

Calidad y medioambiente

Quality and Environment

Una evaluación de calidad para soluciones obtenidas a partir de variaciones algorítmicas del método Multi-Start*

Eduardo Redondo**

Universidad Nacional de Asunción. San Lorenzo, Paraguay

Recibido: 22 de enero del 2018 / Aprobado: 14 de mayo del 2018

doi: 10.26439/ing.ind2018.n036.2449

RESUMEN: Para este trabajo se experimentó con dos variaciones algorítmicas del método heurístico de búsqueda local Multi-Start, para la generación y evaluación de la calidad de soluciones provenientes de diferentes escenarios productivos. Estas variaciones fueron aplicadas a 29 modelos de refinerías petroleras para luego inferir sobre efectividad y conveniencia de cada variación sobre los modelos. Se utilizó un *software*, disponible comercialmente, de uso frecuente en problemas de mezclas en el sector industrial.

Palabras clave: optimización / algoritmos heurísticos

A quality assessment for solutions achieved through algorithmic variations of the Multi-Start method

ABSTRACT: For this research, an experiment with two algorithmic variations of the Multi-Start heuristic method for local search was held in order to generate and assess the quality of solutions coming from different manufacturing scenarios. These variations were applied to 29 oil refinery models to infer about the effectiveness and convenience of each model variation. A commercially available software, frequently used in blending problems at the industrial level, was used.

Key words: optimization / heuristic algorithms

* Se agradece al Dr. Alireza Tehrani por la supervisión de este trabajo.

** Correo electrónico: eredondo@ing.una.py

1. INTRODUCCIÓN

La modelización de operaciones industriales a través de la aplicación de la investigación de operaciones puede llegar a órdenes significativos en cuanto a número de restricciones y variables de los modelos correspondientes. Un modelo operativo industrial de base es aquel que representa algún proceso productivo; en este caso, para el estudio presentado en este trabajo, se dispone de modelos en los que se modelan dos tipos de actividades: producción y mezcla. Estos modelos corresponden a problemas no lineales (NLP) del tipo *The Pooling Problem* (Fieldhouse, 1993) y *The Blending Problem* (Gupte *et al.*, 2017), los cuales, debido a la dificultad computacional que representa su resolución, requieren de algoritmos heurísticos o procedimientos empíricos que proporcionen una solución considerablemente buena por parte del decisor. Entre los primeros algoritmos heurísticos utilizados para realizar búsquedas locales o de vecindario, se encuentra el algoritmo Multi-Start (MS), el cual consiste en realizar múltiples búsquedas a partir de muestras o soluciones iniciales aleatorias (Martí *et al.*, 2015), o, en otras palabras, consiste en iterar la etapa de generación de una solución inicial factible seguida de la etapa de búsqueda de una mejor solución en las inmediaciones de dicho punto inicial, repetir los pasos hasta que se satisfaga algún criterio de parada y devolver el mejor valor de la función objetivo (FO) obtenida.

De acuerdo con Dantzing (1963), cronológicamente la primera aplicación industrial fructífera de la programación lineal (LP) se ha dado en el sector petrolero. No obstante, desde la exploración de puntos de extracción, hasta el refinamiento y la distribución del producto, este es un sector que permite la aplicación de técnicas de programación sin limitarse a las lineales. Varvarezos (2008) afirma que las no-linealidades de los procesos que se llevan a cabo en este ámbito han causado el desplazamiento de las técnicas de optimización hacia el uso de métodos de programación no lineal (NLP). Un ejemplo de esto se puede encontrar en los departamentos encargados de la planificación de la producción de las refinerías, en donde frecuentemente se plantean problemas del tipo *The Blending* al momento de planificar la producción (Schrage, 2006). Para la resolución de estos problemas se emplean heurísticas y metaheurísticas de búsqueda local tales como Multi-Start, Variable Neighborhood Search (VNS), Búsqueda Tabú y Recocido Simulado (Herrán, 2012), y es mediante el uso de estos modelos matemáticos con sus respectivas técnicas de optimización que los encargados de planificar periódicamente la producción calculan los niveles de producción

“óptimos” o buenos para cada producto con el fin de proveer buenos y rentables puntos de operación (por ejemplo: reglaje de temperaturas de las columnas de destilación, unidades y máquinas a utilizar, cantidades de producción de cada producto y subproducto, etc.).

El presente trabajo muestra la evaluación de la calidad para soluciones obtenidas a partir de dos variaciones algorítmicas hechas con el método Multi-Start (MS): MS-Continuo (MSC) y MS-Encadenado (MSE), compara los valores medios de las FO arrojadas por los dos algoritmos aplicados en los modelos estudiados, e infiere sobre la efectividad relativa de cada uno de ellos de acuerdo con los análisis de desviación relativa porcentual (RPD) de los valores de las FO pertenecientes a los mismos modelos. En la segunda sección de este trabajo se presenta una introducción al funcionamiento de las heurísticas de búsqueda local y se describe de forma detallada el método MS y sus variaciones. En la tercera sección, se explica cómo fue realizado el experimento y qué factores fueron los analizados. En la cuarta sección, se discuten los resultados obtenidos y, finalmente, en la quinta sección, se presentan las conclusiones y recomendaciones de continuidad para futuras investigaciones.

2. HEURÍSTICAS DE BÚSQUEDA LOCAL

Como lo explican Martí y Reinelt (2011), las heurísticas, o simplemente métodos aproximativos, son aquellos utilizados en la práctica y en el ámbito de la optimización a fin de caracterizar cierto tipo de método de resolución de problemas catalogados como NP-difíciles que requieren de una resolución rápida y eficiente; es decir, se caracterizan por enfocarse en la velocidad del proceso de búsqueda y en la calidad de la solución obtenida. De acuerdo con la naturaleza o finalidad para la cual fue creada cada heurística, los mismos autores presentan una clasificación de los métodos más conocidos, en la cual se presenta la búsqueda local como una clase de método que busca mejorar una solución al partir de una solución inicial factible para permitir explorar otras soluciones colindantes a la primera de forma controlada por medio de la evaluación de las soluciones de la vecindad o en torno de la solución inicial. La vecindad se define como el grupo de soluciones que pueden ser obtenidas a consecuencia de pequeños cambios (perturbaciones) en la solución inicial, finalizándola cuando se alcanzan criterios de parada o bien cuando se incurre en perturbaciones no admitidas por el programa.

Por otra parte, el tiempo tomado por un método exacto para hallar una solución óptima en un problema complejo (asumiendo que existiera un método apropiado y exacto para la resolución del problema), sería significativamente superior al que representa la ejecución de un método heurístico. En general, aun sabiendo que no es posible asegurar la globalidad de ninguna solución obtenida a partir de una heurística (Martí *et al.*, 2013), debido a la practicidad en relación con los métodos exactos, el decisor termina recurriendo a estos métodos para la resolución de problemas reales de optimización. Posteriormente, a modo de evaluar cuán óptima o buena ha sido la resolución, se realizan evaluaciones de las calidades de las soluciones encontradas por medio de técnicas estadísticas comparativas.

2.1 Caracterización de una heurística de búsqueda local

Dos aspectos que caracterizan la eficiencia de una heurística de búsqueda local corresponden, por un lado, a la velocidad con la que se ejecuta hasta hallar los óptimos, mejorar la solución y/o satisfacer los criterios de parada, y, por otro, a la calidad de la solución hallada en términos de cuán buena es la solución encontrada.

Los métodos de resolución de problemas en investigación de operaciones son representados a través de esquemas o gráficos de bloques que reflejan las acciones lógicas a realizarse sobre los datos, y son descritos en detalle a través de algoritmos, los cuales serán posteriormente traducidos a un lenguaje de programación para su resolución vía herramientas computacionales. A partir de esto, puede explicarse la importancia de la “velocidad”, puesto que, a mayor complejidad del problema (problemas con espacios de búsqueda de grandes dimensiones, restricciones duras que producen espacios de búsqueda muy dispersos, funciones multiobjetivos, variables temporales o que manejan un gran volumen de datos), o del método de resolución o algoritmo que lo describa, mayor será el tiempo utilizado y, por consiguiente, mayores serán los costos asociados al proceso de toma de decisiones. De esta forma, para el caso analizado en este trabajo, debido al volumen de datos y ecuaciones, quedan descartados métodos exactos que pudieran ser utilizados para la optimización del problema de mezcla en refinerías, tales como *Enumerative Search*, *Backtracking*, *Branch and Bound*, *Dynamic Programming* y estrategias de programación lineal y entera, pues en la práctica representarían un elevado consumo de tiempo de ejecución (Nemaschnow, 2014).

En cuanto al segundo aspecto, la evaluación de la calidad de los resultados obtenidos por medio de una heurística se logra mediante la comparación de los resultados obtenidos con valores que pueden pertenecer a otros métodos. Este contraste entre soluciones permite visualizar cuán “óptima” es una solución o bien cuán relativamente acertada resulta la aplicación de una heurística sobre determinado modelo.

Para realizar estos contrastes, generalmente se utiliza la desviación porcentual relativa existente entre el valor de la solución obtenida por la heurística utilizada y los valores de:

- soluciones óptimas halladas en instancias menores del problema o con límites predefinidos que simplifiquen el problema,
- otras soluciones obtenidas a partir de métodos exactos (en caso de que estos métodos existan o sean aplicables al problema), otras heurísticas o métodos híbridos.

2.2 Multi-Start

Este algoritmo posee un enfoque iterativo en el cual, a través de cada iteración, se lleva a cabo una búsqueda de óptimos del problema, a partir de soluciones definidas aleatoriamente y perturbándose los valores de los elementos que componen la solución para generar así soluciones o puntos “vecinos” (colindantes) dentro del espacio solución. El propósito es inspeccionar otras soluciones en otras regiones dentro del espacio solución y hallar aquellas que proporcionen un mejor valor de la función objetivo (González y Rivera, 2015).

El hecho de que el algoritmo MS aplique los reinicios aleatorios le brinda una particular ventaja frente a otros métodos, ya que diversificando el análisis en diferentes regiones del espacio solución se lograría salir de la influencia de los óptimos locales con mayor facilidad.

Esquema básico del algoritmo MS:

- | | |
|---|---------------------|
| (1) Definir $i = 1$ | |
| (2) Mientras la condición de parada no sea satisfecha: | |
| (2.1) Construir una solución x_i | (Generación) |
| (2.2) Aplicar una búsqueda local para mejorar x_i
y asumir x_i' como solución obtenida | (Mejora) |
| (2.3) Si x_i' mejora la mejor solución, actualizarla
Definir $i = i + 1$ | (Evaluación) |
-

2.2.1 Variantes algorítmicas de MS

Martí y Reinelt (2011) afirman que el algoritmo MS puede ser considerado como un puente entre las heurísticas simples o clásicas, y las metaheurísticas complejas o modernas. Así también, otras variaciones del método pueden ser desarrolladas e incluso integradas a otras técnicas que permitan hibridar y mejorar las capacidades de búsqueda. Las variaciones más simples del algoritmo son utilizadas para comparar otros métodos y medir su contribución relativa, tal como se hace en Baluja (1995) para comparar diferentes algoritmos genéticos utilizando MS como base de contraste.

En este trabajo, debido a que se desea evaluar la calidad de dos variaciones propuestas de la heurística MS, se utilizan modelos que disponen de soluciones ya conocidas, lo cual permite contrastar y evaluar las soluciones halladas con estas variaciones. Dichas variantes algorítmicas difieren en el modo de selección de los datos de entrada y en la magnitud en cuanto a la perturbación de los elementos que componen la matriz empírica de base (MEB), cuya definición se presenta en el capítulo siguiente. Estos algoritmos son denominados en función del modo de generación de soluciones: Algoritmo MS-Continuo (MSC) y Algoritmo MS-Encadenado (MSE). Debido a altos tiempos de ejecución, y según experiencias de usuarios experimentados, para este trabajo se acuerda experimentar con la generación aleatoria de 10 casos como mínimo y 50 casos como máximo.

2.2.2 Algoritmo MS-Continuo (MSC)

Con esta variante se pretende generar escenarios secundarios a partir de un escenario de base definido por la MEB cuyos elementos crean el escenario de base Y_0 . Y, mediante una perturbación en esos elementos, se generan nuevos escenarios Y_i con ($i \neq 0$). Un lanzamiento MSC ejecuta 50 búsquedas utilizando escenarios secundarios. Los resultados obtenidos por esta técnica darán aleatoriamente valores en el vecindario más próximo al punto de base. El porcentaje de perturbación de los elementos de la MEB es constante para cada generación.

Esquema del algoritmo MSC:

- (1) Y_0 = Adoptar un escenario de base según la MEB
 - (2) s^* = Buscar y definir el óptimo local en función de Y_0
 - (3) N = Definir el tamaño del MS
 - (4) Repetir hasta alcanzar N
 - (4.1) Y_i = Definir nuevo escenario sobre la base de perturbaciones arbitrarias en Y_0
 - (4.2) s^i = Buscar el óptimo local en función de Y_i
 - (4.3) s^* = Seleccionar el óptimo local: mejor solución s^* vs. s^i
-

La figura 1 muestra un escenario Y_0 sirviendo de base para la generación de otros cincuenta escenarios secundarios que serán adoptados como puntos de partida para la resolución de cincuenta problemas lineales.

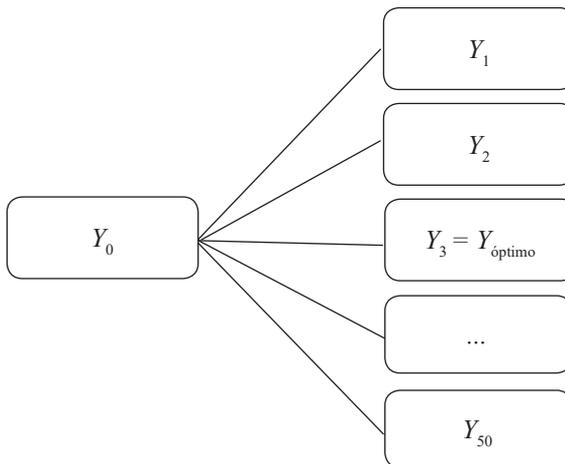


Figura 1. Escenarios secundarios Y_i dependientes del escenario de base Y_0 para la variación algorítmica MSC
Elaboración propia

A modo de interpretar el funcionamiento de la variación MSC, se plantea la hipotética situación en la que existe una sola incógnita en un escenario dado, y esta se simboliza con la letra X_i . Los valores de esta incógnita variarían según la asignación de perturbaciones en la fase de generación de escenarios yendo desde X_0 para la MEB inicial (escenario inicial Y_0) hasta X_{50} para el escenario Y_{50} . Estos valores fueron distri-

buidos en el eje de abscisas en la figura 2. En esta figura, en el eje de ordenadas se representan valores genéricos de las FO asociadas a los cincuenta escenarios definidos por las X_i . Cada círculo representa una de las cincuenta ejecuciones del programa (habiéndose establecido el tamaño de la búsqueda en 50).

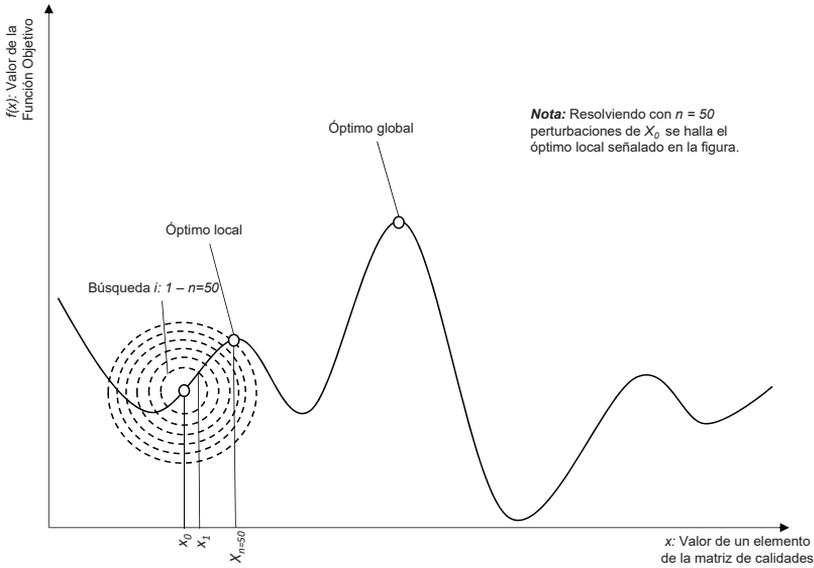


Figura 2. Representación esquemática del proceso de búsqueda local en una función genérica para la variación algorítmica MSC. La abscisa representa un valor asociado a un elemento de la matriz MEB que será perturbado $N = 50$ veces
Elaboración propia

Para la variación algorítmica MSC, la mejor solución hallada corresponde a la máxima/mínima solución obtenida (s^*).

2.2.3 Algoritmo MS-Encadenado (MSE)

Con esta estrategia se resuelven los problemas de forma encadenada, con escenarios de base distintos pero dependientes. Cada escenario de base corresponderá a uno obtenido a partir de una variación realizada sobre el escenario precedente que haya brindado la mejor solución (minimizando o maximizando una FO). En otras palabras, de manera encadenada y sucesiva se crea el escenario Y_i a partir del escenario Y_{i-1} que

haya otorgado la mejor FO, y se agrupan consecutivamente formaciones de MSC. Con esta variante en el algoritmo del método MS, se pretende lograr una búsqueda con un grado de libertad extra, y aumentar así la probabilidad de hallar óptimos globales al saltar posibles estancamientos en óptimos locales.

Esquema del algoritmo MSE:

- (1) Y_0 = Adoptar un escenario de base según la MEB
 - (2) repetir hasta alcanzar N
 - (2.1) s^* = Buscar el óptimo local en función de Y_0
 - (2.2) N = Definir el tamaño del MS
 - (2.3) n = Definir el número de fases
 - (2.4) repetir hasta alcanzar N/n
 - (2.4.1) Y_1 = Definir el nuevo escenario sobre la base de perturbaciones en Y_0
 - (2.4.2) s' = Buscar el óptimo local en función de Y_1
 - (2.4.3) s^* = Seleccionar el óptimo local \rightarrow mejor solución s^* vs. s'
-

En la figura 3 se muestra el encadenamiento logrado entre escenarios del MSC_i y del MSC_{i+1} .

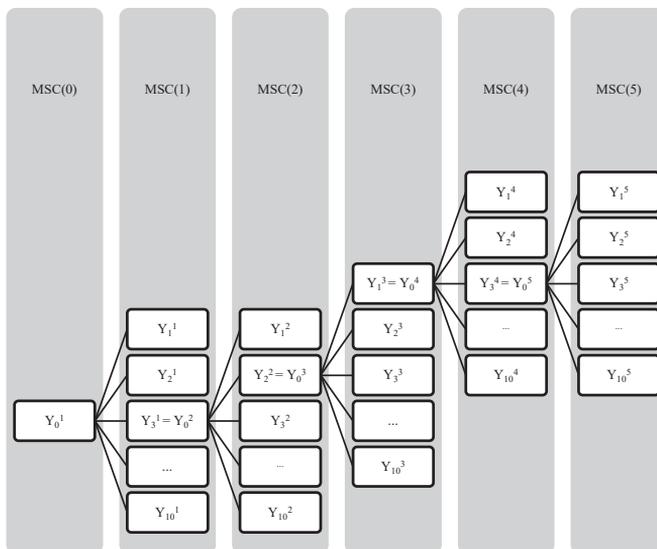


Figura 3. Escenarios secundarios Y_i dependientes del mejor escenario precedente Y_{i-1} para la variación algorítmica MSE
Elaboración propia

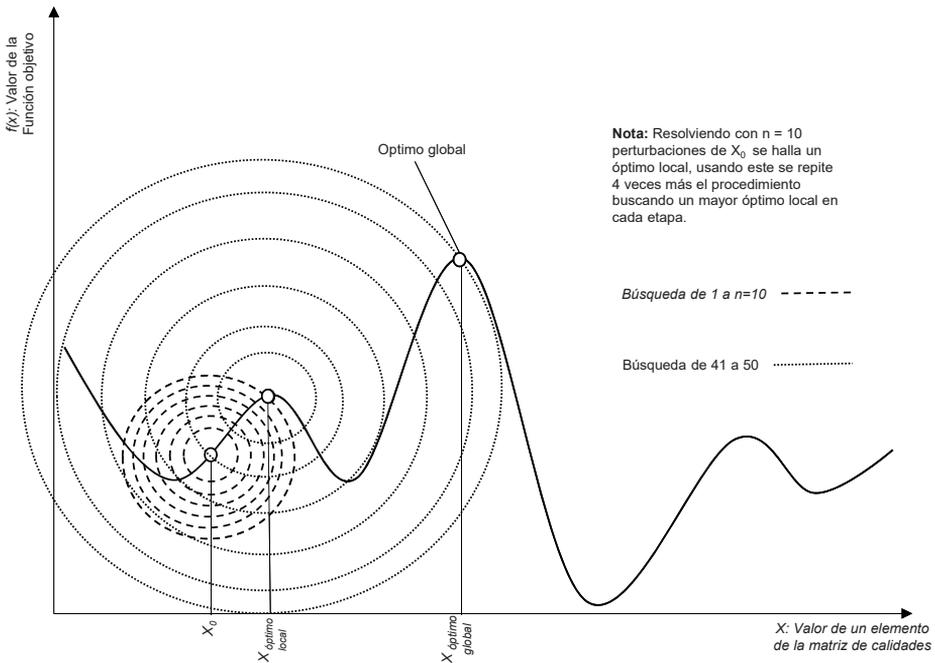


Figura 4. Representación esquemática del proceso de búsqueda local en una función genérica para la variación algorítmica MSE. La abscisa representa un valor asociado a un elemento de la matriz MEB que será perturbado $n \times N = 5 \times 10 = 50$ veces
Elaboración propia

En la figura 4, al igual que para la variación MSC, para una mejor interpretación del funcionamiento de la variación MSE, se plantea la hipotética situación en la que existe una sola incógnita en un escenario dado, y esta se simboliza con la letra X_i . Los valores de esta incógnita variarían según la asignación de perturbaciones en la fase de generación de escenarios. Además, se establece el número de fases $n = 5$, con lo cual se deberán realizar cinco fases de búsqueda con tamaño 10 para totalizar las 50 búsquedas. Cada círculo representa una de las 10 ejecuciones del programa en cada fase. Los grupos de 10 círculos concéntricos componen un MSE, es decir, luego de haberse definido cinco fases, existen cinco grupos de MSE con 10 NLP cada uno. El escenario en el cual se haya encontrado la mejor solución del primer grupo será el escenario de base adoptado para la siguiente búsqueda a partir del cual se realizarán las siguientes generaciones. Al final de la búsqueda, se tendrán tantos óptimos locales como grupos de MSE efectuados. La mejor solución hallada corresponde a la máxima/mínima solución obtenida en todos los grupos MSE.

3. DESCRIPCIÓN DEL EXPERIMENTO

En este trabajo, para la resolución de los modelos matemáticos, se aplican métodos de linealización sucesiva (SLP) como el de Expansión de Taylor de Primer Grado. Con este método se quiebran algunas bilinealidades al asignar valores temporalmente “fijos” a ciertas variables. Estas asignaciones definirán el escenario de aplicación del método MS. Estos valores predefinidos están asociados a un posible escenario “Y”, bajo cuya influencia se resuelve el modelo, esto es, al realizar $i \{i > 0\}$ cambios o conjunto de cambios, se genera el mismo número de escenarios y soluciones, pero con valores y tiempos de resolución distintos. Las i soluciones halladas corresponden a óptimos locales en la región de factibilidad bajo influencia del escenario en cuestión Y, y el número de pasos realizados para hallar la solución que satisface el criterio de parada define la velocidad de convergencia del modelo estudiado.

Fueron utilizados 29 modelos de planificación de la producción que cuentan en promedio con 3000 (tres mil) ecuaciones y 8000 (ocho mil) incógnitas. Estos modelos pertenecen a 10 refinerías petroleras. Para la linealización y resolución de los problemas, se utilizó el programa Generalized Refining Transportation Marketing Planning System (GRTMPS); *software* comercialmente disponible de la compañía Haverly Systems. Y, para la evaluación de la calidad de los resultados arrojados por los algoritmos, los resultados fueron tratados con las herramientas estadísticas de Microsoft Excel.

3.1 Selección de modelos

La toma de decisión y puesta en marcha de los flujos productivos para la mezcla y obtención de derivados propios del refinamiento del crudo se basa en modelos de planificación que a su vez se construyen sobre modelos matemáticos ajustados vía previa simulación. Puede decirse que cada modelo matemático posee un “comportamiento” único que depende de las variables de entrada (condicionantes de frontera), de los métodos de resolución y de las técnicas de búsqueda local utilizadas. Este comportamiento se refleja en el modo único en el que se resuelve el problema hasta hallar la solución al considerar aspectos como si converge o no en una solución factible, número de pasos hasta la convergencia, tiempo utilizado en cada paso, si la solución hallada corresponde a un óptimo o no, si satisface las tolerancias o no, y el valor de la función objetivo hallada.

A modo de lograr un espacio muestral representativo y así visualizar el comportamiento de cada modelo, fueron seleccionados tres de ellos para cada uno de los 10 sitios industriales existentes (A, B, C, \dots, J), a excepción de un sitio cuyo comportamiento reflejó una notable estabilidad, por lo que se optó por recurrir a dos modelos solamente, lo que totalizó en 29 los modelos analizados ($A_1, A_2, A_3, B_1, B_2, B_3, \dots, J_1, J_2$).

A modo de aumentar la fiabilidad del análisis, para cada sitio se seleccionaron modelos que reflejan el flujo productivo promedio asociado a escenarios con niveles de fluctuación poco cambiantes en lo que refiere a cotizaciones internacionales y parámetros productivos dependientes de los niveles de demanda en el mercado. Esta información fue relevada vía entrevistas a operadores y modeladores de la compañía.

3.2 Selección de datos de entrada y parámetros iniciales para la linealización

El MS precisa disponer de datos de base/entrada correspondientes a índices de operación, parámetros industriales, de calidad y de distribuciones, entre otros vinculados a cada sitio en particular (Glover, 2000, 2007). Si bien estos elementos poseen valores desconocidos, estos datos son necesarios pues describen escenarios iniciales para la resolución de los problemas y arranques del método, ya que a partir estos se aplicará el método MS hasta hallar o no los óptimos locales. Algunos métodos, según Chinneck (2008), no permiten llegar a una solución óptima local sin pasar primeramente por soluciones o puntos dentro de la región de factibilidad. Por consiguiente, es deseable que esta primera solución corresponda a una solución conocida proveniente de una colección de resultados históricos, es decir, se utiliza un punto de convergencia factible ya conocido dentro del espacio de soluciones. Mitchell *et al.* (1992), Mammen y Hogg (1997) y Conrad *et al.* (2007) demuestran que en un NLP existe una relación directa entre la dificultad de determinar su factibilidad y la determinación de los parámetros para la aplicación de MS al problema a causa de las restricciones. De hecho, es difícil encontrar una primera solución factible para la generalidad de los NLP. Por lo tanto, existen algunas técnicas que buscan mejorar la convergencia y la resolución del programa; entre ellas, se hallan técnicas de Penalties, Step Bounding, Dynamic Dampening, Dampening y otras técnicas heurísticas como Multi-Start.

Para garantizar la convergencia del problema y que la solución inicial obtenida corresponda a una perteneciente a la región de factibilidad, en concordancia con lo citado en el párrafo anterior para todos los modelos, fueron definidos inicialmente “puntos de base” que corresponden a matrices empíricas de base (MEB), cuyos elementos son incógnitas de operación con valores temporalmente adoptados que servirán para linealizar el problema y resolverlo por medio de un resolvidor de modelos lineales. La MEB es adoptada temporal y empíricamente bajo influencia de resultados obtenidos con el mismo modelo en meses anteriores. De esta forma, pueden conocerse los rangos operativos de las incógnitas (salvo en escenarios operativos amenos), lo cual agiliza significativamente la búsqueda de óptimos, puesto que el punto de partida corresponde a un punto de la vecindad. Los resultados bajo efectos de la MEB se comparan con los valores iniciales, los errores son medidos y analizados para finalmente compararlos con ratios de tolerancia.

3.3 Definición de parámetros para la aplicación de MS

- i. Elementos de la MEB: elementos cuyos valores fueron fijados empíricamente mediante soluciones de ejecuciones anteriores del programa productivo en las plantas industriales, es decir, son valores pasados conocidos que pueden adoptar las variables del problema y que, al saberse que en el pasado han adoptado estos valores, se asume que los óptimos buscados deben rondar esta vecindad. Entre los elementos vinculados a la MEB se encuentran, por ejemplo, las variables asociadas a las calidades físico-químicas de las mezclas transportadas entre las unidades de producción, tales como densidades mínimas y máximas admisibles, tenor máximo de azufre, índice mínimo deseado de RON (Ron Octane Number), viscosidades dinámicas y cinemáticas, densidades, distribuciones de masas o volúmenes, temperaturas de operación y mezclado, límite mínimo para el índice de cetano, temperaturas mínimas (Flash Point), máximas (Coud y Freezing Points).
- ii. Tamaño del MS: número de casos a ser generados {10,50} que indica el número de escenarios “Y” a generar.
- iii. Probabilidad de perturbación de cualquier elemento de la MEB: puede variar entre {0,25; 0,50 y 1,00}
- iv. Semilla de número aleatorio: valor inicial o semilla que se toma de base para generar números pseudoaleatorios. En este caso, este

número aleatorio es empleado para la generación de nuevos coeficientes que serán multiplicados a los elementos de la MEB para así cambiar o perturbar dichos elementos.

- v. Máximo cambio relativo: máxima perturbación relativa y aleatoria aplicable a elementos de la MEB $\{0,5\}$. Un elemento \mathbf{a} puede variar su valor en $\pm 0,5\mathbf{a}$.

3.4 Creación del banco de soluciones

A partir de la resolución de los problemas descritos anteriormente y considerando la posibilidad de combinar distintos tamaños de búsquedas, se analizaron siete combinaciones para cada uno de los 29 modelos. Para la variante MSC, se trabajó con tamaños de 10 y 50 búsquedas, y variaciones en los valores de los elementos de la MEB en órdenes de magnitudes de 25 %, 50 % y 100 %. Para MSE, se trabajó únicamente con 50 como tamaño de la búsqueda, y variaciones en los valores de los elementos de la MEB en órdenes del 25 %, 50 % y 100 %.

Los resultados fueron analizados utilizando el método de desviación porcentual relativa (RPD: Relative Percentage Deviation), la cual es computada en cada instancia de acuerdo con la expresión (1).

$$RPD\% = 100 \cdot \left| \frac{f_{\text{algoritmo}} - f_{\text{benchmark}}}{f_{\text{benchmark}}} \right| \quad (1)$$

donde $f_{\text{benchmark}}$ es la mejor solución conocida obtenida empíricamente por resolución previa mediante el algoritmo MS simple, y $f_{\text{algoritmo}}$ es la solución obtenida con el algoritmo modificado bajo evaluación.

Los datos fueron recopilados para conformar una base de datos consistente en el almacenamiento de las FO obtenidas. Los resultados promediados para cada sitio se presentan en la tabla 1, donde se muestra el RPD con respecto a la FO del escenario inicial, lo cual representa un indicador de mejoría obtenida en la búsqueda efectuada.

4. RESULTADOS

Se evaluaron las calidades de dos variaciones algorítmicas del Método de Búsqueda Local Multi-Start. Las variaciones consistentes en leyes de generación de escenarios mediante parámetros de perturbación de los elementos que componen a las matrices de base (escenarios inicia-

Tabla 1
 RPD entre la solución obtenida con el escenario de base y la mejor de las soluciones halladas por MS según el % de perturbación y los algoritmos MSC y MSE. Nomenclatura XX%-NN-YYY → (% de perturbación, tamaño MS, MSC o MSE).
 Aclaración: NA = dato sesgado

Banco de datos completo - RPD									
Refinerías	25 % 10 MSC	25 % 50 MSC	50 % 50 MSC	100 % 50 MSC	25 % 50 MSE	50 % 50 MSE	100 % 50 MSE	50 % 50 MSE	100 % 50 MSE
A	0,655 %	0,451 %	1,124 %	0,739 %	0,442 %	0,028 %	0,716 %	0,028 %	0,184 %
B	0,060 %	0,182 %	0,222 %	0,237 %	0,147 %	0,222 %	0,184 %	0,222 %	0,184 %
C	0,068 %	0,046 %	0,043 %	0,041 %	0,068 %	0,005 %	NA	0,005 %	NA
D	0,140 %	0,004 %	0,059 %	0,009 %	0,140 %	0,011 %	0,000 %	0,011 %	0,000 %
E	0,386 %	0,386 %	0,009 %	0,371 %	0,611 %	0,001 %	0,000 %	0,001 %	0,000 %
F	0,049 %	0,049 %	0,049 %	NA	0,049 %	0,277 %	NA	0,277 %	NA
G	0,208 %	0,195 %	0,174 %	0,132 %	0,287 %	0,374 %	0,392 %	0,374 %	0,392 %
H	0,432 %	0,450 %	0,447 %	0,270 %	0,444 %	0,288 %	0,270 %	0,288 %	0,270 %
I	0,210 %	0,262 %	0,000 %	0,000 %	0,206 %	0,000 %	0,000 %	0,000 %	0,000 %
J	0,221 %	0,223 %	0,223 %	0,223 %	0,222 %	0,221 %	0,222 %	0,221 %	0,222 %

Elaboración propia

les) fueron aplicadas a diferentes modelos para analizar la desviación porcentual relativa a una solución de base ya conocida. Fueron seleccionados modelos con sus respectivas soluciones conocidas a modo de tener un criterio definido que permita la evaluación de la calidad obtenida mediante la variación algorítmica del método MS. Los resultados fueron expuestos en la tabla 1.

4.1 Inferencias sobre el uso de MSC vs. MSE

Las RPD obtenidas permitieron inferir sobre la conveniencia aplicativa de cada variación para con los modelos. Sin embargo, se generaron inconvenientes en el transcurso del estudio, ya que la generación aleatoria de escenarios sin discriminación de la naturaleza de la variable perturbada tiende a producir. Esto se debe a que las modificaciones aleatorias se dieron sobre todas las variables, en ocasiones, obligando al programa a resolver el problema con valores irreales o fuera de los rangos permitidos por las restricciones del modelo productivo. La detección de estos errores o fallas y su correspondiente corrección hicieron posible robustecer el programa.

Tabla 2
Conveniencias para los 10 sitios industriales en referencia a algoritmos y tamaños de MS para hallar óptimos locales de manera efectiva

Sitios	MSC o MSE	Perturbación %	Tamaño MS
A	MSC	50	19
B	MSC	50	18
C	MSC	25	23
D	MSC	25	14
E	MSC	25	17
F	MSE	50	3X10
G	MSC	50	14
H	MSC	25	39
I	MSC	25	20
J	MSC	25	21

Elaboración propia

A modo global, los resultados obtenidos no demostraron ventajas contundentes en lo que concierne a la aplicación de una u otra variante. A nivel particular, podría inferirse sobre el tamaño de la búsqueda de acuerdo con la posición usual del óptimo hallado en referencia a la posición promedio en donde fue hallado el óptimo por el método MS. De esta manera, se reduce el espectro de búsqueda en cada modelo de acuerdo con los resultados expuestos. De igual forma, podría inferirse sobre el uso apropiado de una de las dos variaciones algorítmicas y la magnitud de la perturbación a realizar en cada modelo (porcentaje de elementos perturbados). Ver la tabla 2.

4.2 Análisis de efectividad de ambos algoritmos

Realizando comparaciones entre los resultados arrojados por ambos algoritmos (ver gráfico 1), y teniendo en cuenta la globalidad de las FO calculadas por cada método en el mismo modelo, se resume que ambos permiten hallar mejorías en la búsqueda de óptimos locales en distintos grados: según el porcentaje de elementos perturbados en la MEB, puede concluirse que el método MSE ha obtenido una ligera mejoría en el hallazgo de FO mayores.

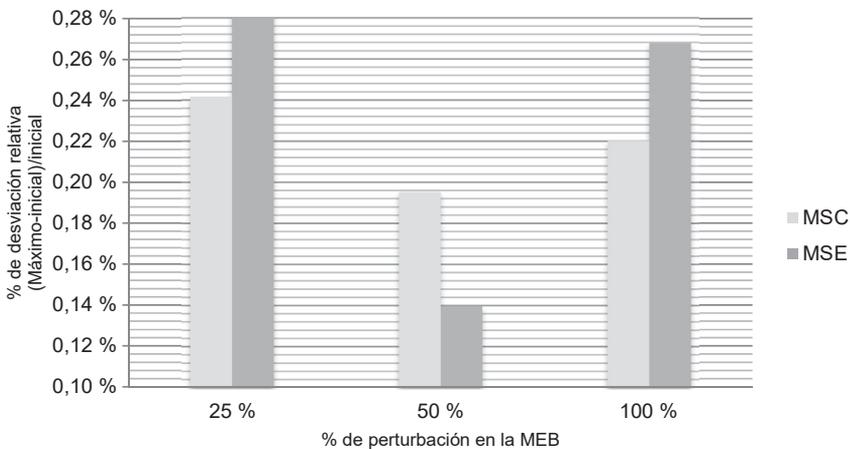


Figura 5. Resultado obtenido en la evaluación de la calidad al analizar las mejoras en las funciones objetivas. Relación entre RPD y porcentaje de perturbación de los elementos de la MEB

Elaboración propia

Sin embargo, como se puede ver en la tabla 3, el resultado no fue contundente debido a la magnitud no considerable de las mejorías. Luego, se asume que ambas estrategias son congruentes a modo general, no así a modo específico. Las estrategias parecieran adaptarse mejor a ciertos modelos al lograr mejoras significativas esporádicas, mientras que en otros modelos no se registraron variaciones.

Tabla 3
*Resumen de mejoras obtenidas para cada variación
 algorítmica según el porcentaje de elementos
 perturbados en la MEB*

Perturbación %	MSC %	MSE %
25	0,24	0,28
50	0,20	0,14
100	0,22	0,27
	0,22	0,23

Elaboración propia

4.3 Efecto secundario de las perturbaciones

Un modelo se degenera cuando los elementos y las variables adoptan valores no previstos en la construcción del modelo. La capacidad de un modelo de soportar perturbaciones es un notorio y básico indicador de su robustez. Por medio de perturbaciones arbitrarias, se logra fortalecer un modelo al hallar sus puntos de quiebre o falla y corregir la falencia. Una fuerte dependencia entre las degeneraciones obtenidas en los modelos y el nivel de perturbación aplicada se ha constatado en todos los modelos evaluados. En el gráfico 2, puede apreciarse en el eje de ordenadas el porcentaje de degeneraciones halladas en el experimento al someterlas al porcentaje de perturbaciones descrito en el eje de abscisas. De esta forma, no solo se logró identificar los sitios con mayores índices de robustez en sus modelos, sino que, a su vez, se logró detectar posibilidades de fortalecimiento del modelo en otros sitios. Los casos con fallas fueron considerados como casos aberrantes y no fueron añadidos al banco de soluciones.

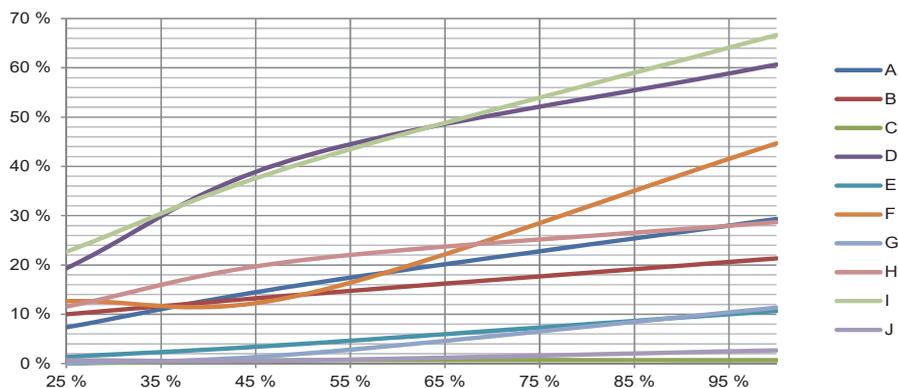


Figura 6. Porcentaje de problemas degenerados o fallas en el eje de ordenadas vs. porcentaje de elementos de la MEB que fueron perturbados para 8 de los 10 sitios analizados
Elaboración propia

5. CONCLUSIONES Y RECOMENDACIONES

La aplicación de heurísticas para incrementar las posibilidades de hallar óptimos locales con mayor eficiencia es muy prometedora. Al ser cada modelo una única estructura de condiciones e interconexión entre agentes, es muy difícil garantizar la efectividad de un algoritmo sin una evaluación de calidad de la heurística preliminarmente. De esta forma, es posible optimizar las operaciones actuales reales de una organización al ajustar el uso de una técnica que ya vienen utilizando cotidianamente. Esta mejora, a corto plazo, se traduce en beneficios monetarios para la industria al agilizar sus operaciones y planificar los ciclos operativos de manera más eficiente.

Esta rama ofrece oportunidades de mejora y adaptación de técnicas existentes a situaciones específicas de cada organización que aún no han sido exploradas. En este contexto, se recomienda continuar el estudio abarcado considerando una hibridación de técnicas evolutivas con uso de memoria adaptativa como las propuestas hechas por Lan y DePuy (2006).

REFERENCIAS

- Baluja, S. (1995). *An Empirical Comparison of Seven Iterative and Evolutionary Heuristics for Static Function Optimization*. Pennsylvania: Carnegie Mellon University.
- Chinneck, J. W. (2008). *Feasibility and Infeasibility in Optimization: Algorithms and Computational Methods*. New York: Springer Science & Business Media.
- Conrad, J., Gomes, C. P., Hoeve, W. V., Sabharwal, A. y Suter, J. (2007). Connections in Networks: Hardness of Feasibility Versus Optimality. *Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems Lecture Notes in Computer Science* (pp. 16-28). Berlín: Springer. doi: 10.1007/978-3-540-72397-4_2.
- Dantzig, G. (1963). *Linear programming and extensions*. Princeton: Princeton University Press.
- Fieldhouse, M. (1993). The pooling problem. En T. Ciriani and R. Leachman (ed.), *Optimization in Industry: Mathematical Programming and Modeling Techniques in Practice* (pp. 223-230). New York: John Wiley & Sons Ltd.
- Glover, F. (1997). Tabu Search and Adaptive Memory Programming-Advances, Applications and Challenges. In *Interfaces in Computer Science and Operations Research: Advances in Metaheuristics, Optimization, and Stochastic Modeling Technologies* (pp. 1-75). Boston: Kluwer Academic Publishers.
- Glover, F. (2000). Multi-Start and Strategic Oscillation Methods-Principles to Exploit Adaptive Memory. In *Computing Tools for Modeling, Optimization and Simulation: Interfaces in Computer Science and Operations Research* (pp. 1-23). Colorado: Kluwer Academic Publishers.
- González, F. y Rivera, J. C. (2015). *A Multi-Start Iterative Local Search for the k-Traveling Repairman Problem*. Medellín: Universidad EAFIT.
- Gupte, A., Ahmed, S., Dey, S. S. y Cheon, M. S. (2017). Relaxations and discretizations for the pooling problem. *Journal of Global Optimization*, 67(3), 631-669. doi:10.1007/s10898-016-0434-4.

- Herrán, A., de la Cruz, J. M. y De Andrés, B. (2012). Global Search Metaheuristics for planning transportation of multiple petroleum products in a multi-pipeline system. *Computers & Chemical Engineering*, (37), 248-261.
- Lan, G. y DePuy, G. W. (2006). On the effectiveness of incorporating randomness and memory into a multi-start metaheuristic with application to the set covering problem. *Computers & Industrial Engineering*, 51(3), 362-374.
- Mammen, D. L. y Hogg, T. (1997). A new look at the easy-hard-easy pattern of combinatorial search difficulty. *Journal of Artificial Intelligence Research*, 7, 47-66.
- Martí R., Lozano J. A., Mendiburu A. y Hernando L. (2015) Multi-start Methods. En: Martí R., Panos P., Resende M. (eds) *Handbook of Heuristics* (pp.1-21). Cham: Springer.
- Martí, R. y Reinelt, G. (2011). *The Linear Ordering Problem: exact and heuristic methods in combinatorial optimization*, volume 175. Berlin: Springer.
- Martí, R., Resende, M. G. y Ribeiro, C. C. (2013). Multi-start methods for combinatorial optimization. *European Journal of Operational Research*, 226(1), 1-8.
- Mitchell, J. E. y Todd, M. J. (1992). Solving combinatorial optimization problems using Karmarkar's algorithm. *Mathematical Programming*, 56(1-3), 245-284.
- Nesmachnow, S. (2014). An overview of metaheuristics: accurate and efficient methods for optimisation. *International Journal of Metaheuristics*, 3(4), 320-347.
- Schrage, L. E. (2006). *Optimization modeling with LINGO*. Chicago: LINDO Systems.
- Varvarezos, D. K., Joffe, B. J., Paules IV, G. E. y Kunt, T. (2008). New optimization paradigms for refinery planning. In *Proceedings Foundations of Computer-Aided Process Operations*, (pp. 441-445). Cambridge: Focapo.

