

# COMPARACIÓN DE TRES METAHEURÍSTICAS PARA LA OPTIMIZACIÓN DE INVENTARIOS CON ESTIMACIÓN DE DEMANDA

## COMPARISON OF THREE METAHEURISTICS TO OPTIMIZE INVENTORIES WITH DEMAND ESTIMATION

Laura Margarita Guerrero Guerra<sup>1</sup>, Juan David Gómez Ruiz<sup>2</sup>, Diego León Zapata Ruiz<sup>3</sup>, Marisol Valencia Cárdenas<sup>4,\*</sup>

### RESUMEN

Dentro de las técnicas de solución a problemas complejos de optimización están las heurísticas y las Metaheurísticas. Este trabajo presenta tres Metaheurísticas: Colonia de Hormigas, Algoritmo Genético, y Programación Evolutiva, cuyo objetivo es establecer una comparación de características, ventajas y desventajas para dichas Metaheurísticas. Se hace una revisión de la literatura, así como también se diseñan los tres algoritmos en el software R, y se aplican en una empresa de confecciones, evaluando el comportamiento de los parámetros, tiempo de cálculo y la calidad en las soluciones. Los resultados experimentales muestran que el Algoritmo Colonia de Hormigas obtiene adecuadas soluciones y tiene más rapidez computacional, al compararlo con los otros dos procedimientos diseñados en el software R. Además, con éste fue posible definir la mejor política de inventarios de la empresa objeto de estudio.

**Palabras clave:** Optimización, Metaheurísticas, Modelos de Inventarios, Pronósticos.

### ABSTRACT

Among the techniques for solving complex optimization problems are heuristics and metaheuristics. This work presents three Metaheuristics: Ant Colony, Algorithm Genetic and Evolutionary Programming, which aims to establish a comparison of features, advantages and disadvantages for those Metaheuristics. A review of the literature is made, and also the design of the three algorithms in R program, and they are applied to a confection company, evaluating the performance in terms of its parameters, time calculation and quality solutions. Experimental results show that the Ant Colony algorithm shows adequate solutions, as well as it has more computational speed, when it is compared with the other two procedures designed in software R. In addition, with this, it was possible to define the best policy inventory of the company under study.

**Keywords:** Optimization, Metaheuristics, Inventory Models, Forecasts.

---

<sup>1</sup>Ingeniera Industrial, Universidad Pontificia Bolivariana. Medellín. Colombia. lauramargarita.guerrero@upb.edu.co

<sup>2</sup>Ingeniero Industrial, Universidad Pontificia Bolivariana. Medellín. Colombia. juandagr07@hotmail.com

<sup>3</sup>Docente, Universidad Pontificia Bolivariana. Medellín. Colombia. diegozapata77@gmail.com

<sup>4</sup>Docente de la Institución Universitaria Tecnológico de Antioquia. Medellín. Colombia. Phd. en Ingeniería-Industria y organizaciones. mvalencia@unal.edu.co, solmarival@gmail.com

\*Autor para correspondencia: mvalencia@unal.edu.co

## INTRODUCCIÓN

La optimización es importante en áreas como las matemáticas y la física, pero también a nivel industrial, donde se encuentra la necesidad de optimizar recursos, espacios y operaciones, para el cumplimiento de objetivos estratégicos. En dicha industria funcionan sistemas que pueden modelarse para representar procesos como producción o inventarios.

La optimización es la selección de la mejor alternativa posible. Sus componentes se resumen en: una función objetivo, como medida cuantitativa del funcionamiento del sistema que se desea optimizar, variables de decisión que deberían tomarse de manera óptima para encontrar el mejor valor de la función objetivo; y las restricciones, que representan el conjunto de relaciones, expresadas mediante ecuaciones o inecuaciones que ciertas relaciones de variables están obligadas a satisfacer (Silver, 2004).

En la industria manufacturera, la planificación de los inventarios juega un papel fundamental para programar adecuadamente los niveles de producto terminado, con el fin de balancear los excesos y los faltantes, además, proveer a la empresa una planeación adecuada de los periodos de producción, evitando sobrecostos. Al respecto muchos autores han elaborado modelos cuantitativos para lograr estados de optimalidad (Jianfeng *et al.*, 2011; Kevork, 2010; Schwartz *et al.*, 2006; Vidal *et al.*, 2004). Valencia *et al.*, (2015), han revisado trabajos relacionados con optimización de inventarios, que cada vez con mayor frecuencia involucran variables dinámicas, como la demanda. En éste, se presentan características de estos modelos, como los tipos de costos, funciones objetivo, entre otros aspectos. También indican que algunos de éstos trabajos han usado heurísticas como Tabú y algoritmos genéticos para encontrar una minimización de costos o para optimizar la producción, pero no se muestran comparaciones de diversas heurísticas, aplicadas en este mismo tipo de problemas, o tampoco se ha visto el uso de la distribución Normal Truncada o el diseño algorítmico en el software R, dentro de dichos algoritmos, como se pretende en este trabajo, que recopila parte de los resultados de un proyecto de investigación en el tema de inventarios (Gómez & Guerrero, 2014).

Los modelos de inventarios pueden ser estáticos, basados en promedios o condiciones estables de la demanda, o dinámicos, con condiciones cambiantes en la demanda. En relación a los modelos estáticos, Pasandideh *et al.* (2011), consideran un sistema con capacidad limitada de espacio para almacenamiento, para el cual se desarrolla un algoritmo que permite encontrar las cantidades de pedido y los niveles máximos de retraso, a partir del establecimiento inicial de un modelo de Cantidad Económica de Pedido (EOQ), a diferencia de lo que es propuesto en este trabajo, ya que se considera una dinámica en las cantidades a proveer cada periodo de tiempo. Nenes *et al.* (2010), consideran una demanda intermitente y desigual, en miles de artículos diferentes, suministrados por proveedores distintos, la cual modelan con ayuda de las distribuciones Gamma y Poisson. En este sentido, se quiere probar en este estudio, si una metaheurística que usa muestreo estadístico puede dar soluciones aproximadas pero rápidas para estos sistemas, en especial, sobre una optimización de inventarios en industrias manufactureras y qué tan buenas son al comparar tres algoritmos diferentes.

Silver (2004) define una heurística como un método que, sobre la base de la experiencia o de un juicio, puede producir una solución razonable a un problema, pero que no se puede garantizar que produzca la solución matemáticamente óptima. Entre la heurística y la metaheurística hay una diferencia; *“Una metaheurística es un proceso maestro iterativo que guía y modifica las operaciones de heurísticas subordinadas para producir eficientemente soluciones de alta calidad”* (Silver, 2004), luego, se espera encontrar mejores soluciones con el segundo procedimiento, pues además puede proporcionar alternativas de solución más prácticas, en casos en que los problemas sean complejos de resolver por algoritmos exactos (Winker & Gilli, 2004) the solutions to these simplified problems might not be satisfying. A different approach consists in applying optimization heuristics such as evolutionary algorithms (simulated annealing, threshold accepting. El origen de las metaheurísticas se puede encon-

trar en la inteligencia artificial y en la investigación de operaciones (Blum *et al.*, 2011), pero también permiten combinar procesos de simulación para la búsqueda de mejores soluciones (Valencia *et al.*, 2015). Este trabajo se basa en una comparación de tres técnicas metaheurísticas: Colonia de Hormigas, Algoritmos Genéticos y Programación Evolutiva, realizando un paralelo de manera cualitativa, acorde con las características, ventajas y desventajas, reseñadas en la literatura; segundo se diseñan los tres algoritmos en el programa R, y se aplican a la optimización de un modelo inventarios de una empresa de manufactura del sector de confecciones de Colombia. Se indica al final, cuál es más eficiente para proveer soluciones, agilidad computacional, y se compara con relación al comportamiento si la empresa hubiese planeado acorde con la solución dada por el algoritmo propone de forma óptima.

La tabla 1 muestra la definición, características, ventajas y desventajas que presentan los algoritmos: Colonia de Hormigas, Algoritmo Genético, Programación Evolutiva. Esto permite realizar el comparativo cualitativo de características, ventajas y desventajas, estableciendo un paralelo entre éstas.

Además, entre las aplicaciones comunes a las tres técnicas, está el problema del agente viajero, enrutamiento de vehículos, asignación de tareas, pero no en muchos casos, optimización de inventarios (Urrea & Torres, 2006). De esta revisión, se puede concluir que el algoritmo ACO presenta más organización, maneja memoria, actualización de sus parámetros, y un método que permite construir un camino de soluciones con aparente muy buena convergencia, en comparación con los otros dos, que pueden quedarse en simulaciones sin que se guarde necesariamente memoria de su recorrido, lo cual haría que se reprocesen los mismos puntos una y otra vez, aumentando el tiempo de cómputo.

**Tabla 1.** Comparativo cualitativo sobre las tres metaheurísticas

COLONIA DE HORMIGAS (ACO)	ALGORITMO GENÉTICO (AG)	PROGRAMACIÓN EVOLUTIVA (PE)
<b>DEFINICION</b>		
Diseñada para la solución de complejos problemas de optimización combinatoria (Dorigo & Blum, 2005). Algoritmo inspirado en la forma en que colonias de hormigas buscan alimentos.	Las estrategias de evolución del algoritmo, principalmente el cruce y la mutación, trabajan sobre los individuos, o soluciones al problema, que van evolucionando a través de las generaciones, hasta que el algoritmo finalmente converge al individuo que optimiza en mayor medida la función objetivo del problema (Agustín, 1998).	Como AG, se emula el proceso biológico de adaptación de los organismos vivos al entorno y las condiciones del medio (Das et al. 2013), aplicándolo a la resolución de dichos problemas. Similar a AG, trabaja sobre una población inicial de soluciones candidatas evaluadas dentro de una función de aptitud.
<b>BASADO EN</b>		
La forma como las hormigas buscan el mejor recorrido desde su nido hasta el sitio en el que se encuentra su alimento (Dorigo & Blum, 2005).	Teoría de la evolución de Darwin: mecanismos de selección natural y reproducción sexual (Holland, 1992).	Teoría de la evolución de Darwin: mecanismos de selección natural y reproducción asexual. (Das et al. 2013)
<b>CARACTERISTICAS</b>		
<ul style="list-style-type: none"> <li>• Conjunto finito de componentes de la solución.</li> <li>• Utiliza un modelo probabilístico de transición.</li> <li>• Fabrica soluciones de manera incremental e iterativa.</li> <li>• Retroalimentación positiva entre hormigas.</li> <li>• Actualización local y/o global de feromona (depósito y evaporación).</li> <li>• Las hormigas tienen memoria, no repiten los puntos visitados.</li> <li>• Parámetros controlables: <math>\alpha</math>, <math>\beta</math>, <math>\rho</math>, número de hormigas, avances, feromona inicial.</li> </ul>	<ul style="list-style-type: none"> <li>• Espacio de búsqueda limitado dentro de cierto rango.</li> <li>• Utiliza reglas probabilísticas de selección.</li> <li>• Trabaja con poblaciones de soluciones.</li> <li>• Trabaja con una codificación, generalmente binaria, de las soluciones.</li> <li>• Utiliza una función objetivo o de aptitud.</li> <li>• Implementa mecanismos de selección, cruce y mutación.</li> <li>• La nueva población es de <math>P = \mu</math> hijos, que nacen de <math>\mu</math> padres.</li> <li>• Parámetros controlables: tamaño de población, frecuencia de cruce y de mutación (porcentaje), parámetro de conversión de números reales a binarios.</li> </ul>	<ul style="list-style-type: none"> <li>• Espacio de búsqueda limitado dentro de cierto rango.</li> <li>• Utiliza reglas probabilísticas de selección.</li> <li>• Trabaja con poblaciones de soluciones.</li> <li>• No usa codificación, utiliza los vectores con valores reales directamente como individuos.</li> <li>• Utiliza una función objetivo o de aptitud.</li> <li>• Implementa mecanismos de selección y mutación.</li> <li>• La nueva población es de <math>P = \lambda</math> padres + <math>\mu</math> hijos, que compiten entre sí.</li> <li>• Parámetros controlables: tamaño de población, tasa de mutación.</li> </ul>

<b>VENTAJAS</b>		
<ul style="list-style-type: none"> <li>• Con la retroalimentación positiva, se logra llegar a buenas soluciones rápidamente.</li> <li>• Ninguna ruta se hace dominante por la evaporación, favoreciendo la exploración de otros puntos del espacio de búsqueda (Dorigo &amp; Gambardella, 1997).</li> <li>• Utilizado para resolver problemas de optimización combinatoria estáticos y dinámicos.</li> </ul>	<ul style="list-style-type: none"> <li>• Aplicables a cualquier problema de optimización combinatoria (Agustín, 1998).</li> <li>• Permite una exploración de un amplio abanico de soluciones, por la simultaneidad (Holland, 1992).</li> </ul>	<ul style="list-style-type: none"> <li>• Simplicidad conceptual y amplia aplicabilidad.</li> <li>• Se puede aplicar a problemas combinatorios difíciles (Gomme &amp; Harrald, 1998).</li> <li>• Potencial para incorporar nuevo conocimiento y crear híbridos con otras técnicas.</li> <li>• Robustas a cambios dinámicos.</li> </ul>
<b>DESVENTAJAS</b>		
<ul style="list-style-type: none"> <li>• Definición muy general que puede dificultar su análisis teórico.</li> <li>• Los resultados del algoritmo original de ACO tienden a ser inferiores a los de otros algoritmos, por lo cual se le han introducido diversos cambios y extensiones (Blum, 2005).</li> </ul>	<ul style="list-style-type: none"> <li>• Pueden tardar mucho en converger, o no converger en absoluto, dependiendo de parámetros como tamaño de la población y número de generaciones.</li> <li>• Representación binaria poco práctica para resolver problemas del mundo real (Coello, 2014).</li> </ul>	<ul style="list-style-type: none"> <li>• Algoritmos estocásticos que no siempre garantizan que se encuentre el óptimo global exacto en todos los casos (Sáez <i>et al.</i>, 2005).</li> </ul>

## METODOLOGÍA

Se realiza en primer lugar, un análisis estadístico en el cual se solucionan problemas de atípicos y variabilidad de las ventas de producto terminado de la empresa de confecciones en estudio, y se estima un modelo de regresión, que propone el pronóstico de la demanda de dichas ventas para el año 2013, dado por  $z_t$ , partiendo de la ecuación general dada por (6).

$$z_t = \beta_0 + \dots + \beta_k z_{t-k} + \dots + \beta_i X_i + \varepsilon \tag{6}$$

Varios modelos son evaluados, con la ayuda del software R (R Core Team, 2014), de acuerdo con la significancia de sus variables, el cumplimiento de los supuestos de los residuos: incorrelación, Distribución Normal y varianza constante; y se evalúa el indicador MAPE (Error Medio Absoluto Porcentual), que se espera sea muy bajo. Utilizando la ecuación de pronósticos final, se procede a calcular los pronósticos de las demandas mensuales de la referencia en cuestión, para el año 2013. Con base en éstos, se inicia el proceso de optimización del modelo de inventarios expresado en las ecuaciones (7) a (14), usando las metaheurísticas explicadas a continuación.

### Diseño de Metaheurísticas

A continuación, se muestra el procedimiento general para cada una de las metaheurísticas: Algoritmo Colonia de Hormigas (ACO), Algoritmo genético (GA) y Algoritmo de Programación Evolutiva (EP), aplicadas a la optimización del modelo de inventarios, cuyos parámetros y variable de decisión se definen así:

Datos:

$t = 1, 2, 3, 4, \dots, N$  (N es el total de períodos de estudio). En el caso de estudio,  $N=10$

$C_t$  = Costo de producción unitario

$H_t$  = Costo de almacenamiento unitario (por período)

$F_t$  = Costo por faltante unitario (por período)

$D_t$  = Demanda en el período t

$I_{t-1}$  = Inventario final del período anterior

Variables:

$X_t$  = unidades a producir en el período t, variables de decisión.

$I_t$  = Inventario final en el período t.

$S_t$  = Número de faltantes en el período t.

Función Objetivo:

$$\text{Min } \sum_{t=1}^N [C_t(X_t)] + \sum_{t=1}^N [h_t(I_t)] + \sum_{t=1}^N [F_t(S_t)] \quad (7)$$

Sujeto a:

1. Balance de inventarios

Sea  $I_t = X_t + I_{t-1} - D_t$  (8)

Si  $I_t < 0$  entonces  $S_t = -I_t$  (9)

$$I_t = 0 \quad (10)$$

Sino  $I_t = I_t$  (11)

$$S_t = 0 \quad (12)$$

2. Nivel de faltantes

$$S_t \leq 0,10 * D_t \quad (13)$$

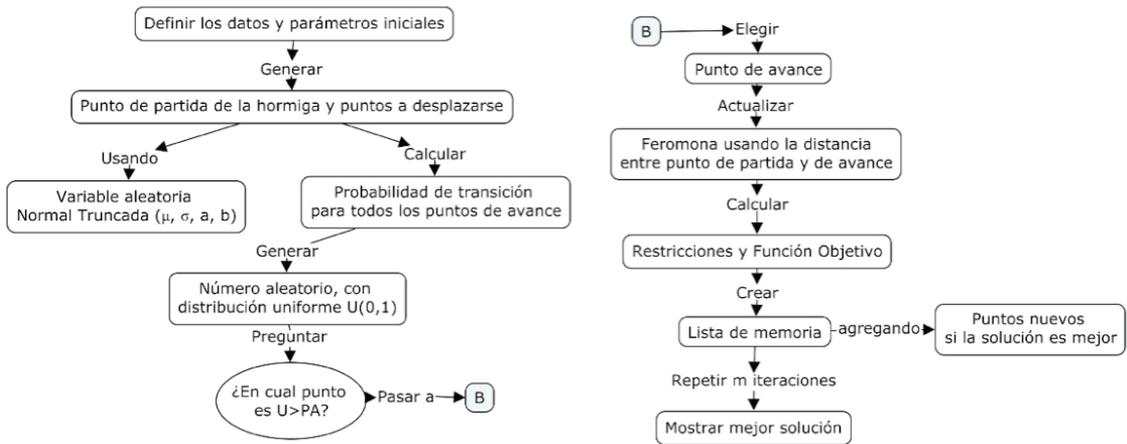
3. Límite de producción

$$X_i \leq \sum_{t=i}^N D_t \quad (14)$$

En (13), se busca fijar un porcentaje para establecer que los faltantes no pueden exceder el 10% de la demanda, para ser almacenado, y con ello, proveer un mejor cubrimiento. Este problema, junto con los pronósticos de demanda de 2013, son ingresados en la herramienta R. Los pasos comprendidos por cada uno de estos algoritmos se explican a continuación:

### Colonia de Hormigas (ACO)

Los pasos del procedimiento algorítmico se resumen en la figura 1 y se detallan en la explicación por pasos posterior.



**Figura 1.** Diagrama de flujo del algoritmo ACO

La descripción de cada uno de los pasos para el algoritmo ACO, diseñado en el programa R, son:

**Paso1.** Se definen los parámetros y datos iniciales, así:

- a. Inventario inicial ( $I_0$ )
- b. Costo unitario de producción ( $C_t$ )
- c. Costo unitario de mantener ( $H_t$ )
- d. Costo unitario por faltante ( $C_{fal}$ )
- e. Valor inicial de la feromona ( $fer$ )
- f.  $\alpha$  (importancia de la traza de feromona)
- g.  $\beta$  (importancia de la visibilidad)
- h.  $\rho$  (tasa de evaporación)
- i. Número avances que puede realizar la hormiga
- j. Vector de demanda pronosticada,  $D_t (1 \times 10)$ : 10 valores de demanda para el año 2013.

**Paso 2.** Generación del punto de partida de la hormiga, y de los posibles puntos hacia los cuales se puede desplazar. Cada punto comprende un vector de 10 cantidades a producir, una para cada mes del año (hasta octubre para el caso de estudio). Para generarlas, se utiliza una variable aleatoria normal truncada con las siguientes características:

- $X_t \geq 0,90 * D_t$ : Valor mínimo igual al 90% del valor de la demanda pronosticada para cada mes
- $X_i \leq \sum_{t=i}^N D_t$ : Valor máximo igual a la suma de las demandas mensuales pronosticadas del año, a partir del mes para el cual se determina la producción.
- $\mu = D_t$ : Media de la distribución de probabilidad Normal Truncada, igual a la demanda mensual pronosticada.

- $\sigma = 1$ : Desviación estándar igual a uno.

La variable de decisión  $X_t$ , mensual es el promedio de una muestra de 100 datos. Con el objeto de identificar variaciones en los resultados generados por el algoritmo, se plantean 4 escenarios de acuerdo con el valor de la desviación estándar de la distribución Normal Truncada ( $\mu, \sigma, a, b$ ), así: 1, 5, 10, 15. Escenarios que se consideraran para la exploración en las tres técnicas heurísticas, de tal forma que permitan establecer comparaciones de los resultados.

**Paso 3.** Cálculo de la probabilidad de transición para todos los puntos de avance:  $X_t$  del espacio de búsqueda, así:  $P(C_i|S) = \frac{[\tau_i]^\alpha \cdot [\eta(C_i)]^\beta}{\sum_{C_j \in N(S)} [\tau_j]^\alpha \cdot [\eta(C_j)]^\beta}$  (Dorigo & Blum, 2005) Para ello, se utiliza el valor de la feromona y las distancias entre cada uno de ellos y el punto de partida. Seguido, se calculan las probabilidades acumuladas ACO para todos los puntos de avance de la variable  $X_t$ .

**Paso 4.** Generación de un número aleatorio, con distribución uniforme entre 0 y 1.

**Paso 5.** Selección del punto hacia el cual se va a desplazar la hormiga, si el número aleatorio es inmediatamente mayor a la probabilidad acumulada, se elige el punto de avance correspondiente a esta probabilidad.

**Paso 6.** Actualización de la feromona, utilizando el valor de la distancia entre el punto de partida y el punto de avance elegido con:  $\tau_i \leftarrow (1 - \rho) \cdot \tau_i + \rho \cdot \sum_{\{s \in S_{upda} | C_i \in s\}} w_s \cdot F(s)$ .

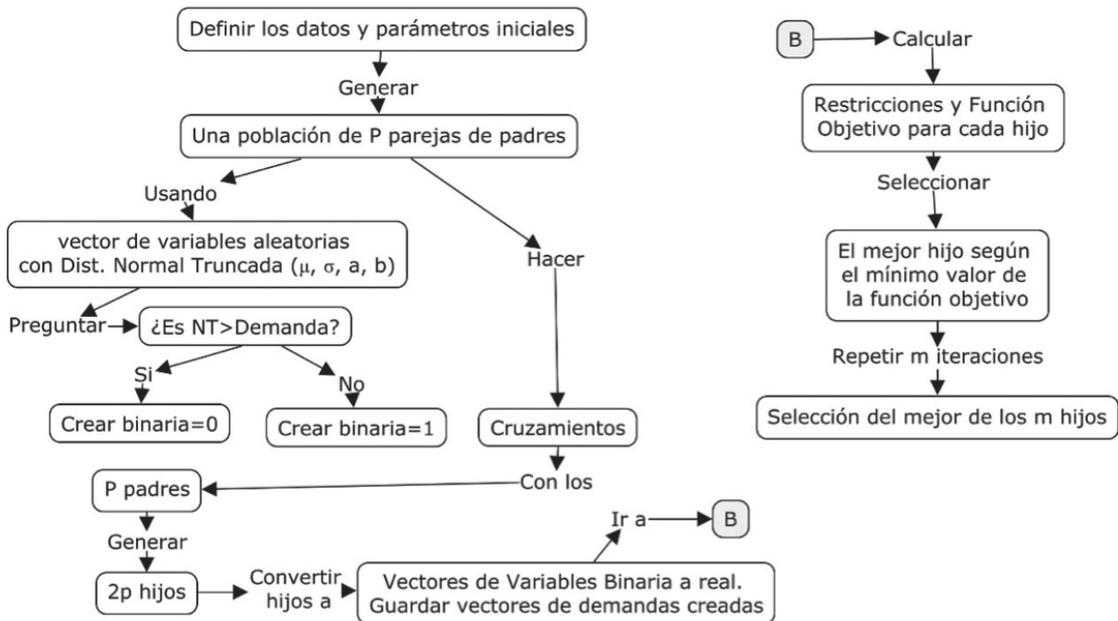
**Paso 8.** Cálculo del costo total, inventarios finales y faltantes, la función objetivo, dados por las ecuaciones (7) a (14).

**Paso 9.** Creación de una lista de memoria, a partir de la mejor solución encontrada por el software: se repite el proceso anterior para generar una nueva solución y, si ésta es mejor que la que se tenía, se incluye en la lista; de lo contrario, se desecha. Así, se van generando nuevas soluciones que sólo son almacenadas si generan una mejor solución que las que se encuentran en la lista. Esta búsqueda se asemeja al proceso Tabú propuesto en (Valencia et al., 2014); y se repite durante m iteraciones.

**Paso 10.** Al finalizar la búsqueda Tabú, el programa muestra la solución que más optimiza la función objetivo, la minimización de costos.

### **Algoritmo Genético (GA)**

Los pasos del procedimiento algorítmico se resumen en la figura 2 y se detallan en la explicación por pasos posterior.



**Figura 2.** Diagrama de Flujo del Algoritmo Genético

Los pasos para el algoritmo genético, diseñado en el programa R, son:

**Paso 1.** Se definen los parámetros del a. al d. del algoritmo ACO, y además, se lee la demanda pronosticada.

**Paso 2.** Generación de una población inicial de P parejas de padres, donde cada padre está conformado por un vector de n cantidades a producir (10 en el caso de estudio). Para su generación, se utiliza la misma variable aleatoria normal truncada del algoritmo ACO. Luego, estas cantidades a producir son convertidas a números binarios, dependiendo de la condición: ser menor (se cambian por ceros); o mayor (se cambian por unos) a la demanda pronosticada para el mes correspondiente.

**Paso 3.** Cruzamiento de P parejas de padres para generar P parejas de hijos. Para cada pareja de padres (padre 1 y padre 2), se generan n probabilidades equitativas (10, en el caso de estudio) y se calculan sus probabilidades acumuladas. A continuación, se genera un número aleatorio con distribución uniforme entre 0 y 1, y se contabiliza el número de probabilidades acumuladas que están por debajo de éste, lo cual permite determinar el punto de cruce entre los componentes que cada padre transmite a los hijos.

**Paso 4.** Generación de P parejas de hijos, de acuerdo con el punto de cruce definido anteriormente. Si (n) es el número de probabilidades acumuladas por debajo del número aleatorio, entonces (n) es el número de componentes que el hijo 1 hereda del padre 1, de izquierda a derecha, y (10 – n) los que hereda del padre 2. Para el hijo 2, sucede lo contrario, hereda (10 – n) del padre 1 y (n) del padre 2. Así sucede con cada pareja de padres y su respectiva pareja de hijos.

**Paso 5.** Conversión de hijos a cantidades a producir. Los componentes de cada hijo se cambian a valores reales utilizando la misma variable aleatoria normal truncada del numeral b, con la siguiente condición: para aquéllos que son cero, los 100 números generados se deben encontrar entre el 90% de la demanda pronosticada y el 100% de ésta, mientras que, para aquéllos que son uno, el valor mínimo es el 100% de la demanda y el máximo es la suma de las demandas restantes del año.

**Paso 6.** Calcular inventarios finales y los faltantes, por mes, a partir de las restricciones de balance de inventarios; con ello, calcular el costo total, sumando el costo de unidades a producir, el de unidades de inventario final y faltantes, correspondientes a cada hijo, en la función objetivo.

**Paso 7.** Selección del mejor hijo, como aquél que minimiza la función objetivo, o genera el menor costo total.

**Paso 8.** Generación de m mejores hijos, al repetir los pasos anteriores durante m iteraciones.

**Paso 9.** Evaluación de los m mejores hijos en la función objetivo y selección del mejor hijo, que constituye la primera solución al problema de minimización de costos.

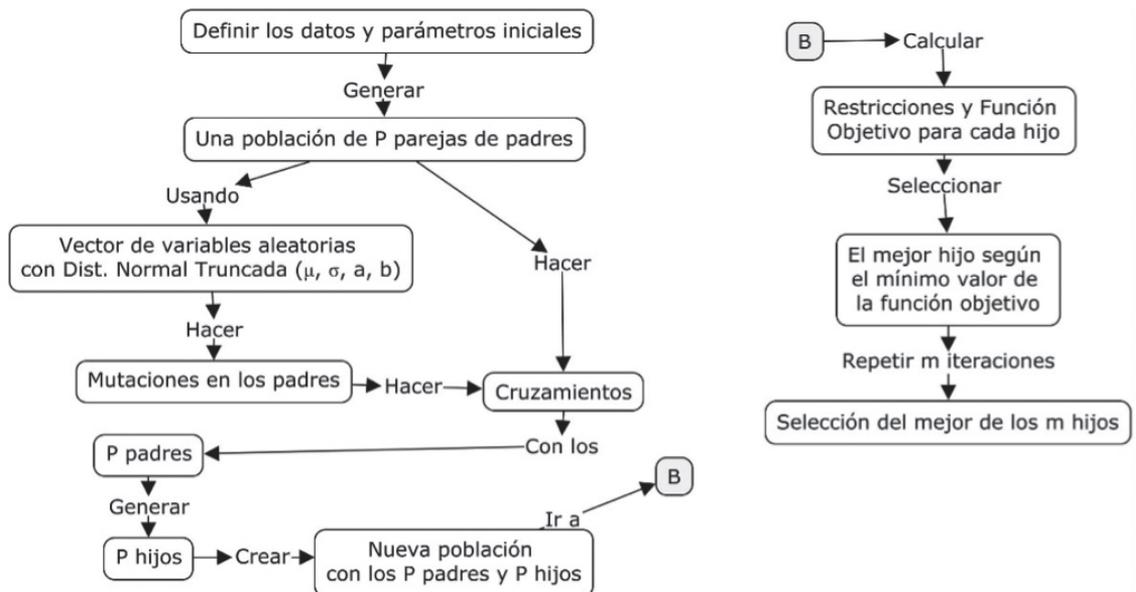
**Paso 10.** Mutación de los mejores hijos, cambiando los componentes que tienen en ceros por unos, y viceversa.

**Paso 11.** Evaluación de hijos mutados en la función objetivo y selección del mejor hijo mutado, que constituye la segunda solución al problema.

**Paso 12.** Comparación de la solución 1 y la solución 2. La mejor es la solución final al problema de minimización.

### Programación Evolutiva (EP)

A diferencia del algoritmo anterior, en este no se consideran variables binarias, se adiciona mutación sobre los padres, y la nueva población de padres se conforma por padres e hijos. Los pasos del procedimiento algorítmico se resumen en la figura 3 y se detallan en la explicación por pasos posterior.



**Figura 3.** Diagrama de Flujo del Algoritmo de Programación Evolutiva

Los pasos para éste algoritmo, diseñado en el programa R, son:

**Paso 1.** Se definen los parámetros del a. al d. del algoritmo ACO, y además, la demanda pronosticada.

**Paso 2.** Generación de una población inicial de  $P$  número de padres, donde cada padre está conformado por un vector de  $n$  posibles cantidades a producir (10 en el caso de estudio), obtenidas a partir de la misma variable aleatoria normal truncada usada para Colonia de Hormigas y Algoritmo Genético.

**Paso 3.** Mutación de  $P$  número de padres para generar  $P$  número de hijos. Cada padre es mutado al sumarle la cantidad correspondiente al coseno de un número generado aleatoriamente, de manera que se obtiene un hijo completamente diferente a su padre.

**Paso 4.** Definición de la nueva población, compuesta por ambas generaciones. Padres e hijos pasan a formar parte de la población cuyo desempeño va a ser evaluado, con el objeto de determinar quién genera el menor costo.

**Paso 5.** Calcular inventarios finales y de los faltantes, por mes, a partir de las restricciones de balance de inventarios y de las cantidades a producir  $X_i$  correspondientes a cada individuo; con ello, calcular del costo total generado por cada individuo, al evaluar el costo de unidades a producir, del inventario final y de faltantes, en la función objetivo.

**Paso 6.** Selección del mejor individuo, aquél que minimiza la función objetivo, o genera el menor costo total.

**Paso 7.** Generación de  $m$  mejores individuos, al repetir los pasos anteriores durante  $m$  iteraciones.

**Paso 8.** Evaluación de los  $m$  mejores individuos en la función objetivo y selección del mejor de ellos, que constituye la primera solución al problema de minimización de costos.

**Paso 9.** Creación de una lista de mejores soluciones, encabezada por la primera solución (determinada en el paso anterior): se repiten los pasos para generar un nuevo mejor individuo. Éste se compara con la primera solución al problema de minimización y, si genera un menor costo, entonces es incluido en la lista. De lo contrario, se desecha. El proceso se repite hasta completar una lista de  $m$  soluciones.

**Paso 10.** Desempeño de los individuos incluidos en la lista. Todas estas soluciones son evaluadas en la función objetivo, para elegir a la que genera el menor costo, la cual se convierte en la segunda solución para el problema de minimización.

**Paso 11.** Comparación de la solución 1 y la solución 2. Aquélla que minimiza la función objetivo en mayor medida constituye la solución final al problema de minimización de costos de inventario.

Algunos autores no crean mutaciones en todos los individuos, a diferencia de lo que se crea aquí, donde se mutan todos los individuos de cada población.

## RESULTADOS

A partir de lo expuesto en la metodología, se procede a aplicar, en primera instancia, un análisis tentativo de la serie correspondiente a las ventas de la referencia estudiada, desde enero de 2007 hasta diciembre de 2012, con el objeto de identificar las características de los datos que se pretenden modelar y generar un modelo de pronósticos adecuado. Este análisis comprende inicialmente la detección de outliers y el análisis descriptivo de la serie.

### Generación del Modelo de Pronósticos

Teniendo en cuenta transformaciones en la variable endógena Ventas, tendencias lineales y cuadráticas, un rezago de orden 1, variables ficticias y funciones trigonométricas para la

modelación de la variación estacional, se exploraron diferentes modelos y se encontró el que se presenta en la tabla de Análisis de Varianza o ANOVA (tabla 1), observando, que todas las variables incluidas en el modelo fueron significativas ya que el valor P es menor al nivel 5%.

**Tabla 1.** Tabla de Análisis de Varianza (ANOVA).

	Sum Sq	Df	F value	Pr(>F)
(Intercept)	0,2624	1	2,6872	0,10720
t2	0,6978	1	7,1471	0,01001
L(serie, 1)	0,4367	1	4,4730	0,03925
Nivel	3,9094	2	20,0197	3,571e-07
T	0,0716	1	0,7330	0,39583
Mes	2,2324	11	2,0785	0,03883
Nivel: t	0,9988	2	5,1148	0,00938
Residuals	5,0772	52		

Además, se cumplen los supuestos sobre la distribución normal, varianza constante e incorrelación en los residuos, y se obtiene el menor MAPE de todos los modelos analizados durante la exploración, correspondiente al 21,01%.

De acuerdo con lo anterior, y utilizando los parámetros estimados a través del software R, el modelo de predicción final para la referencia estudiada es la siguiente:

$$\hat{Z}_t = 0,508 + 0,066*t2 + 0,166* Z_{t-1} + 2,139*In_2 + 2,867*In_3 - 0,1267*t - 0,256* Im_8 - 0,092 * Im_{12} - 0,3758*Im_1 - 0,317*Im_2 - 0,144*Im_7 - 0,318* Im_6 - 0,067* Im_3 - 0,413* Im_5 - 0,617* Im_{11} - 0,343* Im_{10} - 0,558*Im_9 - 0,345In_{2:t} - 0,362In_{3:t}$$

Dónde:

$\hat{Z}_t$ : log. Natural  $\hat{Y}$  (siendo  $\hat{Y}$  las ventas o demanda pronosticada)

$Z_{t-1}$ : retardo de orden 1 en el log. Natural

$I_{n2}, I_{n3}$ : indicadoras de los niveles de ventas medio (2) y alto (3).

$I_{m1}, I_{m2}, I_{m3}, I_{m5}, I_{m6}, I_{m7}, I_{m8}, I_{m9}, I_{m10}, I_{m11}, I_{m12}$ : Indicadoras de los meses desde enero hasta diciembre, excluyendo abril.

$I_{n2:t}, I_{n3:t}$ : interacciones nivel 2: tiempo y nivel 3: tiempo.

Con esta ecuación final, se procede a calcular los pronósticos de las ventas mensuales para el año 2013, que son entrada para el proceso de optimización de inventarios.

### Optimización de inventarios a partir de las técnicas heurísticas seleccionadas

Utilizando los pronósticos de las demandas mensuales para 2013, se realiza la minimización de los costos de inventario a través de las tres técnicas seleccionadas. Para cada una de ellas, se definen los parámetros y datos vistos en la tabla 2. Los costos se presentan en pesos colombianos (COL \$)

**Tabla 2.** Parámetros iniciales de Colonia de Hormigas.

Inventario Inicial	25
Costo unitario de producción (Ct)-(COL \$)	82.490,00
Costo unitario de mantener (Ht) (COL \$)	19.787,18
Costo unitario por faltante (Cfal) (COL \$)	707.510,00
Feromona Inicial (ACO)	1,2
$\alpha$ (ACO)	1
$\beta$ (ACO)	0,5
$\rho$ (ACO)	0,5
Avances (ACO)	10

La información relacionada con el inventario inicial y los costos de producción, mantenimiento y por faltantes, fue suministrada por la empresa. Para el algoritmo ACO, se tuvieron en cuenta valores de parámetros que han sido usados previamente en la literatura. Finalmente, los puntos de avance de ACO, se asignó un valor de diez con el objeto de generar 10 oportunidades de desplazamiento para la hormiga, en busca de mejores soluciones.

Se propone comparar resultados generados por los tres algoritmos, en términos de costos, porcentaje de cumplimiento de la demanda y el tiempo de ejecución, partiendo del pronóstico de ventas de 2013. Con el fin de validar la eficacia de los resultados de cada algoritmo, el inventario final real se recalculó a partir de las ventas reales del 2013. Esto permite determinar tanto el servicio, como los costos reales finales si se produjeran estas cantidades propuestas.

### Comparativo de las metaheurísticas

#### Colonia de Hormigas

En cada una de las técnicas se evaluaron cuatro escenarios, donde sólo se modificó la desviación estándar de la distribución normal truncada usada para la búsqueda aleatoria de las variables de decisión. En la tabla 3 se muestran los resultados de todos los escenarios.

**Tabla 3.** Comparación de los escenarios 1, 2, 3, 4 de inventario final de ACO.

ESCENARIO	COSTO TOTAL (COL\$)	CUMPLIMIENTO (%)
1 (desviación = 1)	25.523.299,00	100
2 (desviación = 5)	29.432.178,00	100
3 (desviación = 10)	36.415.275,00	100
4 (desviación = 15)	43.150.901,00	100

Todos los escenarios generan inventario final cada mes, cumpliendo con la demanda mensual, por lo tanto, la demanda total en un 100%. Sin embargo, los inventarios finales de los escenarios 2, 3 y 4, tienen una tendencia creciente a lo largo del tiempo, lo cual implica mayores costos. El escenario 1, en cambio, mantiene unos niveles de inventario estable y presenta el menor costo total de todos, por lo cual se elige como el mejor escenario, de la técnica ACO.

## Algoritmo Genético

**Tabla 4.** Comparación de los escenarios 1, 2, 3, 4 de inventario final de AG.

ESCENARIO	COSTO TOTAL (COL\$)	CUMPLIMIENTO (%)
1 (desviación = 1)	24.438.084,00	100
2 (desviación = 5)	22.947.103,00	100
3 (desviación = 10)	22.947.103,00	100
4 (desviación = 15)	22.828.380,00	100

En el caso de Algoritmo Genético, se observa que todos los escenarios presentan un 100% de cumplimiento de la demanda y que sus inventarios finales tienden a disminuir a lo largo del tiempo. Tomando como referencia el costo total generado, el mejor escenario es el 4, aunque cabe resaltar que sus resultados son muy similares a los de los escenarios 2 y 3.

## Programación Evolutiva

La tabla 5 muestra los escenarios al variar las desviaciones estándar de la distribución normal truncada usada para la búsqueda de la variable de decisión.

**Tabla 5.** Comparación de los escenarios 1, 2, 3, 4 de inventario final de PE.

ESCENARIO	COSTO TOTAL (COL\$)	CUMPLIMIENTO (%)
1 (desviación = 1)	23.593.660,00	100
2 (desviación = 5)	27.842.260,00	100
3 (desviación = 10)	34.363.570,00	100
4 (desviación = 15)	41.099.200,00	100

Los escenarios de Programación Evolutiva muestran un comportamiento similar a los de Colonia de Hormigas ya que, a medida que se incrementa la desviación estándar, el costo total de inventarios va aumentando también. De acuerdo con esto, el escenario 1 es el que mantiene menores niveles de inventario a través del tiempo, presentando así el menor costo total.

**Tabla 6.** Comparación de los mejores escenarios ACO – AG – PE.

MÉTODO	COSTO TOTAL (COL\$)	CUMPLIMIENTO (%)	TIEMPO EJECUCIÓN
ACO	25.523.299,00	100	2 s
AG	22.828.380,00	100	34 s
PE	23.593.660,00	100	40 s

Finalmente, se comparan los resultados de los mejores escenarios de cada una de las técnicas. Todos logran cumplir con un nivel de servicio del 100%. Sin embargo, Algoritmo Genético es el que logra generar el menor costo total, siendo éste de COL\$ 22.828.380, contrario a Colonia de Hormigas, cuyo costo total es de COL\$ 25.523.299,00. Este hallazgo

puede deberse a que el uso de la variable binaria puede no ser afectado significativamente por los cambios de varianza en la búsqueda de soluciones, y se queda en un mismo espacio, mientras que los otros sí, lo cual en otros problemas podría no ser lo más conveniente.

En este caso, dado que el tiempo de ejecución no es el menor, pero que sigue siendo considerablemente pequeño y que logra minimizar el costo total de inventarios en mayor medida, el Algoritmo Genético sería la mejor opción para la optimización de los costos de inventarios, en las condiciones establecidas.

### Validación del problema real

Aquí se busca validar cuantitativamente los resultados de la optimización realizada con las tres técnicas, usando la demanda real de 2013, determinando cuáles son los costos reales generados por cada algoritmo, al comparar las cantidades propuestas a producir con respecto a las cantidades reales demandadas.

La empresa proporcionó las ventas, los inventarios finales reales y la producción real durante 2013, hasta el mes de octubre, a partir de los cuales se calculan los costos totales mensuales de sus inventarios. La tabla 7 muestra los valores encontrados si se produce con las cantidades simuladas.

El costo total real, generado por la gestión de inventarios para la empresa durante 2013 (hasta octubre), fue de \$37.254.842,26, con un porcentaje de cumplimiento de la demanda del 100% (tabla 10), en la cual se observa que durante todos los meses se generó inventario final en lugar de faltantes. El que más disminuye dicho valor real es el escenario 5, con desviación 11, cuyo porcentaje de cumplimiento es del 100% y su costo total es de \$33.335.157,12, menor que el costo del escenario 3. Éste se convierte, finalmente, en el mejor escenario generado por Colonia de Hormigas.

**Tabla 7.** Escenarios 1, 2, 3, 4, 5 y el real de inventario final de ACO.

ESCENARIO	COSTO TOTAL (COL\$)	CUMPLIMIENTO (%)
1 (desviación = 1)	\$ 64.944.499,58	83,04
2 (desviación = 5)	\$ 52.765.522,94	89,29
3 (desviación = 10)	\$ 33.798.538,40	99,70
4 (desviación = 15)	\$ 39.806.867,48	100
5 (desviación = 11)	\$ 33.335.157,12	100
Real 2013	\$ 37.254.842,26	100

Así mismo se procede para las otras dos metaheurísticas, y se encuentran resultados similares al de ACO, pero ninguno menor (Tabla 8). Para el Algoritmo Genético, el mejor escenario fue definir la media con un 25% mayor a la demanda mensual pronosticada y con desviación estándar = 1, para la distribución normal truncada, generando un costo total de \$33.414.305,84, y un 100% de cumplimiento de la demanda total real. Para Programación Evolutiva, el mejor escenario se encuentra al definir una desviación estándar 12, cumpliendo la demanda en un 100%, generando cero faltantes en cada mes y presentando un costo total de \$33.935.455,08, menor al costo total real de 2013.

**Tabla 8.** Comparación final de los mejores escenarios ACO – AG – PE.

MÉTODO	COSTO TOTAL (COL\$)	CUMPLIMIENTO (%)	TIEMPO EJECUCIÓN
ACO	\$ 33.335.157,12	100	2 s
AG	\$ 33.414.305,84	100	34 s
PE	\$ 33.935.455,08	100	40 s

Según los resultados de las 3 metaheurísticas, vistos en la tabla 3, se generaron costos totales similares, pero menores al costo real generado en 2013, correspondiente a \$37.254.842,26, y con un 100% de cumplimiento de la demanda real. Sin embargo, el menor costo total se presentó con la técnica Colonia de Hormigas, generando un ahorro de \$3.919.685,14, correspondiente al 10,38% del costo total real de los inventarios durante 2013. Además, ACO fue el algoritmo con menor tiempo de ejecución, siendo éste igual al 5,88% del tiempo de ejecución de Algoritmo Genético y al 5% del tiempo de ejecución de Programación Evolutiva.

## DISCUSION

Los parámetros elegidos para el algoritmo ACO han sido producto de una revisión de diversas investigaciones, en especial, de Dorigo and Blum, (2005) y de Blum (2005), las cuales resaltan el proceso constructivo de dicha técnica para encontrar buenas soluciones, pero no realizan una comparación con respecto a Algoritmos Genéticos (AG) y Programación Evolutiva (PE), como se establece aquí. Otras investigaciones como la de Silver (2004) revisa características generales de heurísticas, pero no propone el mismo comparativo realizado en este trabajo, y ninguno de estos lo enfoca en la planeación de una política de inventarios usando éstas metaheurísticas.

El algoritmo de Colonia de Hormigas (ACO) mostró tener la organización, parámetros y estrategias apropiadas para realizar una búsqueda que converge en una solución cercana a la óptima, obteniendo la mejor política de inventarios para el caso de estudio, en relación a lo que ocurrió en el escenario real, y a los otros dos algoritmos comparados: Algoritmo Genético (AG) y Programación Evolutiva (PE). La propuesta de optimización con metaheurísticas constituye una innovación para la empresa, que sólo consideraba una predicción de demandas en su planeación, lo cual representa para ésta, un mejoramiento de sus procesos.

De la revisión realizada, también se aprecian campos inexplorados sobre las aplicaciones de distribuciones probabilísticas que difieran de las usadas por ACO; AG o PE, como quizá, las bayesianas.

## CONCLUSIONES

El comparativo permitió encontrar similitudes entre las metaheurísticas, como el uso de distribuciones de probabilidad en los tres algoritmos, así como aproximaciones a una solución aproximada. Se encontraron también diferencias entre éstas, como la forma de orientar la solución, pues Algoritmos Genéticos y Programación Evolutiva muestran poblaciones grandes de valores de soluciones, explorando más opciones, mientras que ACO realiza un direccionamiento de la búsqueda de puntos de las técnicas estudiadas. También se encuentra que la programación en el programa R tiene ventajas en cuanto a su flexibilidad, permitiendo más modificaciones a los algoritmos diseñados en este trabajo, además, los pasos que se han presentado, pueden reproducirse también en otros programas.

El Algoritmo Genético diseñado en el programa R, logró disminuir en mayor medida los costos totales de inventarios, proporcionando órdenes de producción, y menores cantidades

de inventarios finales que las demás técnicas, con un costo inferior de los demás costos generados con las otras dos metaheurísticas. A pesar de que AG usa una representación binaria y tiene mecanismos de cruce y de mutación, contrario a PE, que trabaja directamente con vectores de valores reales y no utiliza el operador cruce, AG presentó un tiempo de ejecución menor a PE.

En el caso de los resultados arrojados por los algoritmos en relación con la demanda real del año 2013, se concluye que todos logran generar costos totales menores al costo real de los inventarios para Industrias Pétalo. Sin embargo, la metaheurística ACO arroja los mejores resultados, presentando un costo total menor del obtenido con AG y PE; además, genera un ahorro del 10,38% del costo real de la empresa hasta el mes de octubre de 2013, con un servicio del 100%.

## REFERENCIAS

AGUSTÍN, José Luis. *Aplicación De Algoritmos Genéticos Al Diseño Óptimo De Sistemas De Distribución De Energía Eléctrica*. Tesis Doctoral. Universidad de Zaragoza, 1998.

BLUM, Christian., 2005. Ant colony optimization: Introduction and recent trends. In : *Physics of Life Reviews*, 2005, vol. 2, no. 4, p. 353-373.

BLUM, Christian., PUCHINGER, Jakob., RAIDL, Günther R., and ROLI, Andrea. Hybrid metaheuristics in combinatorial optimization: A survey. *Applied Soft Computing*, 2011, Vol. 11, no 6, p. 4135-4151.

COELLO, Carlos. *Introducción a la computación evolutiva*. Centro de Investigación y de Estudios Avanzados del I.P.N., (notas de curso), 2014. [citado: abril 2013]. [Disponible en]: <https://www.cs.cinvestav.mx/~EVOCINV/tutorials/computacionevolutiva..htm#com>

DAS, Swagatam., MALLIPEDDI, Rammohan., and MAITY, Dipankar. Adaptive evolutionary programming with p-best mutation strategy. *Swarm and Evolutionary Computation*, 2013, vol. 9, p. 58-68.

DORIGO, Marco., and BLUM, Christian. Ant colony optimization theory: A survey. *Theoretical Computer Science*, 2005, vol. 344, no. 2, p. 243-278.

DORIGO, Marco., and GAMBARDELLA, Luca Maria. Ant colonies for the travelling salesman problem. *Biosystems*, 1997, vol. 43, no. 2, p. 73-81.

GOMME, Paul., and HARRALD, Paul G. Applying evolutionary programming to selected set partitioning problems. *Fuzzy sets and systems*, 1998, vol. 95, no. 1, p. 67-76.

GOMEZ, Juan., and GUERRERO, Laura. Exploración de tres técnicas heurísticas para la optimización de inventario caso de estudio: Empresa Antioqueña. Directores: ZAPATA, Diego., VALENCIA, Marisol. Trabajo de Grado, Universidad Pontificia Bolivariana, 2014. <<http://editio.upb.edu.co/janium-bin/detalle.pl?Id=20160811092146.>>

HOLLAND, John. Algoritmos Genéticos. *Investigación y Ciencia*, 1992, vol.192, p. 38-45.

JIANFENG, Huang., JINGYING, Zhao., and XIAODONG, Wu. Research on the Optimization Strategy of Maintenance Spare Parts Inventory Management for Petrochemical Vehicle. *International Conference on Information Management, Innovation Management and Industrial Engineering*. IEEE, 2011, p. 45-48.

KEVORK, Ilias S., Estimating the optimal order quantity and the maximum expected profit for single-period inventory decisions. *Omega*, 2010. vol. 38, no. 3, p. 218-227.

NENES, George., PANAGIOTIDOU, Sofia., and TAGARAS, George. Inventory management of multiple items with irregular demand: A case study. *European Journal of Operational Research*, 2010, vol. 205, no. 2, p. 313-324.

PASANDIDEH, Seyed Hamid Reza., NIAKI, Seyed Taghi Akhavan., and NIA, Ali Roozbeh. A genetic algorithm for vendor managed inventory control system of multi-product multi-constraint economic order quantity model. *Expert Systems with Applications*, 2011, vol. 38, no. 3, p. 2708-2716.

R CORE TEAM. *A Language and Environment for Statistical Computing*. Vienna, Austria : R Foundation for Statistical Computing. [Disponible en] : <<http://www.r-project.org/>> [citado Dic 2013].

SÁEZ, Yago., ISASI, Pedro., and SEGOVIA, Javier. Interactive Evolutionary Computation algorithms applied to solve Rastrigin test functions. *Soft Computing as Transdisciplinary Science and Technology*. Springer Berlin Heidelberg, 2005. p. 682-691.

SCHWARTZ, Jay D., WANG, Wenlin., and RIVERA, Daniel E. Simulation-based optimization of process control policies for inventory management in supply chains. *Automatica*, 2006, vol. 42, no. 8, p. 1311-1320.

SILVER, Edward Allen. An overview of heuristic solution methods. *Journal of the Operational Research Society*, 2004, vol. 55, no. 9, p. 936-956.

URREA, Andrea., and TORRES, Fidel. Optimización de una política de inventarios por medio de búsqueda Tabú. En *III Congreso colombiano y I Conferencia Andina internacional*, 2006, p. 8.

VALENCIA, Marisol., GONZÁLEZ, Daniela., and CARDONA, Juan E. Metodología de un modelo de optimización para el pronóstico y manejo de inventarios usando el metaheurístico Tabú. In : *Revista Ingeniería*, 2014, vol. 24, no. 1, p. 13-27.. [Disponible en]: <<http://www.revistas.ucr.ac.cr/index.php/ingenieria/issue/view/1426>> [citado: junio de 2014]

VALENCIA, Marisol., DÍAZ, Francisco J., and CORREA, Juan Carlos. Planeación de inventarios con demanda dinámica. Una revisión del estado del arte. *Revista DYNA*, 2015, vol. 82, no. 190, p. 182-191.

VIDAL, Carlos., LONDOÑO, Julio C. and CONTRERAS, Fernando. Aplicación de Modelos de Inventarios en una Cadena de Abastecimiento de Productos de Consumo Masivo con una Bodega y N Puntos de Venta. *Ingeniería y Competitividad*, 2004. vol. 6, no. 1, p. 35-52.

WINKER, Peter., and GILLI, Manfred. Applications of optimization heuristics to estimation and modelling problems. *Computational Statistics and Data Analysis*, 2004, vol. 47, no. 2, p. 211-223.