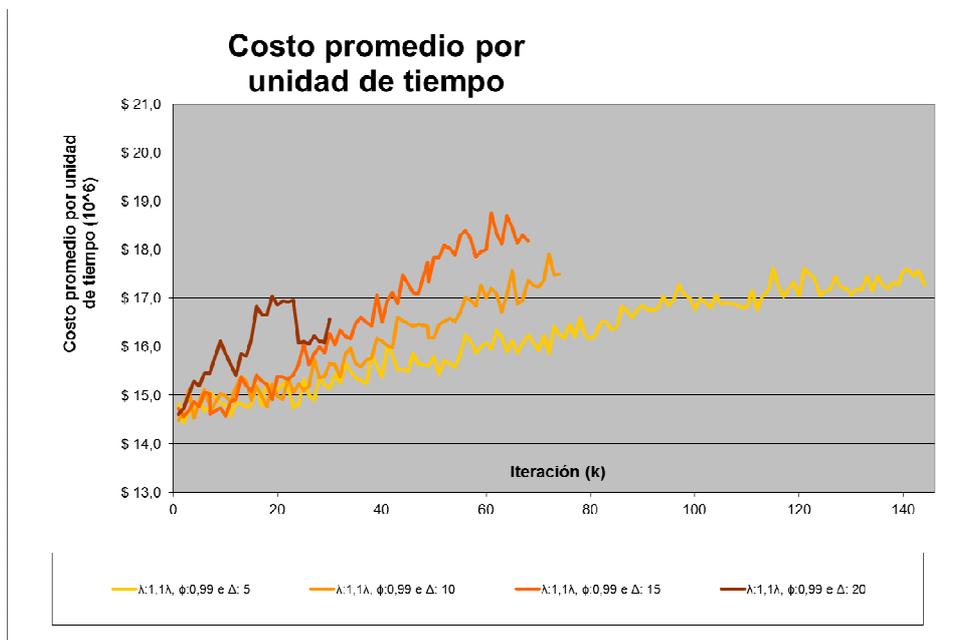


Figura 7. Evolución del costo promedio por unidad de tiempo,

En la figura 8 se observa que el costo promedio por unidad de tiempo al variar la tasa de llegada de clientes ( $\lambda$ ), es menor a mayor  $\lambda$ . Esto se debe a que a mayor tasa de llegada de clientes menor tiempo de almacenamiento será necesario para los productos. Por otro lado se observa que cuando la tasa disminuye llega un momento donde el sistema se satura y en consecuencia el costo de almacenamiento no aumenta significativamente.

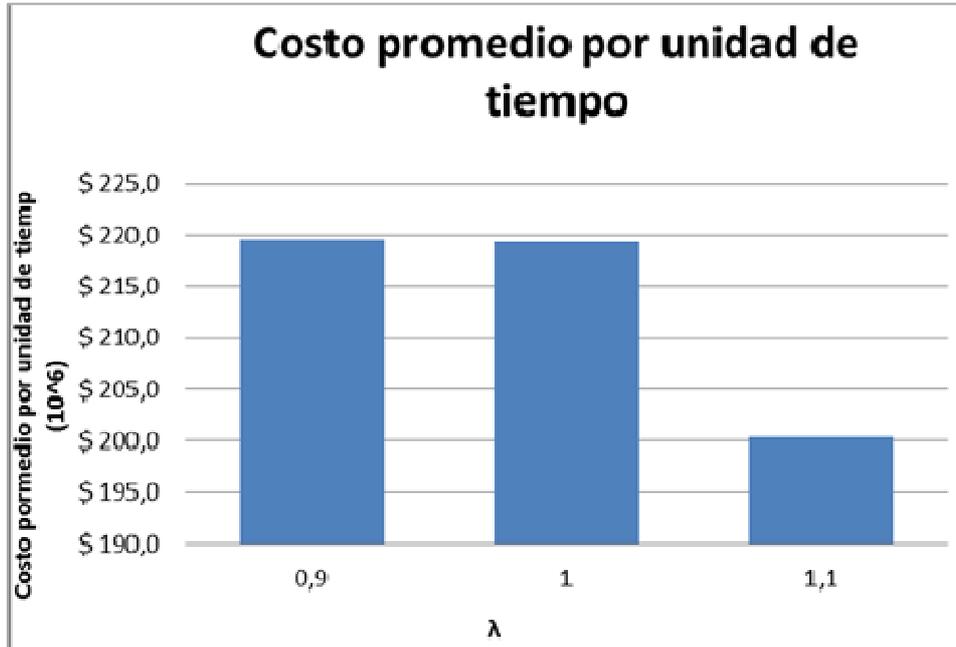


Figura 8. Costo promedio por unidad de tiempo, variando  $\lambda$

Por otro lado al realizar un análisis del tiempo de simulación requerido para encontrar la solución factible, como se observa en la figura 9 a mayor nivel de servicio deseado, mayor tiempo de simulación. Además la proporción en la cantidad de tiempo de simulación requerido para encontrar una solución factible respecto al nivel de servicio deseado cada vez es mayor, como se observa en la figura, la diferencia en tiempo entre un  $\Phi=99\%$  y  $\Phi=95\%$  es mayor que la que hay entre  $\Phi=95\%$  y  $\Phi=90\%$ .

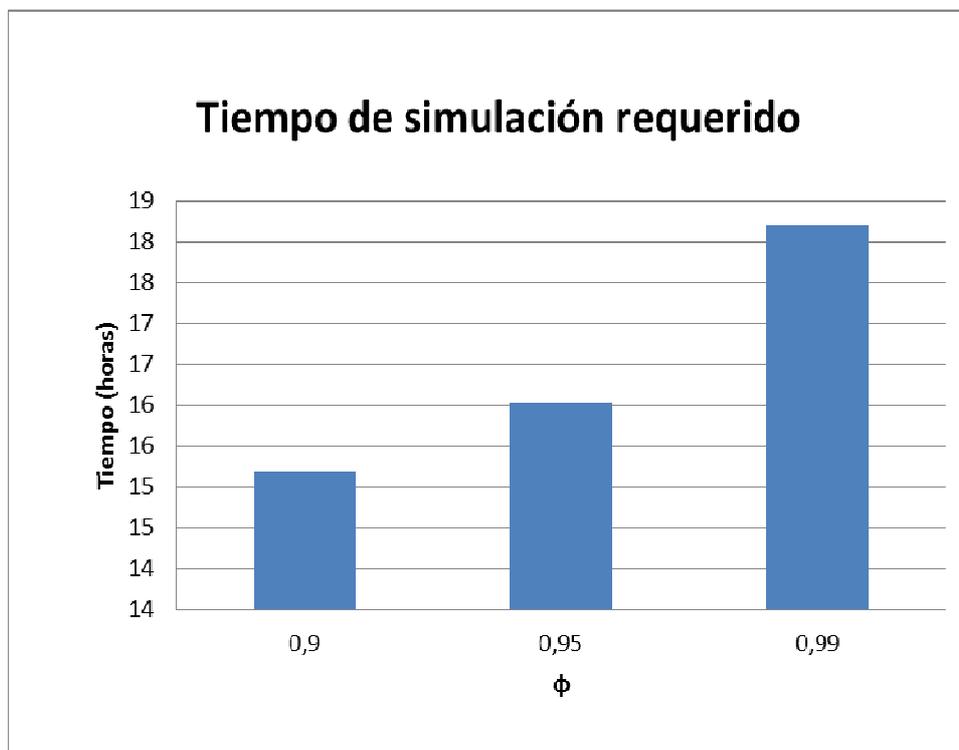


Figura 9. Tiempo de simulación requerido, variando  $\Phi$

Por otro lado al realizar incrementos ( $\Delta$ ) mayores en las capacidades de los *buffers*, el tiempo de simulación requerido es menor, como se observa en la figura 10; es de aclarar que la proporción no es lineal.

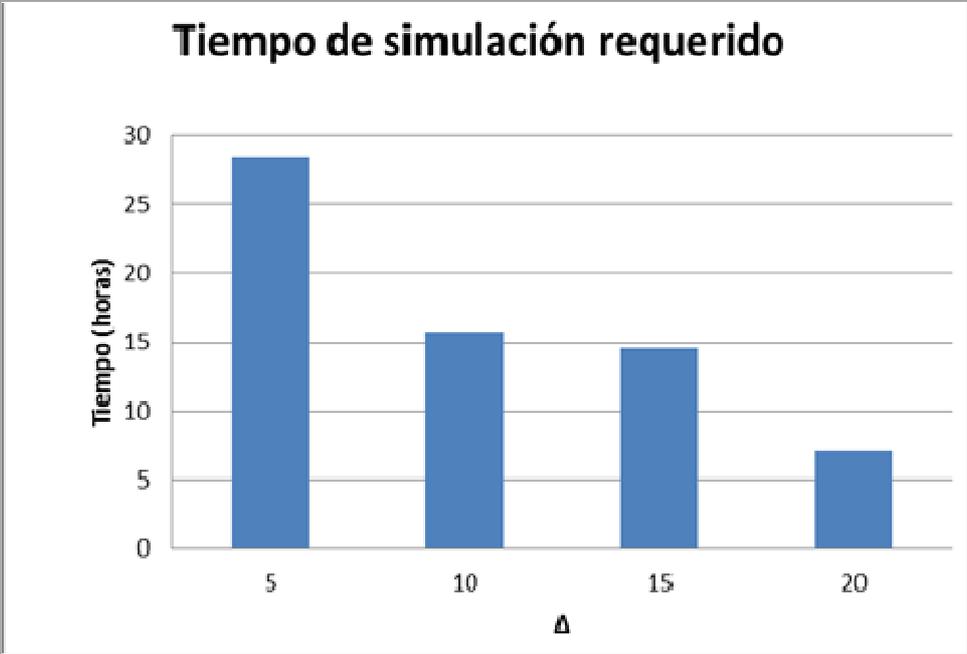


Figura 10. Tiempo de simulación requerido, variando  $\Delta$

En la figura 11 se observa que a mayor tasa de llegada de clientes ( $\lambda$ ), mayor tiempo de simulación es requerido. Esto ocurre debido a que el nivel de servicio generalmente tiende a disminuir cuando la cantidad de clientes por unidad de tiempo aumenta.

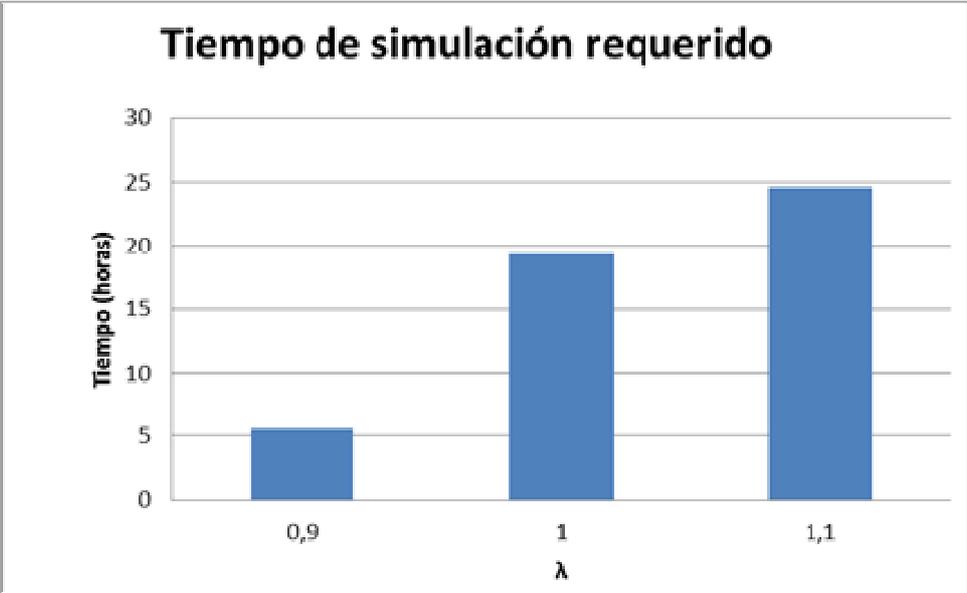


Figura 11. Tiempo de simulación requerido, variando  $\lambda$

Finalmente la figura 12 muestra la evolución del nivel de servicio para  $\lambda=1$ ,  $\lambda_i$  y  $\Phi=99\%$ . Se observa que a menor  $\Delta$ , el número de iteraciones requeridas para encontrar una solución factible es mayor y la variación del mismo es más pequeña, es decir que a  $\Delta$  menores mayor tiempo de simulación será requerido para encontrar la solución factible.

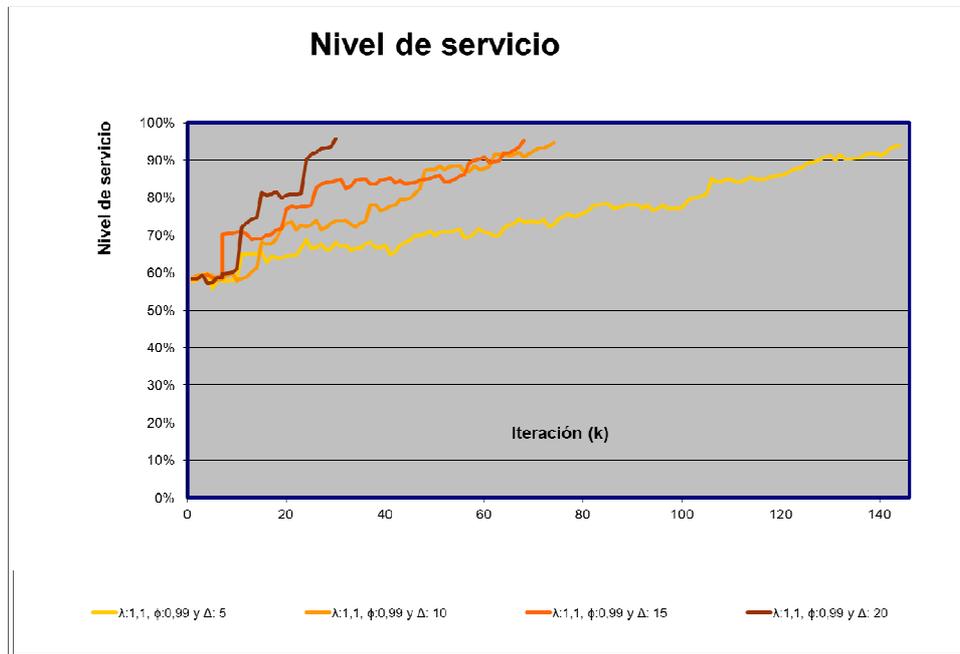


Figura 12. Evolución del nivel de servicio

## Conclusión

Este artículo aborda el problema de asignación de *buffers* intermedios con una capacidad de almacenamiento en sistemas de producción sujetos a fallas y tiempos de proceso aleatorios. Se consideran diferentes costos de mantenimiento de inventario unitarios para cada *buffer*. El objetivo de esta investigación es minimizar el costo promedio por unidad de tiempo mientras el nivel de servicio promedio alcanza un valor predefinido. Se propuso un heurístico para incrementar dinámicamente el tamaño de un *buffer* hasta que el valor de nivel de servicio deseado es alcanzado. Para estimar el costo promedio por unidad de tiempo y el nivel de servicio se propuso un modelo de simulación con eventos discretos, y para evaluar el desempeño del enfoque propuesto, se usó una instancia de prueba de la literatura. El heurístico propuesto resolvió efectivamente la prueba de instancias con varios casos de pruebas con diferentes tamaños y complejidad en orden a entender mejor el desempeño del heurístico bajo diferentes circunstancias.

El tiempo de simulación requerido para encontrar una solución factible es considerable, pero no es un problema debido a que por el objetivo que tiene el modelo, el cual es encontrar las capacidades para cada *buffer*, hacen que la ejecución de la simulación no sea rutinaria; es decir, generalmente una compañía no cambia los tamaños de sus centros de almacenamiento frecuentemente.

## Referencias

- Amiri, M., Mohtashami, A. (2012). Buffer Allocation in Unreliable Production Lines Based on Design of Experiments, Simulation, and Genetic Algorithm. *The International Journal of Advanced Manufacturing Technology*, Vol. 62, No. 1-4, pp. 371-383.
- Bertrand, J.W., Fransoo, J.C. (2002). Operations Management Research Methodologies Using Quantitative Modeling. *International Journal of Operations and Production Management*, Vol. 22, No 2, pp. 241-264.
- Chelbi, A., Ait-Kadi, D. (2004). Analysis of a Production/Inventory System with Randomly Failing Production Unit Submitted to Regular Preventive Maintenance. *European Journal of Operational Research*, Vol. 156, No. 3, pp. 712-718.
- Demir, L., Tunalı, S., Løkketangen, A. (2010). A Tabú Search Approach for Buffer Allocation in Production Lines with Unreliable Machines. *Engineering Optimization*, Vol. 43, No. 2, pp. 213-231.
- Demir, L., Tunalı, S., Eliiyi, D.T. (2012). An Adaptive Tabú Search Approach for Buffer Allocation Problem in Unreliable Non-Homogenous Production Lines. *Computers and Operations Research*, Vol. 39, No. 7, pp. 1477-1486.
- Dolgui, A., Ereemeev, A., Kolokolov, A., Sigaev, V. (2002). A Genetic Algorithm for the Allocation of Buffer Storage Capacities in a Production Line with Unreliable Machines. *Journal of Mathematical Modelling and Algorithms*, Vol. 1, No. 2, pp. 89-104.
- Gharbi, A, Kenne, J., Beit, M. (2007). Optimal Safety Stocks and Preventive Maintenance Periods in Unreliable Manufacturing Systems. *International Journal of Production Economics*, Vol. 107, No. 2, pp. 422-434.
- Massim, Y., Yalaoui, F., Amodeo, L., Chatelet, E., Zebalah, A. (2010). Efficient Combined Immune-Decomposition Algorithm For Optimal Buffer Allocation in Production Lines for Throughput and Profit Maximization. *Computers & Operations Research*, Vol. 37, No. 4, pp. 611-620.
- Nahas, N, Ait-Kadi, D., Nourelfath, M. (2006). A New Approach for Buffer Allocation in Unreliable Production Lines. *International Journal of Production Economics*, Vol. 103, No. 2, pp. 873-881.
- Nourelfath, M., Nahas, N., Ait-Kadi, D. (2005). Optimal Design of Series Production Lines with Unreliable Machines and Finite Buffers. *Journal of Quality in Maintenance Engineering*, Vol. 11, No. 2, pp. 21-138.
- Radhoui, M., Rezg, N., Chelbi, A. (2009). Integrated model of preventive maintenance, quality control and buffer sizing for unreliable and imperfect production systems. *International Journal of Production Research* 47(2): 389–402.
- Rezg, N., Dellagi, S., Chelbi, A. (2008). Joint Optimal Inventory Control and Preventive Maintenance Policy. *International Journal of Production Research*, Vol 46, No 19, pp. 5349-5365.
- Rezg, N., Chelbi, A., Xie, X. (2005). Modeling and Optimizing a Joint Inventory Control and Preventive Maintenance Strategy for a Randomly Failing Production Unit: Analytical and Simulation Approaches. *International Journal of Computer Integrated Manufacturing*, Vol. 18, No. 2, pp. 225-235.

Shi, C., Gershwin, S.B. (2009). An Efficient Buffer Design Algorithm for Production Line Profit Maximization. *International Journal of Production Economics*, Vol. 122, No. 2, pp. 725-740.

Spinellis, D.D., Papadopoulos, C.T. (2000). A Simulated Annealing Approach for Buffer Allocation in Reliable Production Lines. *Annals of Operations Research*, Vol. 93, No. 1, 373-384.

Zequeira, R., Valdes, J.E., Berenguer, C. (2008). Optimal Buffer Inventory and Opportunistic Preventive Maintenance under Random Production Capacity Availability. *International Journal of Production Economics*, Vol. 111, No. 2, pp. 686-696.

Zequeira, R., Prida, B., Valdés, J.E. (2004). Optimal Buffer Inventory and Preventive Maintenance for an Imperfect Production Process. *International Journal of Production Research*, Vol. 42, No. 5, pp. 959-974.