

Universidad Central “Marta Abreu” de Las Villas,
Facultad de Matemática, Física y Computación,
Centro de Estudios de Informática.



Algoritmo Híbrido basado en VMO y ECS.
Implementación y Validación.

Autor: Sergio Luis Ruiz Manzo.

Tutores: Lic. Carlos Alberto Martínez Pérez.

Dr. Yasel José Costa Salas.

Santa Clara, 2013

Dictamen con derechos de autor para MFC



Hago constar que el presente trabajo de diploma fue realizado en la Universidad Central “Marta Abreu” de Las Villas como parte de la culminación de estudios de la especialidad de Ciencia de la Computación, autorizando a que el mismo sea utilizado por la Institución, para los fines que estime conveniente, tanto de forma parcial como total y que además no podrá ser presentado en eventos, ni publicado sin autorización de la Universidad.

Firma del Autor

Los abajo firmantes certificamos que el presente trabajo ha sido realizado según acuerdo de la dirección de nuestro centro y el mismo cumple con los requisitos que debe tener un trabajo de esta envergadura referido a la temática señalada.

Firma del Autor

Firma del Tutor

Firma del Tutor

Firma del Responsable del Seminario

“La alegría está en la lucha, en el intento, en el sufrimiento implicado.

No en la victoria propiamente dicha.”

Mahatma Gandhi

A Él, gloria y potencia...

A toda mi gran familia.

A Dios por darme la oportunidad.

A mis padres por dar sus vidas para que yo haga realidad este sueño.

A toda mi familia por enseñarme lo más importante, el amor.

A mis hermanos Alejandro, Rodolfito y Boris por estar siempre.

*A mis amigos del terruño, cuyos nombres no cabrían en este documento, por crecer
conmigo y apoyarme siempre.*

*A todos mis amigos de la universidad por su amistad incondicional en los malos y buenos
momentos.*

A Carlos que más que mi tutor ha sido mi amigo.

Al tigre por su incondicional apoyo.

*A todos mis profesores, desde el primero hasta el último, que me han inculcado valores
imprescindibles para llegar a ser un hombre de verdad.*

A todos los que me han apoyado durante este camino.

*A todos los que en algún momento dudaron de mi o pusieron algún obstáculo en mi
camino, si es que hay alguien que lo haya hecho.*

*A todos los que de algún modo han contribuido a que hoy este sueño se haya hecho
realidad.*



SÍNTESIS

La hibridación entre meta heurísticas poblacionales y meta heurísticas de trayectoria simple constituye, en la mayoría de los casos, una efectiva propuesta algorítmica para resolver complejos problemas de optimización continua. Una buena explotación de las soluciones ya encontradas, así como una correcta exploración del espacio de búsqueda son factores imprescindibles para obtener soluciones de calidad. Dicha hibridación se hace mucho más eficaz cuando la búsqueda local es aplicada en las áreas más prometedoras del espacio de soluciones. Esta investigación estará encaminada a implementar y validar un modelo que combina de manera equilibrada estos factores, tomando como base poblacional para la exploración el modelo basado en Mallas Variables dentro del cual será implementada la Búsqueda por Agrupamiento Evolutivo, estrategia que detecta zonas promisorias en el espacio de soluciones. Sobre dichas zonas será incorporada una etapa de Búsqueda Local para lograr la explotación de las mismas, con el objetivo de presentar un modelo más robusto. Los resultados de esta investigación son la obtención de distintas variantes de un algoritmo híbrido con mejor comportamiento que el modelo basado en VMO puro o con búsqueda local aplicada a priori para la resolución de problemas de optimización continua.



ABSTRACT

Hybridization between populational metaheuristics and simple path metaheuristics constitutes, in the majority of cases, an effective algorithmic approach to solve complex problems of continuous optimization. A good operation of the solutions already encountered, and a good search space exploration are essential factors to obtain quality solutions. Such hybridization is much more effective when local search is applied in the most promising areas of the solution space. This research will aim to implement and validate a model that in a balanced way combines these factors, using the model based on VMO for the exploration in which will be implemented the Evolutionary Clustering Search, strategy that detects promising areas in the solution space. On such areas will be incorporated a stage of local search to achieve exploiting them, in order to present a more robust model. The results of this research are to obtain different variants of a hybrid algorithm that improve the model based on VMO without local search or with local search applied a priori for solving continuous optimization problems.



TABLA DE CONTENIDOS

INTRODUCCIÓN.....	1
1 COMPONENTES PARA LA HIBRIDACIÓN	7
1.1 OPTIMIZACIÓN BASADA EN MALLAS VARIABLES.....	8
1.1.1 Descripción general de la meta heurística	9
1.1.2 Proceso de expansión.....	9
1.1.3 Proceso de contracción	13
1.2 MÉTODOS DE BÚSQUEDA LOCAL EMPLEADOS	15
1.2.1 Método Solis Wets	17
1.2.2 Método Simplex de Nelder y Mead.....	19
Reflexión, Expansión y Contracción	20
Ventajas y Desventajas del método.....	22
1.3 FUNDAMENTOS DE LA BÚSQUEDA POR AGRUPAMIENTO EVOLUTIVO	22
2 ALGORITMO HÍBRIDO BASADO EN VMO Y ECS	25
2.1 ALGORITMO HÍBRIDO PROPUESTO	25
2.2 PROCESO DE CLUSTERIZACIÓN O AGRUPAMIENTO.....	27
2.2.1 Asimilación simple	29
2.2.2 Asimilación por cruzamiento.....	30
2.2.3 Asimilación por camino.....	31
2.3 MÓDULO DE ANÁLISIS	33
2.4 OPERADOR PARA FOMENTAR LA DIVERSIDAD BASADO EN CLUSTERING.....	34
2.5 ALTERNATIVAS DEL MODELO HÍBRIDO BASADO EN VMO Y ECS ESTUDIADAS	35
3 ANÁLISIS EXPERIMENTAL DE LOS RESULTADOS	37
3.1 TÉCNICAS ESTADÍSTICAS	37
Descripción de los tests estadísticos utilizados.....	38
3.2 MARCO EXPERIMENTAL.....	39
3.3 ANÁLISIS DE LOS RESULTADOS DE LOS TEST ESTADÍSTICOS PARA LAS DISTINTAS VARIANTES DE DISEÑO	40
Dimensión 10.....	40
Dimensión 30.....	44
Dimensión 50.....	45
CONCLUSIONES.....	48



RECOMENDACIONES	49
REFERENCIAS BIBLIOGRÁFICAS	50
ANEXOS.....	55
ANEXO A. TABLAS DE LOS VALORES OBTENIDOS PARA CADA UNA DE LAS FUNCIONES DE PRUEBA DEL CEC'2005 POR LOS MODELOS ESTUDIADOS.	55
ANEXO B. BREVE DESCRIPCIÓN DE LAS FUNCIONES UTILIZADAS COMO PRUEBA	63



INTRODUCCIÓN

Los problemas de optimización continua desempeñan un papel esencial en la formulación de muchos problemas básicos de la vida real. Numerosas investigaciones aseguran este hecho, tanto en la gestión de servicios (planificación de los servicios hospitalarios (Lamiri, Grimaud et al. 2009), cabina de servicio (Schilde, Doerner et al. 2011) los problemas de planificación de la salud (Nickel, Schröder et al. 2012) (Shariff, Moin et al. 2012), etc.), como aplicaciones industriales (planificación de la producción (Clark, Almada-Lobo et al. 2011), el avance ingeniería de diseño (Liao 2010), la planificación financiera (Guillén, Badell et al. 2007), la gestión de riesgos (Papadakos, Tzallas-Regas et al. 2011), entre otros, todos ellos van desde problemas clásicos hasta problemas más complejos que se abordan necesariamente desde la perspectiva de la Inteligencia Artificial, algunos de estos problemas de optimización son relativamente fáciles de resolver como los problemas lineales (las restricciones y la función objetivo constituyen expresiones lineales) y otros no tan simples, los cuales por su propia naturaleza no pueden ser abordados mediante un procedimiento de búsqueda exhaustivo que obtenga la mejor solución tras una búsqueda por todo el dominio de posibles valores (Molina Cabrera 2007).

Debido a la gran complejidad que estos problemas han alcanzado, en la mayoría de las veces, la aplicación de un método exacto para su solución, se ha vuelto impracticable. Esto ha proporcionado el auge de los métodos heurísticos, vocablo este, que surge del griego *heuriskein*, que significa encontrar. Un método o procedimiento heurístico pudiera definirse entonces como (Martínez, Puris et al. 2006):

Un método heurístico es un procedimiento para resolver un problema complejo de optimización mediante una aproximación intuitiva, en la que la estructura del problema se utiliza de forma inteligente para obtener una buena solución de manera eficiente.

Una de las principales desventajas de estos métodos, es que dependen en gran medida del problema para el cual fueron diseñados. Por esta razón surge un grupo de métodos bajo



el nombre de meta heurísticas, con el objetivo de obtener mejores soluciones que los heurísticos tradicionales para un mayor número de problemas con solo realizársele pequeñas modificaciones. En (Glover and Kochenberger 2003) se definen los procedimientos meta heurísticos como:

Los procedimientos meta heurísticos son una clase de métodos aproximados que están diseñados para resolver problemas complejos de optimización, en los que los heurísticos clásicos no son efectivos. Las metas heurísticas proporcionan un marco general para crear nuevos algoritmos híbridos combinando diferentes conceptos derivados de la inteligencia artificial, la evolución biológica y los mecanismos estadísticos.

Numerosos son los procedimientos meta heurísticos creados, sobre todo a partir de la formulación del así denominado Teorema de No Free Lunch (Wolpert and Macready 1997) que dio pie a una suerte de catarsis, a partir de la cual quedó definitivamente claro que un algoritmo de búsqueda u optimización se comporta en estricta concordancia con la cantidad y calidad del conocimiento específico del problema que incorpora, o sea que no existe un único algoritmo capaz de resolver con la mayor calidad todo tipo de problemas.

Dentro de las tantas clasificaciones que a lo largo de los años se les han dado a las meta heurísticas, una de las más comunes es subdividir las en: métodos de búsqueda local o meta heurísticas de trayectoria simple y meta heurísticas poblacionales.

Los métodos de Búsqueda Local (BL), se van a enfocar en la explotación de las soluciones ya encontradas, o sea, realizar un proceso de búsqueda en las vecindades de una solución, denominada solución actual, que en un inicio se pudiera generar de manera aleatoria, sufriendo un proceso de transformación en el cual se obtiene una nueva solución que reemplazará a la actual si mejora a esta siguiendo algún criterio de comparación. Este proceso se repetirá hasta que se generen un número máximo de soluciones o se alcanza un óptimo local.

Gran cantidad de métodos de Búsqueda Local han sido desarrollados, podemos citar por ejemplo: los métodos de Cuasi-Newton (Luenberguer 1989) entre los cuales encontramos L-BFGS y L-BFGS-B, la muy utilizada Búsqueda Tabú (Glover and Laguna



1997), el Método Solis Wets (Solis and Wets 1981), Método Simplex de Nelder y Mead (Spendley, Hext et al. 1962).

Dichos métodos son muy precisos y de rápida convergencia, pudiendo caer estancados en óptimos locales y no alcanzando en ocasiones el óptimo global que es lo que realmente se pretende. Esto es conocido como el fenómeno de la convergencia prematura.

En contra parte están los algoritmos poblacionales APs. Estos emplean un conjunto de soluciones (población) en cada iteración del algoritmo en lugar de utilizar una única solución como las meta heurísticas del grupo anterior, por lo que su búsqueda realiza el énfasis en el fomento de la exploración. Estas proporcionan de forma intrínseca un mecanismo de exploración paralelo al proceso de búsqueda de soluciones, y su eficacia depende en gran medida de cómo se manipule dicha población. Dentro de esta clasificación se destacan los Algoritmos Evolutivos (AEs) y los algoritmos basados en Inteligencia Colectiva.

Dentro de este subgrupo podemos citar entonces los Algoritmos Genéticos (AGs) (Goldberg 1998), la Búsqueda Dispersa (Laguna and Martí 2003), los Algoritmos Basados en estimación de Distribuciones (Estimation of Distribution Algorithms, EDA) (Lozano, Larrañaga et al. 2006), la Optimización basada en colonia de hormigas (Ant Colony Optimization; ACO) (Dorigo and Caro 1999) y los Sistemas de Partículas (Particle Swarm Optimization, PSO) (Kennedy and Eberhart 1995), Agujero Negro (Hatamlou 2013), Mallas Variables de su inglés (*Variable Mesh Optimization*) (Puris, Bello et al. 2012) entre otras.

En la práctica a la hora de resolver un problema de optimización ha quedado demostrado, que para obtener buenos resultados, es necesario que el algoritmo que se emplee utilice de manera equilibrada la exploración del espacio de búsqueda y la explotación de las soluciones ya encontradas. Partiendo de esto, surgen los denominados algoritmos híbridos AHs, los cuales combinan estos mecanismos.

Las ventajas del uso de este tipo de modelos están dadas por la unión de las facilidades que brindan el empleo de los algoritmos poblacionales y los métodos de BL pues pueden alcanzar mejores resultados en términos de precisión, ya que el uso de la BL permite

mejorar rápidamente la precisión, con lo que se introducirían mejores soluciones. Por otra parte puede producirse un incremento de la velocidad de convergencia partiendo de que en un AP la explotación es realizada principalmente durante las últimas generaciones, en un AH se realiza durante todo el proceso (mediante la aplicación de la BL).

Números modelos híbridos han sido presentados por diversos autores, podemos citar por ejemplo: Algoritmo híbrido con múltiples colonias (ACOR) (Herrera, Cardoso et al. 2009), *Genetic_Local_Search* (Aarts 1997), Algoritmo Memético Basado en Encadenamiento de Búsquedas Locales para Problemas de Optimización Continua (Herrera, Molina Cabrera et al. 2009), (AMCR-BLC²P) (Molina Cabrera 2007), etc.

Pero no solo se necesita una componente de búsqueda local para la explotación y una de búsqueda global para la exploración, la clave del éxito de estos AHs radica en combinar dichos factores de la manera más eficiente posible. Para ello surgen determinadas estrategias inteligentes que recopilan información útil a la hora de decidir cuándo y dónde aplicar una componente u otra.

Estrategias para ello existen muchas. La Búsqueda por Agrupamiento Evolutivo de su inglés *Evolutionary Clustering Search (ECS)* (Oliveira and Lorena 2007; Chaves and Lorena 2010), se basa en la detección de zonas promisorias en el espacio de búsqueda. Con este enfoque, un proceso iterativo de *clustering* o agrupamiento trabaja simultáneamente con el algoritmo poblacional teniendo en cuenta la actividad (selección o actualización) en las áreas de búsqueda y la identificación de cuál de ellas merece una atención especial, dicha estrategia es la empleada en el algoritmo híbrido AH_VMO-ECS desarrollado por el grupo de Inteligencia Artificial de la Universidad Central “Marta Abreu” de Las Villas, el cual no se ha implementado ni validado con el objetivo de conocer si las soluciones que obtiene muestran una verdadera mejoría de las soluciones estadísticamente hablando con respecto a las que obtiene la meta heurística VMO (también desarrollada por el grupo) por si sola o con BL aplicada a priori.



Siendo nuestro **problema de investigación** el siguiente:

- ¿Cómo encontrar diferentes alternativas, en la implementación del modelo híbrido basado en VMO y ECS, para obtener soluciones que superen en cuanto a calidad a las proporcionadas por el modelo VMO en su forma original y modelos híbridos basados en VMO con BL aplicada a priori en dominios continuos?

En consecuencia con nuestro problema pueden plantearse las siguientes **preguntas de investigación**:

- ¿Podrán encontrarse configuraciones para los parámetros del algoritmo híbrido basado en VMO y ECS, luego de ser implementado, que hagan que este brinde soluciones de mejor calidad que el modelo VMO puro y modelos híbridos basados en VMO con BL aplicada a priori?
- ¿Si fuesen logradas estas mejoras, será la diferencia con el modelo VMO puro y modelos híbridos basados en VMO con BL aplicada a priori estadísticamente significativa?

Visto lo anterior el **objetivo general** del trabajo será:

- Implementar y validar el modelo híbrido basado en VMO con ECS como estrategia de hibridación para dominios continuos.

Para ganar en claridad durante el proceso investigativo, desglosamos este objetivo general en los siguientes **objetivos específicos**:

- Implementar el modelo híbrido basado en VMO y ECS.
- Obtener distintas alternativas del modelo teniendo en cuenta diversos factores como intensidad, profundidad y características de los nodos involucrados en la BL.
- Analizar el comportamiento del modelo híbrido frente al modelo VMO puro y modelos híbridos basados en VMO con BL aplicada a priori.



Luego de haber construido el marco teórico se formuló la siguiente **hipótesis de investigación**:

El modelo híbrido basado en VMO y ECS como estrategia para detectar zonas promisorias del espacio de soluciones, aporta resultados de mayor calidad respecto al modelo VMO puro y modelos híbridos basados en VMO con BL aplicada a priori.

Dicha hipótesis será validada si se comprueba que las soluciones obtenidas por el modelo híbrido presentado, difieren, mostrando mejoría, a las del VMO puro y modelos híbridos basados en VMO con BL aplicada a priori.

Esta memoria se compone de un capítulo 1 en el cual se exponen las componentes fundamentales del modelo híbrido basado en VMO y ECS. Un capítulo 2 en el que se aborda de manera específica el modelo híbrido en sí y sus distintas alternativas y un capítulo 3 en el cual se realizan análisis estadísticos con el fin de validar el modelo propuesto.

Capítulo 1
Componentes para la hibridación

1 COMPONENTES PARA LA HIBRIDACIÓN

A la hora de conformar un Algoritmo Híbrido, los principales elementos a tener en cuenta son: el Algoritmo Poblacional que se va a encargar de la realización de la búsqueda global, la estrategia de Búsqueda Local y el mecanismo de hibridación que los combine.

No es un secreto que la componente esencial de estos algoritmos híbridos es el algoritmo poblacional pues con las soluciones generadas por este van a trabajar las demás componentes, de ellos tenemos que son meta heurísticas que emplean un conjunto de soluciones en cada iteración del algoritmo, estas proporcionan de forma intrínseca un mecanismo de exploración paralelo al proceso de búsqueda en el espacio de soluciones, y su eficacia depende en gran medida de cómo se manipule dicha población.

Dentro de este tipo de meta heurísticas se destacan dos grandes grupos, los Algoritmos Evolutivos (AEs) y los algoritmos basados en Inteligencia Colectiva.

Ambos tienen en común el haber sido inspirados en algún proceso natural, en el primero de los casos en la teoría de la evolución de Darwin (Darwin 1859), quien planteó que la evolución de las especies se produce por tres conceptos: replicación, variación y selección natural mientras que los algoritmos basados en Inteligencia Colectiva toman su inspiración en ejemplos biológicos de comportamiento colectivo (enjambres) como es el caso de las colonias de insectos, las bandadas de aves y los cardúmenes de peces.

Los Algoritmos Evolutivos son métodos de optimización y búsqueda de soluciones basados en los postulados de la evolución biológica. En ellos se mantiene un conjunto de entidades que representan posibles soluciones, las cuales se mezclan, y compiten entre sí, de tal manera que las más aptas son capaces de prevalecer a lo largo del tiempo, evolucionando hacia mejores soluciones cada vez. Numerosos han sido los modelos de AEs propuestos, entre ellos podemos destacar: Algoritmos Genéticos (AGs) (Goldberg 1998), Búsqueda Dispersa (Laguna and Martí 2003), Algoritmos Basados en estimación de Distribuciones (*Estimation of Distribution Algorithms, EDA*) (Lozano, Larrañaga et al. 2006), Evolución Diferencial (Storn and Price 1997), Búsqueda por re-encadenamiento de camino (Jourdan (2009)), Algoritmos Bionómicos (Lawler 1976), etc.



Por otra parte, la principal característica de los algoritmos basados en la Inteligencia Colectiva viene determinada por la estrecha colaboración social que presentan a través del sistema de comunicación que surge entre los individuos del grupo. Esta comunicación, a su vez, puede aparecer de forma directa o indirecta. La comunicación indirecta ocurre cuando un individuo altera el medio en que se desarrollan y los otros son capaces de captar estos cambios. La comunicación directa es aquella que ocurre a través de la obtención de la ubicación de otros individuos mediante sonido, visibilidad u otra forma directa de interacción. Entre los algoritmos más significativos y estudiados de Inteligencia Colectiva están: Sistemas de Partículas (*Particle Swarm Optimization, PSO*) (Kennedy and Eberhart 1995), Optimización basada en colonia de hormigas (*Ant Colony Optimization; ACO*) (Dorigo and Caro 1999), Búsqueda por ruido (Lourenço (2003)) y Mallas Variables (VMO) (Puris, Bello et al. 2012) meta heurística desarrollada por el grupo de Inteligencia Artificial de la Universidad Central “Marta Abreu” de las Villas, la cual se ha utilizado en nuestra investigación.

Precisamente este capítulo comienza con la descripción del VMO, luego se presentan los métodos de BL empleados y posteriormente el mecanismo de hibridación utilizado en nuestro diseño.

1.1 Optimización Basada en Mallas Variables

La Optimización Basada en Mallas Variables (*Variable Mesh Optimization; VMO*) es una meta heurística poblacional con características evolutivas donde un conjunto de nodos que representan soluciones potenciales a un problema de optimización, forman una malla (población) que dinámicamente crece y se desplaza por el espacio de búsqueda (evoluciona). Para ello, se realiza un proceso de expansión en cada ciclo, donde se generan nuevos nodos en dirección a los extremos locales (nodos de la malla con mejor calidad en distintas vecindades) y el extremo global (nodo obtenido de mejor calidad en todo el proceso desarrollado); así como a partir de los nodos fronteras de la malla. Luego se realiza un proceso de contracción de la malla, donde los mejores nodos resultantes en cada iteración son seleccionados como malla inicial para la iteración siguiente. La formulación

general de la meta heurística abarca tanto los problemas de optimización continuos como los discretos, en nuestro caso realizaremos énfasis en los problemas continuos. En (Puris 2009) se da una descripción detallada de la misma.

1.1.1 Descripción general de la meta heurística

La esencia del método VMO es crear una malla de nodos en el espacio m -dimensional, donde se realiza el proceso de optimización de una función $FO(x_1, x_2, \dots, x_m)$; la cual se mueve mediante un proceso de expansión hacia otras regiones del espacio de búsqueda. Dicha malla se hace más “fina” en aquellas zonas que parecen ser más promisorias. Es variable en el sentido que la malla cambia su tamaño (cantidad de nodos) y configuración durante el proceso de búsqueda. Los nodos se representan como vectores de la forma $n(x_1, x_2, \dots, x_m)$.

El proceso de generación de nodos en cada ciclo comprende los pasos siguientes:

- Generación de la malla inicial.
- Generación de nodos en dirección a los extremos locales.
- Generación de nodos en dirección al extremo global.
- Generación de nodos a partir de las fronteras de la malla.

El método incluye los parámetros:

- Cantidad de nodos de la malla inicial (S_{IM}).
- Cantidad máxima de nodos de la malla en cada ciclo (S_{TM} , donde $3 \cdot S_{IM} \leq S_{TM}$).
- Tamaño de la vecindad (k).
- Condición de parada (M).

1.1.2 Proceso de expansión

El algoritmo realiza un proceso de expansión mediante el cual la población se traslada a través del espacio de soluciones. Esta operación se realiza siguiendo los pasos descritos a continuación (Puris, Bello et al. 2012).

Paso 1: Generación aleatoria de la malla inicial: Para cada uno de los nodos n_i de la malla inicial se valoriza aleatoriamente cada dimensión con un valor real entre el intervalo definido en cada caso.

Paso 2: Generación de nodos en dirección a los extremos locales: El primer tipo de exploración del modelo se basa en llevar a cabo un análisis de las vecindades para cada uno de los nodos de la malla. La vecindad de n_i esta definida por los k vecinos más cercanos a este (teniendo en cuenta la distancia). El mejor nodo dentro de esta vecindad es seleccionado como extremo local n_i^* . Únicamente es generado un nodo entre n_i y n_i^* si n_i^* posee mejor valor de la función objetivo que n_i . Para calcular los vecinos más cercanos de cada nodo de la malla se utiliza como función de distancia la euclidiana, definida por (1.1):

$$D_{euclidiana}(n_1, n_2) = \sqrt{\sum_{i=1}^M (v_i^1 - v_i^2)^2} \quad (1.1)$$

Los nuevos nodos (n_z) son calculados usando la función definida por (1.2):

$$n_z = F(n_i, n_i^*, Pr(n_i, n_i^*)) \quad (1.2)$$

donde Pr es el factor de cercanía y representa la relación entre la aptitud del nodo actual y su extremo local. Este factor se calcula por la ecuación (2.3). Se toma un valor en el intervalo $[0, 1]$, más grande cuando tiene una mejor calidad n_i :

$$Pr(n_i, n_i^*) = \frac{1}{1 + |fitness(n_i) - fitness(n_i^*)|} \quad (1.3)$$

La función F puede ser descrita de diferentes maneras. En este trabajo la componente v_j^z del nodo n_z será calculada por (1.4):

$$v_j^z = \begin{cases} \bar{m}_j & \text{if } |\bar{m}_j - v_j^{i*}| > \xi_j \text{ and } U[0,1] \leq Pr(n_i, n_i^*) \\ v_j^{i*} + U[-\xi_j, \xi_j] & \text{if } |\bar{m}_j - v_j^{i*}| \leq \xi_j \\ U[v_j^i, \bar{m}_j] & \text{othercase} \end{cases} \quad (1.4)$$

donde $\bar{m} = \text{average}(v_j^i, v_j^{i*})$, $U[x, y]$ denota un valor aleatorio (uniforme) en el intervalo $[x, y]$, y ξ_j define el distancia mínima permitida para cada componente. Su valor disminuye durante el funcionamiento del algoritmo, calculado por (1.5):

$$\xi_j = \begin{cases} \frac{\text{rango}(a_j, b_j)}{4} & \text{si } 0.15\% C \\ \frac{\text{rango}(a_j, b_j)}{8} & \text{si } 0.15\% C \leq c < 0.3\% C \\ \frac{\text{rango}(a_j, b_j)}{16} & \text{si } 0.3\% C \leq c < 0.6\% C \\ \frac{\text{rango}(a_j, b_j)}{50} & \text{si } 0.6\% C \leq c < 0.8\% C \\ \frac{\text{rango}(a_j, b_j)}{100} & c \geq 0.8\% C \end{cases} \quad (1.5)$$

donde C y c denotan el número máximo de evaluaciones en la función objetivo permitidas y el número actual de evaluaciones respectivamente. Además, el $\text{range}(a_j, b_j)$ denota el dominio de amplitud (a_j, b_j) para cada componente.

La función F se comporta de la siguiente manera: en el primer caso, el valor medio entre el nodo actual y el extremo local es obtenido por la j -ésima componente. En segundo caso, la vecindad del extremo local se desplaza en función de un valor de distancia para la iteración actual. En último caso, es generado un número aleatorio entre el valor medio y el nodo actual.

Paso 3: Generación de nodos en dirección al extremo global: Con el objetivo de acelerar la convergencia se realiza un exploración en dirección al nodo con mejor valor de la función objetivo perteneciente a la población actual denominado extremo global n_g . Los nuevos nodos en dirección a n_g son generados usando al ecuación (1.6):

$$n_g^*(x_j) = G(n_i, n_g, \text{Pr}(n_i, n_g)) \quad (1.6)$$

donde Pr es el factor de cercanía entre el fitness del nodo actual y el extremo. Toma valores entre 0 y 1 y se calcula mediante la ecuación:

$$Pr(n_i, n_g) = \frac{1}{1 + |fitness(n_i) - fitness(n_g)|} \quad (1.7)$$

La función G es representada de la siguiente manera (1.8):

$$v_j^x = \begin{cases} ave(v_j^i, v_j^g), & \text{si } U[0,1] \leq Pr(n_i, n_g) \\ U[ave(v_j^i, v_j^g), v_j^g] & \text{en otro caso} \end{cases} \quad (1.8)$$

Paso 4: Generación de nodos a partir de los nodos más externos de la malla: Con este paso se completa el proceso de expansión. Solo es aplicado si el total de nodos creados en los pasos anteriores es menor que T .

En el paso 4 son considerados los nodos más internos n_u y externos n_s de la malla que conforman la frontera interior y exterior respectivamente a partir del punto que representa el centro del espacio de soluciones. Para detectar esos nodos es empleada la distancia euclidiana, los nodos de mayor distancia conformarán n_s y los de menor distancia n_u . Partiendo de esos conjuntos, nuevos nodos son creados (uno por cada nodo de la frontera) empleando las ecuaciones 1.9 y 1.10

$$v_j^h = \begin{cases} v_j^s + w_j & \text{si } v_j^s > 0 \\ v_j^s - w_j & \text{si } v_j^s < 0 \end{cases} \quad (1.9)$$

$$v_j^h = \begin{cases} |v_j^u + w_j| & \text{si } v_j^u > 0 \\ |v_j^u - w_j| & \text{si } v_j^u \geq 0 \end{cases} \quad (1.10)$$

donde w_j representa un desplazamiento para cada una de las componentes y se calcula de manera decreciente de acuerdo con la ecuación (1.11):

$$w_j = (w_j^0 - w_j^1) * \frac{C-c}{C} + w_j^1 \quad (1.11)$$

C y c denotan un máximo de evaluaciones permitidas de la función objetiva y el número de la evaluación actual respectivamente, por otra parte w_j^0 representa el desplazamiento inicial y w_j^1 su valor final. Para conseguir el decrecimiento $w_j^0 > w_j^1$ sus valores son calculados como $w_j^0 = rango(a_j, b_j)/10$ y $w_j^1 = rango(a_j, b_j)/100$. Donde $rango(a_j, b_j)$ representa la amplitud del dominio de (a_j, b_j) .

1.1.3 Proceso de contracción

Con la operación de contracción son seleccionados los nodos de la población que serán empleados en la siguiente iteración del modelo. Nodos con mejor valor de la función objetivo son seleccionados entre los pertenecientes a la malla actual y los nuevos nodos creados en el proceso de expansión. Luego de la selección un método para incrementar la diversidad poblacional es aplicado para preservar un mínimo de distancia entre los nodos de la malla. Para conseguir esto un operador de limpieza adaptativo es propuesto. Seguidamente se brinda una descripción más detallada de dicho proceso:

Paso 5: Los nodos de la malla son ordenados de acuerdo con el valor de su *fitness* (ascendente).

Paso 6: La diferencia entre cada nodo y su sucesor es calculada. Sucesores con una diferencia menor el ζ calculado, son eliminados de la malla. ζ es calculado por la ecuación 1.12:

$$\zeta = \begin{cases} \frac{\text{rango}(a_j, b_j)}{4} & \text{si } 0.15\% C \\ \frac{\text{rango}(a_j, b_j)}{8} & \text{si } 0.15\% C \leq c < 0.3\% C \\ \frac{\text{rango}(a_j, b_j)}{16} & \text{si } 0.3\% C \leq c < 0.6\% C \\ \frac{\text{rango}(a_j, b_j)}{50} & \text{si } 0.6\% C \leq c < 0.8\% C \\ \frac{\text{rango}(a_j, b_j)}{100} & c \geq 0.8\% C \end{cases} \quad (1.12)$$

Paso 7: Los nodos con mejor *fitness* son seleccionados para la siguiente iteración del algoritmo. Si el número de nodos necesarios para la realización de una nueva iteración de la malla no es alcanzado, esta se completa con nodos generados de manera aleatoria.

A continuación se presenta un esquema que muestra la estructura general del algoritmo tomando como condición de parada un número máximo de evaluaciones de la función objetivo (Figura 1.1):

Generación de la malla inicial (N_i) de forma aleatoria.

Evaluar los nodos de la malla inicial, seleccionar el mejor n_g .

Repetir:

Para cada nodo en la malla inicial del ciclo **hacer**

Encontrar sus k nodos más cercanos.

Determinar el mejor de los vecinos n_l .

Si n_l es mejor que el nodo actual, **entonces**

Generar nuevo nodo usando la ecuación. 1.3

Fin Si

Fin Para

Para cada nodo en la malla inicial del ciclo **hacer**

Generar un nuevo nodo usando la ecuación 1.6

Fin Para

Seleccionar los nodos más externos de la malla.

Generar nuevo nodo usando las ecuaciones 1.9 y 1.10

Mientras los N nodos de la malla en el ciclo actual **no se hayan generado**

Generar nuevo nodo aleatoriamente.

Construir la malla inicial actual de manera elitista.

Ordenar los nodos de la malla inicial según su calidad.

Aplicar operador de limpieza adaptativo.

Hasta M evaluaciones

Figura 1.1 Esquema general de modelo VMO

1.2 Métodos de Búsqueda Local Empleados

En el diseño de un AH es necesario el uso de una componente que se encargue de la explotación de las soluciones ya encontradas.

Los métodos de Búsqueda Local BL o meta heurísticas de trayectoria simple, definen un proceso de transformación que parte de una solución inicial generada de forma aleatoria o mediante algún otro algoritmo la cual constituye la solución actual. A esta solución se le aplican dichos métodos para obtener un conjunto de soluciones en un entorno cercano. O

sea mediante un proceso iterativo dada una solución actual buscan una mejor solución en su vecindario, dicha mejor solución pasa a ser la solución actual y se repite el proceso hasta que se generan un número máximo de soluciones o se alcanza un óptimo local.

A su vez existen, dentro de las propias estrategias de búsqueda local, dos subgrupos bien definidos: por un lado los métodos de BL deterministas los cuales no incluyen ningún tipo de aleatoriedad, es decir que, partiendo de una misma solución inicial y habiendo transitado un mismo número de iteraciones siempre se llegará a una misma solución. Cuando un modelo de BL determinista es combinado con otro método de búsqueda de manera que es empleado trascurrido cierto tiempo, se aplica el concepto de memoria de la búsqueda (almacenar, se haya mejorado o no, la solución final luego de haber aplicado BL a una determinada solución inicial cierto número de veces) para no malgastar tiempo y evaluaciones de la función objetivo. Por otro lado están los métodos estocásticos, ya acá es incorporada cierta aleatoriedad, por lo que, aunque en ocasiones es empleada, la memoria de la búsqueda es menos aplicada. Ejemplos de estos métodos son: Métodos de Cuasi-Newton (Luenberguer 1989), Búsqueda Local Guiada (Voudouris and E.Tsang 1995), Búsqueda Tabú (Glover and Laguna 1997), Método Solis Wets (Solis and Wets 1981), Método Simplex de Nelder y Mead (Spendley, Hext et al. 1962), etc.

A continuación mostramos una descripción más detallada de los métodos de Búsqueda Local (BL) empleados en nuestro modelo, son estos los anteriormente mencionados: el método Simplex de Nelder y Mead (Nelder and Mead 1965) y el método Solis y Wets (Solis and Wets 1981). Ambos pertenecen al grupo de los métodos estocásticos y siguen un proceso iterativo en el que en cada paso se genera una nueva solución a partir de una solución actual que reemplazará a esta en caso de que la mejore, pero se diferencian en la estrategia que emplean para ir obteniendo las nuevas soluciones. En ambos casos el proceso de búsqueda se repite hasta que se cumpla determinada condición de parada que en nuestro caso será un número máximo de evaluaciones de la función objetivo.

1.2.1 Método Solis Wets

Este método de BL sigue un esquema de ascensión de colinas con un tamaño de salto adaptativo. En cada iteración se parte de una solución actual x . Se genera un valor d mediante una distribución normal con 0 de media y ρ de desviación estándar. La nueva solución se obtiene sumando dicho valor d junto con un valor *bias* que mantiene un cierto grado de “inercia” sobre los movimientos anteriores. Este método se caracteriza en varios aspectos:

- Las nuevas soluciones se obtienen mediante la suma de una variable de incremento (generada mediante una distribución normal $N(0, \rho)$ y una variable *bias*. Esta variable *bias* permite mantener una inercia (orientando la búsqueda hacia direcciones exitosas en iteraciones anteriores).
- En cada paso se considera una dirección y si el resultado no es mejorado, se toma como nueva dirección de búsqueda, el sentido contrario a la dirección empleada hasta ese momento.
- Cuenta el número de éxitos (soluciones que mejoran a la actual) o fallos (soluciones peores que la actual) consecutivos. En función de unos valores máximos, el parámetro ρ es incrementado o decrementado, para aumentar o disminuir el espacio de búsqueda sobre la solución actual.

En la siguiente figura (1.2) se muestra un esquema del método Solis y Wets antes mencionado.

Método Solis y West

- a. $fitness_{actual} \leftarrow evaluación(x)$. $n_{eval} \leftarrow 1$.
- b. Inicializa ρ .
- c. $bias \leftarrow 0$.
- d. $numExitos \leftarrow 0$. $numFallos \leftarrow 0$.
- e. Mientras $n_{eval} < n_{int}$
 - 5.1 $d \sim N(0, \rho)$.
 - 5.2 $x' \leftarrow x + bias + d$.
 - 5.3 $fitness(x') \leftarrow evaluación(x')$. $n_{eval} \leftarrow n_{eval} + 1$.
 - 5.4 Si $fitness(x') < fitness_{actual}$
 - 5.4.1 $x \leftarrow x'$
 - 5.4.2 $bias \leftarrow 0.2 \cdot bias + 0.4 \cdot (d + bias)$.
 - 5.4.3 $numExitos \leftarrow numExitos + 1$. $numFallos \leftarrow 0$.
 - 5.5 EOC
 - 5.5.1 $x'' \leftarrow x - bias - d$.
 - 5.5.2 $fitness(x'') \leftarrow evaluación(x'')$. $n_{eval} \leftarrow n_{eval} + 1$.
 - 5.5.3 Si $fitness(x'') < fitness_{actual}$
 - $x \leftarrow x''$.
 - $bias \leftarrow bias - 0.4 \cdot (d + bias)$.
 - $numExitos \leftarrow numExitos + 1$. $numFallos \leftarrow 0$.
 - 5.5.4 Si no
 - $bias \leftarrow 0.5 \cdot bias$.
 - $numExitos \leftarrow 0$. $numFallos \leftarrow numFallos + 1$.
 - 5.6 Si $numExitos \geq maxExitos$
 - $numExitos \leftarrow 0$
 - $\rho \leftarrow 2\rho$
 - 5.7 Si $numFallos \geq maxFallos$
 - $numFallos \leftarrow 0$
 - $\rho \leftarrow \frac{\rho}{2}$

Figura 1.2: Esquema del método Solis-Wets de BL

1.2.2 Método Simplex de Nelder y Mead

Constituye un algoritmo de optimización ampliamente utilizado. Diseñado por Nelder y Mead (Nelder and Mead 1965) es un método para minimizar una función objetivo en un espacio multidimensional.

El método utiliza el concepto de un simplex que es una figura de $N + 1$ vértices en N dimensiones: un segmento de línea en una línea, un triángulo en un plano, un tetraedro en un espacio tridimensional y así sucesivamente.

El método busca de modo aproximado una solución óptima local a un problema con N variables. Por ejemplo, un ingeniero que diseñe un puente colgante ha de elegir el grosor de los cables laterales, los cables más largos y del soporte que irá asfaltado. Estos elementos están ligados para un correcto diseño del puente y no es fácil imaginar el efecto en el cambio de cada uno de los espesores. El ingeniero puede usar el método Nelder y Mead para generar diseños de prueba, fijando los grosores de los elementos, que son probados en un modelo de ordenador que tiene en cuenta otros parámetros (vibraciones, vientos, materiales de construcción n...). Así se introduce una función, llamémosla “inestabilidad del puente” que depende de los grosores de los elementos con los que se construye, que interesa hacer mínima ante otros factores externos (vibraciones, vientos, etc.).

Como cada vez que se ejecuta este modelo que tiene en cuenta los factores externos se consume mucho tiempo de cálculo es importante variar los grosores con idea de no malgastar recursos. El método Nelder-Mead genera una nueva posición de prueba (valor de los grosores) extrapolando el comportamiento de la función en los vértices de un simplex. Así no es necesario calcular probar todos los valores posibles de la función (todos los grosores) sino que el algoritmo va reemplazando cada vez uno de los puntos de prueba ajustando con idea para encontrar la solución que minimice la función más rápidamente. El modo más sencillo de hacerlo es reemplazar el peor punto con un punto reflejado en el resto de $N - 1$ puntos considerados como un plano (de ahí la extrapolación). Si este punto da mejor resultado, el algoritmo prueba a “estirarse” tomando los valores exponencialmente en una línea que contenga este punto. Por otra parte, si este nuevo punto

no es mucho mejor que el valor previo, entonces estamos en un valle (buscamos un mínimo, como un gran hoyo) y el algoritmo encoge el simplex hacia el mejor punto.

La naturaleza del método es puramente geométrica y, en ausencia de derivadas, la optimización se efectúa mediante la aplicación sobre el simplex de las operaciones conocidas como reflexión, expansión y contracción. Sobre el espacio de soluciones en N dimensiones, el algoritmo genera iterativamente nuevos vértices del simplex que reemplazan a los peores y hacen que éste se comprima progresivamente entorno a un mínimo o máximo en la región de búsqueda.

Reflexión, Expansión y Contracción

Cada vértice del simplex representa a una posible solución potencial al problema bajo análisis. A su vez, la precisión y calidad de cada uno de los vértices se mide en base a una función de coste o función de fitness. Cada vértice del simplex quedará representado por el vector

$$C = (M_1, F_1, \dots, M_i, F_i, \dots, M_n, F_n) \quad (1.13)$$

donde M_i , F_i representan la amplitud y fase de la excitación de cada elemento de corriente del modelo equivalente.

Una vez se han inicializado de forma aleatoria los $N+1$ puntos que conforman el simplex, el algoritmo somete los vértices a una serie de operaciones con el objetivo de reemplazar aquellos que representan a las soluciones de menor calidad por vértices más precisos. Posteriormente y en cada iteración del algoritmo se selecciona en base al fitness el mejor de los vértices, C_m , y el peor: C_{p1} y se aplican sobre ellos las operaciones de reflexión, expansión y contracción. Como resultado, de cada una de las operaciones se obtienen nuevos puntos que irán reemplazando al peor de todos los vértices en la iteración en curso.

La operación de reflexión genera un nuevo vértice C^R de la siguiente manera:

$$C^R = (1 + \alpha)C^m - \alpha C_{p1} \quad (1.14)$$

donde α es el coeficiente de reflexión y C^m es el centroide del simplex calculado excluyendo a C_{p1} . Si el Fitness en C^R mejora respecto al vértice C_{p1} entonces puede pensarse que una expansión más allá del punto de reflexión pudiera mejorar aún más el resultado. Esta situación se dará si estamos en una pendiente ascendente o descendente, según se trate de maximización o minimización, respectivamente. La expansión se rige por la siguiente expresión:

$$C^E = \beta C^R + (1 + \beta)C^M \quad (1.15)$$

donde β es el coeficiente de expansión. Si el valor del Fitness en C^E mejora el valor obtenido en C_{p1} , se sustituye C_{p1} por C^E en el simplex y se repite el proceso. En caso contrario, si la expansión conduce a una solución peor, entonces es el vértice C^R el que se utiliza para remplazar al C_{p1} .

Ahora bien, en el supuesto de que la reflexión no depare mejoría alguna, entonces se precisa aplicar una contracción sobre el peor punto, dado que lo más probable es que el análisis se esté llevando a cabo sobre un valle. La contracción se rige por la expresión (2.16), donde γ representa el denominado coeficiente de contracción.

$$C^C = \gamma C_{p1} + (1 + \gamma)C^M \quad (1.16)$$

Si aun así existe alguna situación en la que no se contempla que con la contracción se mejore respecto al vértice C_{p1} , Nelder y Mead proponen la contracción del simplex en todas las direcciones y hacia el mejor punto, lo que algunos autores llaman encogimiento hacia C^M .

Las constantes α, β, γ representan el factor con el que se incide sobre el simplex. Los valores típicos $\alpha = 1.0$, $\beta = 2.0$ y $\gamma = 0.5$ demuestran ser los más idóneos y son los propuestos por defecto en las rutinas utilizadas como referencia para implementar el método en este trabajo.

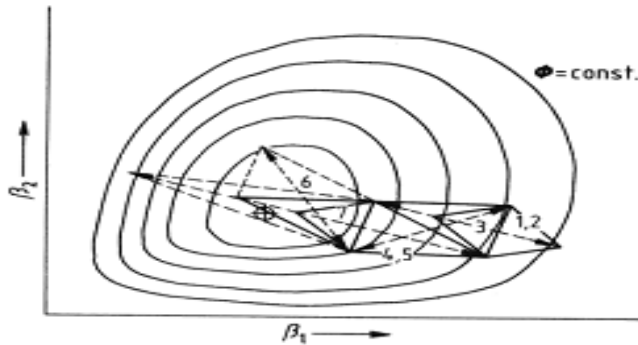


Figura 1.3: Representación del Método Simplex con geometría variable. Los números representan los números de reflexión, expansión o contracción.

Ventajas y Desventajas del método

Como ventajas podemos mencionar que es un método heurístico. Se basa en consideraciones geométricas y no requiere el uso de derivadas de la función objetivo. Es de gran eficacia incluso para ajustar gran número de parámetros. Se puede usar con funciones objetivo muy sinuosas pues en las primeras iteraciones busca el mínimo más ampliamente y evita caer en mínimos locales fácilmente. Es fácil de implementar y usar, y sin embargo tiene una alta eficacia. Por otra parte se dice que converge más lentamente que otros métodos pues requiere mayor número de iteraciones.

1.3 Fundamentos de la Búsqueda por Agrupamiento Evolutivo

La búsqueda por agrupamiento evolutivo del inglés *Evolutionary Clustering Search* (ECS) (Oliveira and Lorena 2007; Chaves and Lorena 2010) es un algoritmo iterativo que tiene como objetivo combinar meta heurísticas poblacionales y meta heurísticas de trayectoria simple o lo que es lo mismo métodos de búsqueda local. El proceso de búsqueda gana intensidad sólo en aquellas áreas del espacio de soluciones que merecen una atención especial (llamadas regiones promisorias). Dichas regiones promisorias están enmarcadas por grupos o *clusters*, que pueden ser definidos como una tupla $G = \{c, r, s\}$, donde c y r son el centro y el radio de la región (*cluster*), respectivamente. El término s indica diferentes estrategias de búsqueda asociadas a los clusters. El centro de la

agrupación es una solución que representa el clúster, la identificación de su ubicación en el espacio de búsqueda. Inicialmente, los centros son obtenidos aleatoriamente, pero progresivamente, tienden a caer sobre los puntos realmente prometedores en el subespacio enmarcado. El radio denota la distancia máxima, empezando desde el centro, para que una solución pueda ser asociada a la agrupación (*cluster*). La estrategia de búsqueda es una intensificación sistemática de la búsqueda, en el que las soluciones de un grupo interactúan entre sí a lo largo del proceso de agrupación, generando nuevas soluciones.

La hibridación (Véase la figura 2.1) del proceso basado en ECS se compone de cuatro componentes independientes (Oliveira and Lorena 2004):

1. Meta heurística base para la búsqueda (MB).
2. Componente de clusterización iterativa (AI).
3. Módulo de análisis (MA).
4. Estrategia de Búsqueda local (BL).

La meta heurística base funciona como un generador de soluciones a tiempo completo, de acuerdo con su estrategia de búsqueda específica, y su trabajo no depende en absoluto de ninguna de las restantes componentes, en esencia se realiza la manipulación de un conjunto de $|P|$ soluciones ($|P| > 1$ para EA). Los individuos son seleccionados, a partir de ellos se generan nuevos individuos, y son actualizados para las próximas generaciones. Todo este proceso funciona como un bucle infinito, en el que las soluciones se generan a lo largo de las iteraciones.

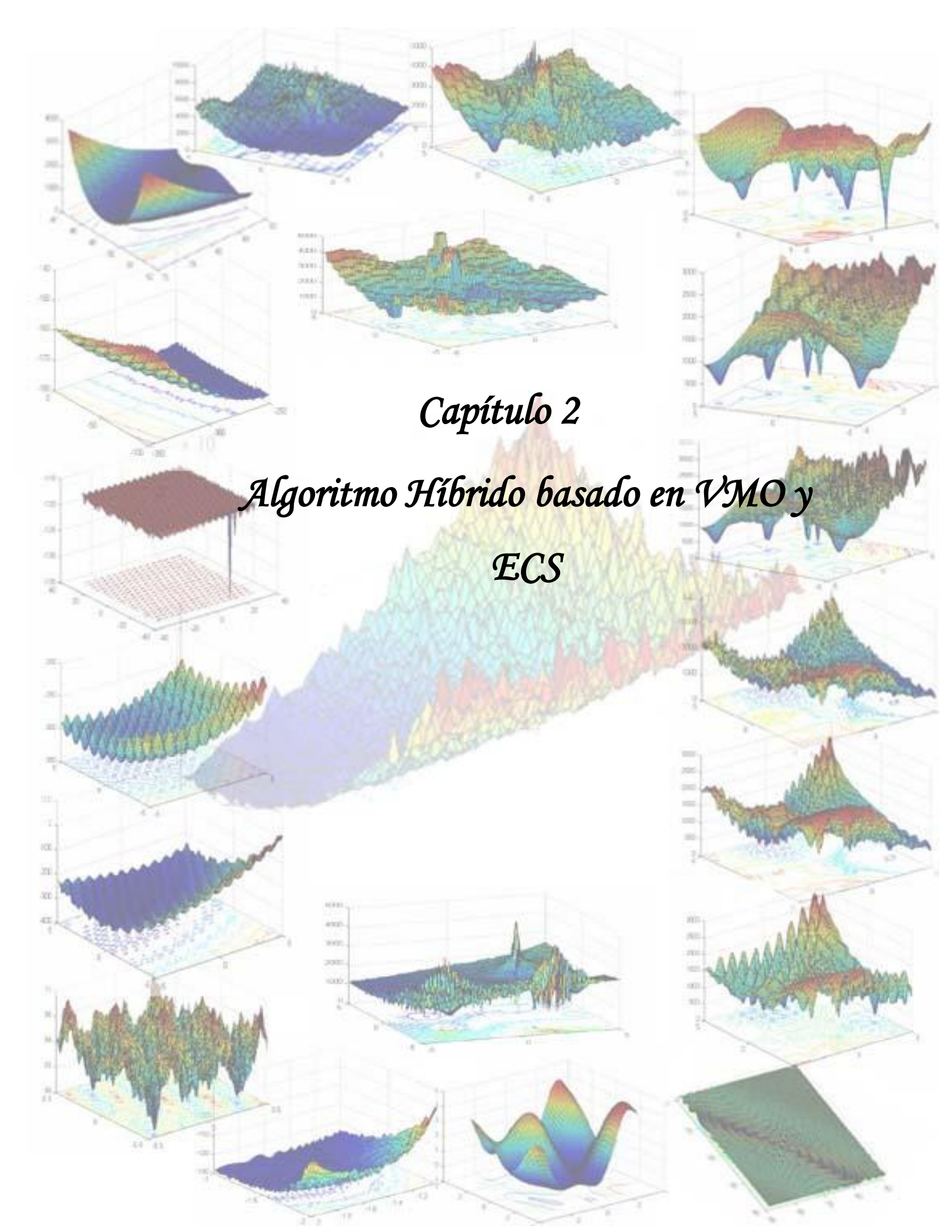
IC se encarga de agrupar soluciones similares dentro de los grupos, manteniendo en cada momento para cada uno de estos un centro como elemento representativo de los mismos. La cantidad máxima de clústeres posibles a crearse (NC) es limitada, pues se evita con esto la creación incontrolada de los mismo. Evidentemente para la creación de los mismos es necesaria una métrica de distancia ϕ , la cual defina la medida de similaridad para los *clusters*.



El módulo de análisis (AM) inspecciona cada uno de los clústeres en intervalos regulares indicando cual o cuales de estos pudieran considerarse como promisorios. La densidad del clúster δ_i , es una medida que indica el nivel de actividad dentro del clúster. Cuando δ_i alcanza determinado valor indica que existe cierta información predominante en las soluciones generadas por la meta heurística base, o sea se vuelve tentador el hecho de investigar esa zona con el objetivo de acelerar la convergencia.

Otra característica del AM es que posee un mecanismo que permite la creación de nuevos clústeres, pues elimina los clúster que posean poca actividad (δ_i bajo).

La cuarta de las componentes es la estrategia de búsqueda local, esta realizará una intensificación de la búsqueda en zonas cercanas a los centros de los *clusters* detectados por AM como promisorios.



Capítulo 2

Algoritmo Híbrido basado en VMO y ECS

2 ALGORITMO HÍBRIDO BASADO EN VMO Y ECS

En este capítulo se describe la implementación del algoritmo híbrido para resolver problemas de optimización continua sin restricciones. Este tipo de problemas de optimización que clásicamente se formaliza como:

$$\min/\max f(x), x = (x_1, x_2, x_3, \dots, x_n)^T \in R^n \quad (2.1)$$

donde x_i es la variable de decisión definida en el dominio $L_i < x_i < U_i$. Los términos L_i y U_i son los límites inferiores y superiores, respectivamente, que son definidos a priori. En concreto, en este trabajo, nos centramos en las funciones multimodales (véase, para una mejor descripción, (Suganthan, Hansen et al. 2005)), que son más difíciles que los típicos unimodales.

2.1 Algoritmo híbrido propuesto

El algoritmo híbrido propuesto en esta investigación consiste en combinar una eficaz meta heurística poblacional y dos procedimientos de búsqueda local bien conocidos mencionados en el capítulo anterior. La combinación de estas meta heurísticas es guiada por ECS, que indica cuando un grupo de soluciones enmarcan una zona promisoría y luego en el centro de tales zonas la búsqueda local es aplicada (ver fig. 2.1).

De manera general la implementación del algoritmo híbrido queda propuesta de la siguiente manera:

- Generación de la malla inicial.
- Generación de nodos en dirección a los extremos locales.
- Generación de nodos en dirección al extremo global.
- Generación de nodos a partir de las fronteras de la malla.
- Aplicación del operador de limpieza para preparar la malla inicial del próximo ciclo.
- Aplicación del proceso de agrupamiento.
- Análisis para detectar zonas promisorias en el espacio de soluciones generadas.

- Aplicación de la búsqueda local sobre el centro de las zonas promisorias.

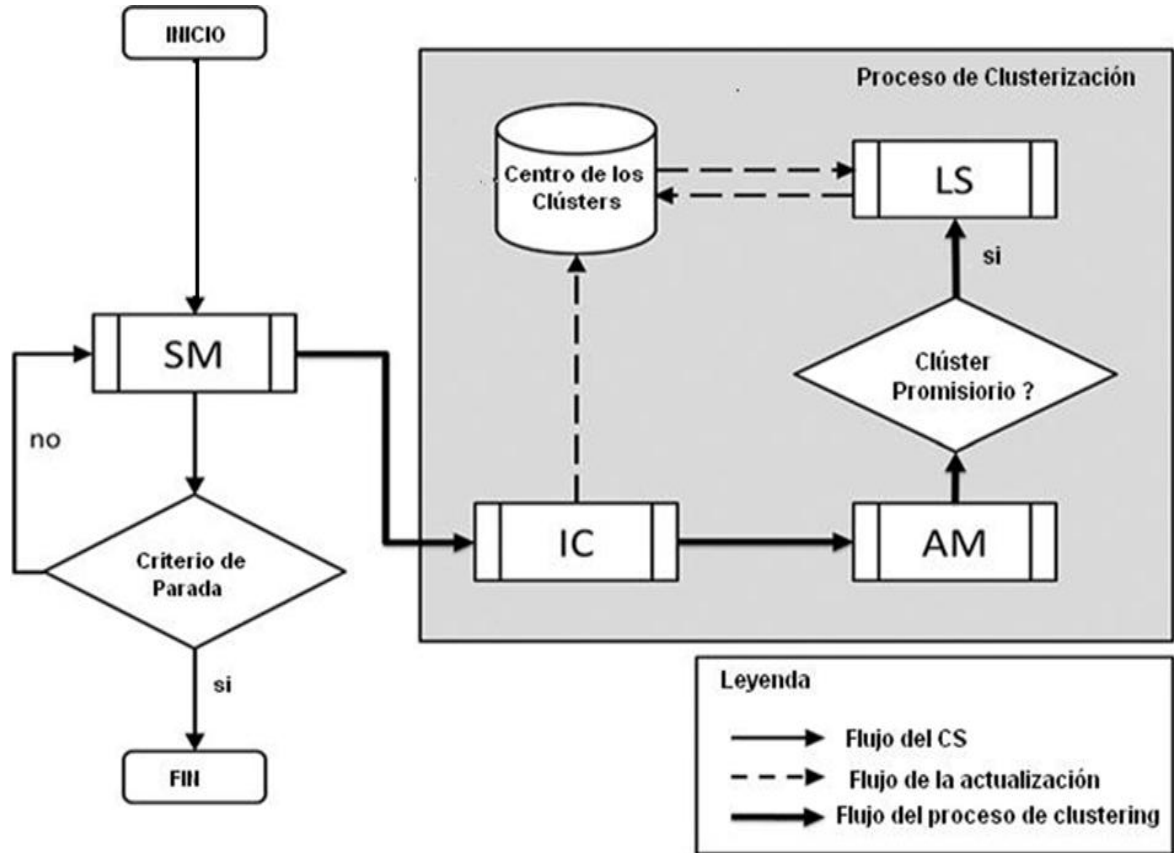


Figura 2.1 Diagrama del AH.

El modelo VMO, introducido por (Puris, Bello et al. 2012), se encarga de generar soluciones que serán agrupadas. Posteriormente, un agrupamiento iterativo es desarrollado en la componente IC. Cada i^{th} clúster tiene su propio centro. El radio es común para el clúster i en cada generación t . En esta componente, una métrica de distancia específica ϕ es definida para optimizar los parámetros de la función. El radio para todos los clusters en esta métrica de distancia está determinado por (2.2):

$$r_t = \frac{x_{sup} - x_{inf}}{F * n \sqrt{|C_t|}} \quad (2.2)$$

Donde x_{sup} y x_{inf} son, respectivamente, los conocidos límites superior e inferior del dominio de la variable. La dimensión de este problema de optimización se denota por el término n definido por el usuario, y $|C_t|$ representa el número actual de grupos (inicialmente, $|C_t| = NC$). Dependiendo del valor r_t , una solución generada por el modelo VMO podría pertenecer a varios grupos. Esta coincidente se evita cuando cada solución se inserta al clúster con mínimo valor entre su centro y la solución analizada. Debido al particular comportamiento evolutivo que presenta el modelo VMO en la generación de soluciones, el parámetro F definido por el usuario depende del número de evaluaciones alcanzado (CE). Más específicamente, se ajusta este parámetro en función de:

$$F = \begin{cases} 6 & \text{si } 0 < CE \leq C/4 \\ 10 & \text{si } C/4 < CE \leq C/2 \\ 20 & \text{si } C/2 < CE \leq 2C/3 \\ 50 & \text{si } 2C/3 < CE \leq 3C/4 \\ 100 & \text{si } 3C/4 < CE \leq C \end{cases} \quad (2.3)$$

2.2 Proceso de Clusterización o agrupamiento

El proceso de clusterización constituye la componente esencial del ECS, pues se va a encargar de clasificar y mantener en la población solamente la información relevante, guiando la intensificación de la búsqueda hacia las zonas promisorias. Esta componente es aplicada cuando la malla ha pasado un proceso de limpieza y se encuentra lista para la siguiente iteración.

Las soluciones generadas por la meta heurística base S_k son procesadas por la componente IC para intentar agruparlas teniendo en cuenta ϕ . Si una información es considerada lo suficientemente nueva, pasa a conformar el centro de un nuevo *cluster* C_{nuevo} , de no ser así la información redundante activa el centro más cercano C_i , causando cierta perturbación en este. Esta perturbación es conocida como proceso de asimilación en

el cual el centro del clúster es actualizado por la nueva información incorporada al grupo. En otras palabras, si G_j ($j = 1, 2, \dots$) son todos los *clusters* detectados un momento dado:

$$C_{nuevo} = S_k, \quad \text{si } \varphi(S_k, C_j) > r_j, \forall G_j; \text{ or}$$

$$C'_i = \text{asimilación de } S_k \text{ por } C_i, \text{ en otro caso.}$$

Agrupamiento Iterativo (IC)

```

Inicio{
  {meta heurística (VMO)}
  Para cada nodo de la malla hacer
    Si nodoActual no es centro y no pertenece a ningún cluster hacer
      Buscar el centro más cercano dentro del radio r
      Si se encuentra un centro hacer
        Asimilación del nodoActual a dicho centro
      sino
        Si # actual de clusters es menor que # máx. de clusters hacer
          nodoActual se convierte en el centro de un nuevo cluster
        sino
          Buscar el centro más cercano
          Asimilación del nodoActual a dicho centro
      Fin Si
    Fin Si
  Fin Para
}Fin
  
```

Figura 2.2 Pseudocódigo del proceso de agrupamiento.

Asimilación

El proceso de asimilación es aplicado sobre el centro más cercano C_i y considerando la nueva solución generada S_k . De manera general la asimilación puede expresarse como:

$$C'_i = C_i \oplus \beta(S_k \ominus C_i) \quad (2.4)$$

donde $\oplus$ y $\ominus$ son las operaciones abstractas sobre C_i y S_k , significando respectivamente, la adición y sustracción de las soluciones. La operación $(S_k \ominus C_i)$ representa el vector de diferencias entre cada una de las n variables que conforman S_k y C_i , teniendo en cuenta la métrica de distancia. Un porcentaje β del vector constituye el paso de actualización para el centro C_i transformándolo en C'_i . En dependencia de cómo se defina β es el tipo de asimilación empleado. En la siguiente figura se muestran los tres tipos de asimilación propuestos por los autores. Seguidamente son descritos con un mayor nivel de detalle.

2.2.1 Asimilación simple

En la asimilación simple, $\beta \in [0,1]$ es una constante aleatoria que determina cierto movimiento de C_i , centro del cluster, en dirección a S_k , solución analizada. Solo un punto es generado en el intervalo, más o menos cerca de C_i , dependiendo de β . Para un valor grande de β menos conservador se hace el movimiento. En la figura 2.4 se puede observar el pseudocódigo de este tipo de asimilación. De manera específica puede definirse como:

$$C'_i = C_i + \beta(S_k - C_i) \quad (2.5)$$

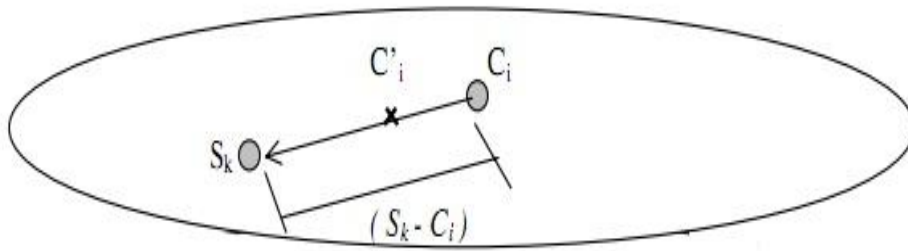


Figura 2.3 Asimilación simple (Oliveira and Lorena 2004)

Asimilación Simple

Inicio{

 $C \rightarrow$ centro actual $S \rightarrow$ solución que entra al cluster $newC \rightarrow$ nuevo centro obtenido tras la asimilaciónSe genera un $\beta \in [0,1]$ Para cada componente i del centro C hacer*Aplicar ecuación 2.5*

fin Para

}Fin

Figura 2.4. Pseudocódigo de la Asimilación simple**2.2.2 Asimilación por cruzamiento**

A pesar de su nombre, la asimilación por cruzamiento no está necesariamente asociada con el operador evolutivo, solo se refiere a una operación aleatoria sobre dos candidatos que genera otros como su descendencia. En este tipo de asimilación, β constituye un vector aleatorio n -dimensional y C'_i asume una posición aleatoria dentro del hiper plano formado por S_k y C_i . Partiendo de esto la asimilación puede reformularse como:

$$C'_i = C_i + \vec{\beta}(S_k - C_i) \quad (2.6)$$

el parámetro $\vec{\beta}$ depende del tipo de cruzamiento empleado, por ejemplo en el caso de la optimización continua un cruzamiento combinado (BLX- α) (Eshelman and Schaffer 1993) puede ser empleado en la reformulación de la ecuación (2.4), quedando finalmente $\vec{\beta}$ de la siguiente forma:

$$\vec{\beta} = -blx\alpha + \vec{\beta}'(1 + 2blx\alpha) \quad (2.7)$$

donde $\vec{\beta}' \in [0,1]$, $blx\alpha$ es el parámetro BLX- α resultando $\vec{\beta} \in [-blx\alpha, 1 + blx\alpha]$.

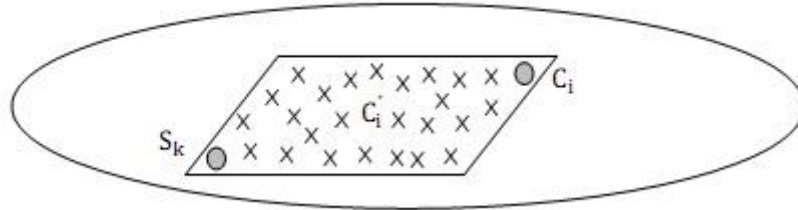


Figura 2.5 Asimilación por cruzamiento (Oliveira and Lorena 2004)

Asimilación por cruzamiento

Inicio{

$C \rightarrow$ centro actual

$S \rightarrow$ solución que entra al cluster

$newC \rightarrow$ nuevo centro obtenido tras la asimilación

Se genera $\beta' \in [0,1]$

Se genera el vector aplicando la ecuación 2.7

Para cada componente i de C hacer

 Aplicar la ecuación 2.6

fin Para

}Fin

Figura 2.6. Pseudocódigo de la Asimilación por cruzamiento

2.2.3 Asimilación por camino

En el tipo de asimilación propuesto anteriormente, solo es generado un punto interno. La asimilación por camino genera un grupo de puntos internos, los evalúa y selecciona para el nuevo centro al mejor de estos. Esta idea posee sus ventajas, pero claramente es más costosa que las otras dos antes mencionadas.

En esta asimilación, β es un vector n-dimensional de parámetros constantes y uniformemente separados, empleado para generar n muestras en el camino entre C_i y S_k . Como cada muestra es evaluada por la función objetivo, la asimilación por camino es en sí, un mecanismo de intensificación dentro del clúster. El nuevo centro se obtiene a partir de la ecuación:

$$C'_i = C'_v, f(C'_v) = \min\{f(C'_1), f(C'_2), \dots, f(C'_n)\} \quad (2.8)$$

$$f(C'_1) = C_i + \beta_j(S_k - C_i) \quad (2.9)$$

$$\beta_j \in \{\beta_1, \beta_2, \dots, \beta_n\} \quad (2.10)$$

donde $\beta_j \in \{(0,1) \cup (1, \infty]\}$ y $f(C'_v)$ es la mejor evaluación de la función objetivo para todas las muestras generadas en el camino. El intervalo infinito propuesto en (5) indica que se pueden seguir generando puntos indefinidamente más allá de S_k mientras la solución se vaya mejorando.

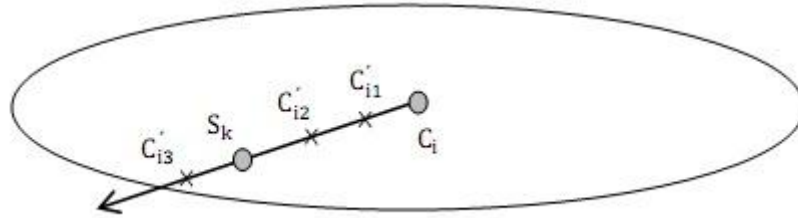


Figura 2.7 Asimilación por camino (Oliveira and Lorena 2004)

Asimilación por camino

Inicio{

C -> centro actual

S -> solución que entra al cluster

newC -> nuevo centro obtenido tras la asimilación

Mientras no sea satisfecha la condición de parada hacer

$$C'_i = \min(f(C'_1), f(C'_2), \dots, f(C'_n))$$

$$f(C'_1) = C_i + \beta_j(S_i - C_i)$$

$$\beta_j \in (\beta_1, \beta_2, \dots, \beta_n)$$

$$\beta_j \in [0,1]$$

}Fin

Fig. 2.8 Pseudocódigo de la Asimilación por camino.

2.3 Módulo de Análisis

Una vez terminado el proceso de clusterización, se pasa a analizar cada módulo existente. La componente AM lleva a cabo un análisis de todos los *clusters*. Los grupos obtenidos tienen su contabilidad de densidad en δ_i (que puede ser tomada, por ejemplo, como la cantidad de soluciones contenidas en el cluster *i*-ésimo), donde algunos o todos se vuelven inactivos y algunos o ninguno se detecta como promisorio, siendo “eliminados” posteriormente los inactivos (entiéndase como inactivos aquellos clusters en los que no ha habido actividad durante varias iteraciones) mientras en aquellos declarados como promisorios la componente LS es aplicada. Concretamente, esta componente se activa si:

$$\delta_i \geq PD * \frac{NS}{|C_t|} \quad (2.11)$$

donde *PD* denota la presión de densidad. Este parámetro permite controlar la sensibilidad de la componente AM. El término *PD* se considera como la densidad de cluster deseable más allá de la densidad normal, lo cual se obtiene si *NS* (número máximo de soluciones generadas por SM en cada iteración) fue igualmente divide a todos los grupos.

Módulo de Análisis (AM)

```
Inicio{
    {meta heurística}
    {Agrupamiento Iterativo}
    Para cada centro hacer
        Si centroActual está inactivo hacer
            ver cuantas iteraciones lleva inactivo
            Si lleva más de k iteraciones inactivo hacer
                liberar todos los nodos que pertenecen a centroActual
            fin Si
        sino
            calcular cuántos nodos pertenecen a centroActual
            Si cantidad de nodos del centroActual es  $\geq$  a patrón de densidad
                Declarar a centroActual como promisorio
            fin Si
        fin Si
    fin Para
    Para cada centro promisorio hacer
        Aplicar BL a centroActual
    fin Para
}Fin
```

Figura 2.9 Pseudocódigo del Módulo de Análisis.

2.4 Operador para fomentar la diversidad basado en *Clustering*

Como observación final en nuestro algoritmo híbrido, podemos evitar la convergencia prematura por medio de reposición estratégica, que consiste en comprobar si el extremo global persiste sin actualizarse después de un cierto número de iteraciones (5 iteraciones). Cuando este hecho sucede, el 50% de las peores soluciones que pertenecen al *cluster* analizado son perturbadas (sustituidas por otras generadas de manera aleatoria).

2.5 Alternativas del modelo híbrido basado en VMO y ECS estudiadas

Antes de conformar cualquiera de las configuraciones que serán analizadas en el capítulo siguiente es necesario hacer un ajuste de varios parámetros del algoritmo híbrido esenciales para su funcionamiento, estos son:

Parámetros del algoritmo:

- Tamaño de la malla inicial: 50 nodos.
- Número de nodos que definen la vecindad de un nodo en la malla: $k = 5$.
- Número máximo de evaluaciones en la función objetivo: $C = 100\,000$.
- Número máximo de soluciones generadas en cada iteración: $NS = 175$.
- Presión de densidad: $PD = 50$.
- Número máximo de *clusters*: $NC = 175$.

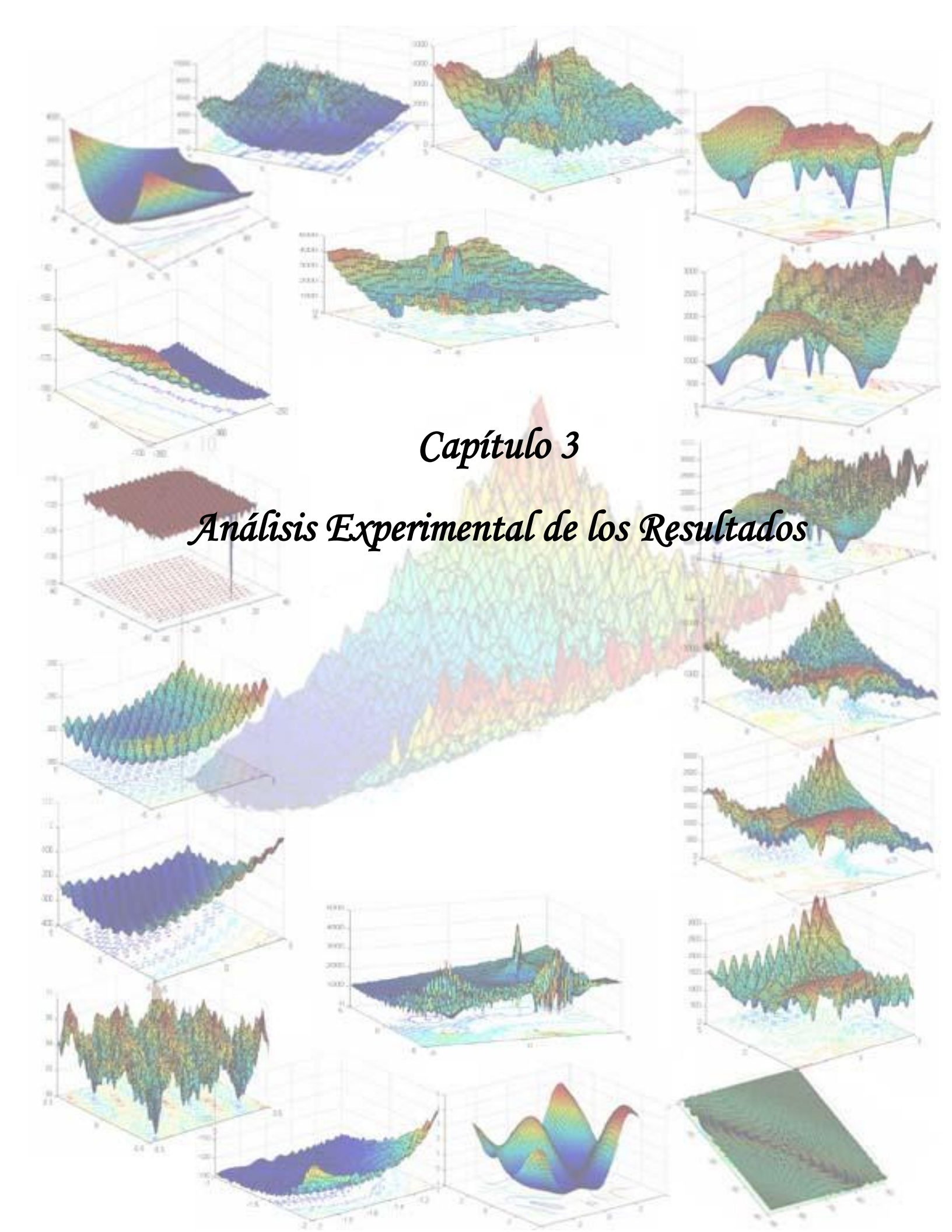
Después de realizado este ajuste se decidió construir 12 configuraciones del AH para su posterior análisis estadístico, dichas alternativas fueron implementadas teniendo en cuenta el tipo de asimilación usado en el proceso de *clustering*, el método de búsqueda local empleado y la cantidad de evaluaciones en la función objetivo utilizadas durante la etapa de búsqueda local (*NE-OF*). De lo antes expuesto resultaron los modelos presentados en la tabla 2.1. Es importante aclarar que para el análisis estadístico de las variantes, resultantes de la implementación, propuestas en esta memoria, no se tuvo en cuenta la asimilación simple pues al ser analizados los primeros resultados estos no fueron satisfactorios.

Tabla 2.1 Configuraciones del AH.

Configuración	Tipo de Asimilación	Método BL	<i>NE-OF</i>
M1	Cruzamiento	Solis-Wets	50
M2	Cruzamiento	Simplex	50
M3	Cruzamiento	Solis-Wets	25
M4	Cruzamiento	Simplex	25
M5	Cruzamiento	Solis-Wets	10



M6	Cruzamiento	Simplex	10
M7	Camino	Solis-Wets	50
M8	Camino	Simplex	50
M9	Camino	Solis-Wets	25
M10	Camino	Simplex	25
M11	Camino	Solis-Wets	10
M12	Camino	Simplex	10



Capítulo 3

Análisis Experimental de los Resultados

3 ANÁLISIS EXPERIMENTAL DE LOS RESULTADOS

En este capítulo presentamos el análisis de los resultados de las distintas variantes de diseño de nuestro AH, así como una descripción de los test estadísticos utilizados. Además nos referimos a todo el marco experimental y el proceso de experimentación desarrollado en este trabajo.

3.1 Técnicas Estadísticas

A la hora de comparar algoritmos con múltiples conjuntos de datos no existe un procedimiento ya establecido, esto se debe en gran medida a que estos mantienen un comportamiento no determinístico, por lo que la diferencia detectada entre los resultados de dos algoritmos podría deberse a factores aleatorios, y no a una mejora real.

Numerosas son las posibles técnicas a aplicar para determinar si un algoritmo difiere de forma significativa respecto a otro. Principalmente existen dos grupos:

- Los tests paramétricos (Luengo, García et al. 2009), son los más empleados. Se caracterizan en que utilizan por cada algoritmo y función el error medio alcanzado de un conjunto de ejecuciones y lo utilizan para identificar si la diferencia entre dos algoritmos es estadísticamente significativa. Un ejemplo de estos tests son el test de t-Student (para comparar dos algoritmos) y ANOVA (para comparar múltiples algoritmos).

Para poder aplicar dichos tests paramétricos con un mínimo de fiabilidad, los resultados de los algoritmos deben de seguir una distribución normal, y mantener una varianza homogénea. Cuando estas características son satisfechas los análisis paramétricos son correctos y seguros, y más sensibles (perceptivos) que los de la siguiente categoría, los tests no paramétricos.

- Los tests no-paramétricos (Sheskin 2006), que utilizan una representación ordinal de los valores basada en el orden de los algoritmos para cada uno de los problemas. Un ejemplo de estos tests son el test de Wilcoxon (Wilcoxon 1945)(para comparar los resultados de dos algoritmos) y el test de Bonferroni-Dunn o el test de Holm (Holm 1979)(para comparaciones de varios).

A lo largo de nuestra investigación haremos uso del segundo de estos grupos de pruebas, pues los resultados obtenidos por las meta heurísticas no cumplen las condiciones requeridas para poder usar de forma correcta comparaciones paramétricas, además también usaremos algunos estadísticos descriptivos.

Descripción de los tests estadísticos utilizados

En los experimentos computacionales, hemos aplicado varios métodos estadísticos. Ellos son: el cálculo del promedio, el cálculo del gap, el test de Wilcoxon (Wilcoxon 1945).

Seguidamente mostramos una descripción de las pruebas antes mencionadas teniendo en cuenta su funcionalidad básica y la utilidad de su empleo en nuestra investigación.

Cálculo de media: un cálculo simple de promedio, que permite obtener el algoritmo de mejor media en cuanto a los restantes.

Cálculo del porcentaje de desviación (gap): El rendimiento de nuestra propuesta, relacionada con la calidad de la solución, se evalúa sobre la base de una estadística clásica, porcentaje de desviación. La brecha en el porcentaje de una prueba a una función multimodal consiste en la desviación relativa entre una solución, tomada como referencia, y la propuesta de solución analizada. El gap entre un par de soluciones puede ser calculado entonces mediante la expresión x:

$$g(SA) = \frac{z_{ref} - z_{sa}}{z_{ref}} \times 100 \quad (3.1)$$

Donde z_{ref} es considerado como la solución de referencia (los resultados de VMO reportados) y z_{sa} el enfoque de solución analizada. Un valor positivo de la brecha significa que el enfoque de solución es superior a los resultados considerados como referencia.

Test de Ranking de Signos de Wilcoxon: A la hora de hallar las diferencias entre los resultados de dos algoritmos y calcular un ranking utilizando dicho valor puede emplearse este test. Se trata de una alternativa no paramétrica al t -test de muestras apareadas. Nótese que en este caso, el ranking d va desde 1 hasta N , en vez de hasta k , como era el caso de los tres tests anteriores. Tras sumar los rankings diferenciándolos entre si son negativos o positivos, se obtienen dos valores R^+ y R^- . Si el menor de ellos es menor o igual al valor de la distribución T de Wilcoxon para N grados de libertad (T), se rechaza la hipótesis nula, y el algoritmo asociado al mayor de los valores es el mejor.

3.2 Marco experimental

Como se había planteado, el conjunto de nodos involucrados, la intensidad y profundidad con que se aplique la búsqueda local pueden influir considerablemente en los resultados obtenidos. A continuación mostramos un análisis de algunas de las variantes de nuestro modelo, teniendo en cuenta estos factores, siendo analizadas con los test estadísticos mencionados en el apartado anterior. Estos análisis fueron realizados para funciones de dimensión 10, 30 y 50, salvo “1” y “2” que solo se realizaron para dimensión 10 y las configuraciones resultantes se mantuvieron en los experimentos con dimensión 30 y 50 para mantener cierto grado de coherencia en los resultados obtenidos en todas las dimensiones:

1. Hallar la variante de hibridación de menor media, entre ambos tipos de asimilaciones, cuando es seleccionado el método Solis Wets para la BL.
2. Hallar la variante de hibridación de menor media, entre ambos tipos de asimilaciones, cuando es seleccionado el método Simplex para la BL.
3. Calcular los gaps para:
 - VMO puro y “1”
 - VMO puro y “2”
 - VMO puro y VMO con BL aplicada a priori
4. Test de Wilcoxon contrastando “1” y “2”.

5. De existir diferencias significativas, la variante de mejor comportamiento se comparará mediante el test de Wilcoxon con el modelo VMO puro y el híbrido basado en VMO con BL aplicada a priori.
6. Si no existiesen dichas diferencias entre “1” y “2”, cualquiera de ellas se comparará mediante el test de Wilcoxon con el modelo VMO puro y el híbrido basado en VMO con BL aplicada a priori.

3.3 Análisis de los resultados de los test estadísticos para las distintas variantes de diseño

Para determinar las mejores configuraciones, dividimos el conjunto presentado al final del capítulo anterior en dos subgrupos teniendo en cuenta el método de búsqueda local empleado y se realizó el procedimiento estadístico antes descrito aplicado a los modelos analizados para cada una de las dimensiones (se empleó en todos los casos un enfoque de minimización). Para ver los resultados alcanzados por cada una de las configuraciones resultantes de la implementación del AH ver las tablas del Anexo A.

Dimensión 10

En el primer subgrupo se ubican las alternativas que utilizaron Solis-Wets como método de búsqueda local, ver los resultados en la tabla 3.1. En el segundo subgrupo se pueden apreciar aquellas en las que se empleó Simplex como meta heurística de trayectoria simple y también para estas se aplicó el cálculo del promedio, ver los resultados en la tabla 3.2.

Tabla 3.1 Configuraciones que emplearon Solis-Wets

Configuraciones	Media
M1	247.28
M3	243.67
M5	238.99



M7	310.69
M9	319.35
M11	310.08

Tabla 3.2 Configuraciones que emplearon Simplex

Configuraciones	Media
M2	248.08
M4	240.80
M6	239.41
M8	258.65
M10	302.89
M12	253.35

Los resultados del promedio para ambos subgrupos indican que las mejores alternativas alcanzadas por el algoritmo son M5 y M6 (ver valores en negrita).

Partiendo de lo antes visto, ambas variantes fueron seleccionadas como modelos a aplicársele el gap, contrastándolos con los resultados de VMO puro y además con un modelo híbrido, basado también en VMO y el mismo método de búsqueda local, pero en el cual la BL se aplica sin tener en cuenta ninguna estrategia. Este último análisis se realiza para probar que las mejoras en la calidad de los resultados (si se obtienen) están determinadas por la aplicación de una estrategia inteligente (*ECS*) para determinar cuándo y cómo aplicar la fase de BL. Aclarar nuevamente que con estas configuraciones seleccionadas se trabajará también en los experimentos con dimensión 30 y 50 por razones antes explicadas.

La tabla 3.3 muestra los resultados para cada una de las funciones de prueba, los valores en **negrita** indican el mejor valor de la función objetivo para cada una de las

instancias. Las dos primeras columnas representan, respectivamente, el código de la función y los resultados de VMO puro. El resto de las entradas de la tabla representan el promedio de las soluciones (*sol*, promedio de los resultados obtenidos en 25 corridas) y el gap (*g*) para las propuestas de solución analizadas. Estas propuestas son:

- VMO-LS: Modelo de AH con la fase de BL implementada por el método Simplex aplicada al mejor nodos de la malla transcurridas 10 iteraciones de VMO ($NE-OF=10$)
- Propuestas de AH basadas en ECS con mejor media seleccionados en los grupos de comparación.

Tabla 3.3 Resultados para dimensión 10.

Test function	VMO	VMO-LS		HA-M5		HA-M6	
		<i>sol</i>	<i>g</i>	<i>sol</i>	<i>g</i>	<i>sol</i>	<i>g</i>
F6	7.56E+01	4.53E+01	40.09	1.87E+01	75.25	3.45E+01	54.35
F7	1.27E+03	1.01E+03	20.70	1.88E+01	98.52	5.78E+00	99.54
F8	2.03E+01	2.04E+01	-0.14	2.03E+01	0.15	2.03E+01	0.15
F9	1.15E+01	1.21E+01	-5.74	8.68E+00	24.26	7.02E+00	38.74
F10	8.37E+00	8.17E+00	2.32	7.32E+00	12.50	8.32E+00	0.55
F11	1.96E+00	1.69E+00	13.90	1.06E+00	45.92	1.14E+00	41.84
F12	4.46E+02	3.46E+02	22.36	2.33E+02	47.72	3.03E+02	32.02
F13	1.05E+00	1.04E+00	1.50	8.31E-01	21.16	9.53E-01	9.58
F14	2.69E+00	2.75E+00	-2.32	2.02E+00	24.85	2.27E+00	15.55
F15	2.94E+02	2.59E+02	12.02	2.32E+02	21.17	1.71E+02	41.90
F16	1.10E+02	1.08E+02	1.52	1.07E+02	2.55	1.06E+02	3.46
F17	1.11E+02	1.12E+02	-0.56	1.11E+02	0.27	1.11E+02	0.27
F18	5.43E+02	6.20E+02	-14.11	5.04E+02	7.25	6.06E+02	-11.52
F19	5.65E+02	6.07E+02	-7.43	6.17E+02	-9.16	5.90E+02	-4.39
F20	6.40E+02	5.64E+02	11.87	6.24E+02	2.52	5.51E+02	13.92



F21	4.60E+02	4.92E+02	-6.96	4.36E+02	5.22	4.44E+02	3.48
F22	7.59E+02	7.66E+02	-0.87	7.57E+02	0.26	7.55E+02	0.53
F23	5.54E+02	5.66E+02	-2.14	6.46E+02	-16.59	6.24E+02	-12.62
F24	2.12E+02	2.12E+02	0.00	2.11E+02	0.47	2.36E+02	-11.32
F25	2.00E+02	2.12E+02	-6.01	2.24E+02	-12.00	2.11E+02	-5.50
Average			4.00		17.61		15.53

Los estadísticos descriptivos muestran la eficacia de la hibridación cuando esta es aplicada empleando ECS. Cuando dicha hibridación es aplicada a las 20 funciones las alternativas de hibridación M5 y M6 obtienen la mejor solución en 17 de los 20 casos. El modelo VMO puro solamente supera a los modelos M5 y M6 en 3 funciones, mientras que el híbrido VMO-LS para dimensión 10 no supera en ninguna instancia dichos AHs. Además es interesante resaltar que M5 y M6 presentan resultados para las funciones F6 y F7 que superan en el doble a la calidad de las soluciones obtenidas para las mismas funciones por el modelo VMO-PURO.

Con el objetivo de analizar si las mejoras obtenidas son estadísticamente significativas se aplica el test de Wilcoxon considerando un valor de significación de 0.05 (tabla 3.4). Además se compararon los modelo M5 y M6 para determinar el de mejor comportamiento entre ellos.

Tabla 3.4 Resultados del test de Wilcoxon para dimensión 10.

	Comparación	R^+	R^-	p -value	Hipótesis
dimensión $n = 10$	HA-M6 vs HA-M5	94.50	76.50	0.695	Aceptada
	VMO vs HA-M5	131.5	39.5	0.045	Rechazada
	VMO-LS vs HA-M6	94.5	76.5	0.010	Rechazada

La comparación entre M5 y M6 arrojó que no existen diferencias significativas entre ellos. Por otro lado, se puede comprobar que sí son estadísticamente significativas las diferencias entre los resultados obtenidos por los modelos híbridos de mejores medias y los modelos VMO puro y VMO con BL aplicada a priori.

Dimensión 30

La tabla x muestra los resultados obtenidos para esta dimensión.

Tabla 3.5 Resultados para dimensión 30

Test function	VMO	VMO-LS		HA-M5		HA-M6	
		<i>sol</i>	<i>g</i>	<i>sol</i>	<i>g</i>	<i>sol</i>	<i>g</i>
F6	2.44E+03	2.58E+03	-5.64	2.47E+03	-1.22	2.38E+03	2.50
F7	4.71E+03	4.72E+03	-0.12	2.86E+03	39.23	2.80E+03	40.59
F8	2.10E+01	2.10E+01	-0.09	2.10E+01	0.05	2.10E+01	0.12
F9	1.45E+02	1.68E+02	-15.84	1.34E+02	7.09	1.51E+02	-4.35
F10	8.67E+01	1.00E+02	-15.81	6.99E+01	19.35	6.10E+01	29.65
F11	2.31E+01	2.27E+01	1.72	2.00E+01	13.75	1.86E+01	19.64
F12	2.41E+04	2.81E+04	-16.25	1.87E+04	22.42	1.98E+04	18.02
F13	7.50E+00	1.04E+01	-39.05	8.36E+00	-11.37	8.51E+00	-13.39
F14	1.24E+01	1.23E+01	0.98	1.20E+01	3.36	1.21E+01	3.08
F15	4.46E+02	4.69E+02	-5.10	3.99E+02	10.58	3.60E+02	19.31
F16	2.94E+02	1.96E+02	33.28	1.81E+02	38.39	1.85E+02	37.11
F17	2.99E+02	2.85E+02	4.66	1.74E+02	41.73	1.70E+02	43.02
F18	9.19E+02	9.18E+02	0.02	9.01E+02	1.88	9.00E+02	2.03
F19	9.13E+02	9.18E+02	-0.56	9.00E+02	1.43	9.01E+02	1.35
F20	9.21E+02	9.18E+02	0.38	9.00E+02	2.35	9.00E+02	2.27
F21	5.92E+02	5.37E+02	9.38	5.40E+02	8.79	5.41E+02	8.56
F22	9.81E+02	9.67E+02	1.37	8.79E+02	10.35	8.87E+02	9.52



F23	6.17E+02	6.44E+02	-4.32	6.16E+02	0.22	6.57E+02	-6.46
F24	2.41E+02	2.00E+02	16.87	3.64E+02	-51.33	2.27E+02	5.60
F25	2.00E+02	2.44E+02	-21.93	4.51E+02	-125.51	3.61E+02	-80.42
Average			-2.80		1.58		6.89

Como se observa en la tabla 3.5 para dimensión 30 los modelos híbridos presentados (que emplean *ECS*) superan, en cuanto a calidad de resultados, al modelo VMO puro, este último solamente es superior en 3 de las 20 instancias. Además para esta dimensión el modelo híbrido que emplea la búsqueda local a priori presenta resultados desventajosos cuando estos son comparados con el resto de los obtenidos por las alternativas híbridas propuestas dentro del grupo.

Al aplicarse la prueba de Wilcoxon (tabla 3.6), sucede algo similar a lo ocurrido para dimensión 10: los modelos híbridos basados en *ESC* no difieren entre sí, sin embargo respecto al resto del grupo, sí lo hacen de manera positiva.

Tabla 3.6 Resultados del test de Wilcoxon para dimensión 30.

	Comparación	R^+	R^-	p -value	Hipótesis
dimensión $n = 30$	HA-M5 vs HA-M6	120.5	89.5	0.563	Aceptada
	VMO vs HA-M5	162	48	0.033	Rechazada
	VMO-LS vs HA-M6	169	41	0.017	Rechazada

Dimensión 50

Para dimensión 50 en la tabla 3.7 puede apreciarse que VMO puro solamente supera a los modelos híbridos en 1 de las instancias al igual que el modelo VMO con BL aplicada a priori. Además de lo arrojado por el estadístico descriptivo *gap* se puede constatar que el

modelo híbrido HA-M5 supera en el doble a la calidad obtenida por VMO puro en las instancias F12, F24 y F 25, mientras que HA-M4 lo hace en la instancia F24.

Tabla 3.7 Resultados para dimensión 50

Test function	VMO	VMO-LS		HA-M5		HA-M6	
		<i>sol</i>	<i>g</i>	<i>sol</i>	<i>g</i>	<i>sol</i>	<i>g</i>
F6	2.64E+03	4.17E+03	-58.04	4.23E+03	-60.37	3.66E+03	-38.53
F7	6.24E+03	6.29E+03	-0.69	4.97E+03	20.39	5.07E+03	18.74
F8	2.12E+01	2.12E+01	0.07	2.12E+01	0.04	2.12E+01	0.12
F9	4.17E+02	4.85E+02	-16.23	3.82E+02	8.37	3.77E+02	9.68
F10	2.54E+02	2.74E+02	-7.67	1.75E+02	31.12	1.65E+02	35.02
F11	5.19E+01	4.97E+01	4.19	4.55E+01	12.36	4.58E+01	11.66
F12	2.39E+05	3.14E+05	-31.54	1.02E+05	57.26	1.62E+05	32.08
F13	2.62E+01	3.32E+01	-26.99	2.44E+01	6.69	2.77E+01	-5.95
F14	2.21E+01	2.22E+01	-0.35	2.13E+01	3.91	2.12E+01	4.30
F15	5.25E+02	6.32E+02	-20.27	4.37E+02	16.80	4.36E+02	16.95
F16	2.85E+02	2.70E+02	5.08	1.97E+02	30.88	1.93E+02	32.32
F17	3.24E+02	2.65E+02	18.28	2.17E+02	33.16	2.23E+02	31.14
F18	9.86E+02	9.83E+02	0.32	9.50E+02	3.59	9.56E+02	2.98
F19	9.94E+02	9.89E+02	0.48	9.58E+02	3.60	9.55E+02	3.91
F20	9.93E+02	9.92E+02	0.06	9.56E+02	3.70	9.53E+02	3.96
F21	5.52E+02	6.79E+02	-23.07	6.03E+02	-9.22	6.51E+02	-17.88
F22	1.03E+03	1.04E+03	-1.74	9.53E+02	7.18	9.57E+02	6.80
F23	1.01E+03	9.65E+02	4.76	7.05E+02	30.41	7.53E+02	25.67
F24	7.32E+02	1.02E+03	-39.39	3.11E+02	57.53	3.51E+02	52.05
F25	7.26E+02	9.95E+02	-37.00	2.72E+02	62.56	2.36E+02	67.48
Average			-11.49		16.00		14.62



Nuevamente sucede, al aplicarse el test de Wilcoxon, algo similar. Las alternativas híbridas propuestas en nuestra investigación superan estadísticamente a las variantes VMO puro y VMO con BL aplicada a priori y no difieren entre sí (tabla 3.8).

Tabla 3.8 Resultados del test de Wilcoxon para dimensión 50.

	Comparación	R^+	R^-	p -value	Hipótesis
dimensión $n = 50$	HA-M6 vs HA-M5	127	83	0.411	Aceptada
	VMO vs HA-M5	182	28	0.004	Rechazada
	VMO-LS vs HA-M6	210.0	0.00	0.00	Rechazada

CONCLUSIONES

Como resultado de esta investigación se implementaron y validaron las diversas alternativas presentadas del Algoritmo Híbrido basado en el modelo VMO y ECS como estrategia para la hibridación; cumpliéndose de esta forma el objetivo general planteado, ya que:

1. Se realizó un estudio minucioso de las componentes del Algoritmo Híbrido propuesto, obteniéndose así las mejores configuraciones del modelo, en cuanto a valores de los parámetros de las mismas.
2. Se implementaron las distintas variantes de hibridación del modelo VMO empleando ECS con cada uno de los métodos de BL propuestos.

A partir de lo antes expuesto se pudo comprobar mediante la aplicación de varios tests estadísticos: Media, Porcentaje de desviación y el test de Ranking de Signos de Wilcoxon, que las variantes del Algoritmo Híbrido estudiadas e implementadas presentan, de manera significativa, una mejoría en cuanto a calidad de resultados a los arrojados por el modelo VMO puro. Además se comprobó también que al ser aplicada una estrategia de hibridación inteligente, en nuestro caso guiada por ECS para detectar zonas promisorias en el espacio de soluciones, se obtuvieron resultados significativamente mejores que los obtenidos por modelos híbridos basados también en VMO pero con BL aplicada a priori.

En lo que se refiere a cuál alternativa de nuestro Algoritmo Híbrido presentó los mejores resultados, también podemos concluir que de manera general no presentan diferencias significativas entre sí, aunque las configuraciones en las que se utilizó Asimilación por Cruzamiento presentaron un mejor comportamiento, esto quedó en evidencia gracias a los resultados arrojados después de aplicar el cálculo de la media ya que las dos seleccionadas (de menor promedio) resultaron ser configuraciones en las que se aplicó dicha Asimilación.

RECOMENDACIONES

La optimización es un campo que no se agota con la investigación realizada, por tanto para proseguir con la misma existen varios aspectos a recomendar, entre los cuales destacan:

1. Estudiar y aplicar otros métodos de Búsqueda Local, así como analizar el comportamiento de estos dentro del algoritmo híbrido.
2. Implementar otros tipos de asimilación similares a los operadores de cruce basados en entorno, con características propias de explotación, las cuales contribuyan al proceso de refinamiento de las soluciones dentro de un *cluster*.
3. Analizar nuevas alternativas de selección de nodos para la Búsqueda Local, no solo a tener en cuenta los centros de los *clusters* sino también los mejores nodos dentro de estos.



REFERENCIAS BIBLIOGRÁFICAS

Aarts, E. a. G. V. (1997). HandBook of Evolutionary Computation, Oxford University Press.

Chaves, A. A. and L. A. N. Lorena (2010). "Clustering search algorithm for the capacitated centered clustering problem." Computers & Operations Research **37**(3): 552-558.

Clark, A., B. Almada-Lobo, et al. (2011). " Lot sizing and scheduling: industrial extensions and research opportunities." International Journal of Production Research **49**(9)(2457-2461).

Darwin, C. (1859). On the Origin of Species, London: John Murray.

Dorigo, M. and G. D. Caro (1999). The Ant Colony Optimization meta-heuristic. New Ideas in Optimization. D. Corne, M. Dorigo and F. Glover. London UK, McGraw-Hill: 11-32.

Eshelman, L. J. and J. D. Schaffer (1993). Real-coded genetic algorithms and interval-schemata. Foundation of Genetic Algorithms 2, Morgan Kaufmann.

Glover, F. and G. Kochenberger, Eds. (2003). Handbook of Metaheuristics., Kluwer Academic Publishers.

Glover, F. and M. Laguna (1997). Tabu Search. MA, Kluwer Academic Publishers.

Goldberg, D. E., Ed. (1998). Genetic Algorithms in Search. Optimization and Machine Learning. University of Alabama, Addison-Wesley Publishing Company.

Guillén, G., M. Badell, et al. (2007). "A holistic framework for short-term supply chain management integrating production and corporate financial planning." International Journal of Production Economics **106**(1)(288-306).



Hatamlou, A. (2013). "Black hole: A new heuristic optimization approach for data clustering. ." Information Sciences **222(1)**(175-184).

Herrera, F., P. Cardoso, et al. (2009). ACOR híbrido con multiples colonias para problemas de optimizacion continua. CD de Memoria del VI Congreso Español sobre Metaheurísticas, Algoritmos Evolutivos y Bioinspirados (MAEB'09). Málaga: 465-472.

Herrera, F., D. Molina Cabrera, et al. (2009). Algoritmo Memético Basado en Encadenamiento de Búsquedas Locales para Problemas de Optimización Continua. CD de Memoria del VI Congreso Español sobre Metaheurísticas, Algoritmos Evolutivos y Bioinspirados (MAEB'09). Málaga: 481-486.

Holm, S. (1979). "A simple sequentially rejective multiple test procedure." Scand Journal Stat **6**: 65-70.

Jourdan, L., Basseur, M., & Talbi, E.-G. ((2009)). " Hybridizing exact methods and metaheuristics: A taxonomy." European Journal of Operational Research **199(3)**(620-629).

Kennedy, J. and R. C. Eberhart (1995). Particle swarm optimization. IEEE International Conference on Neural Networks, Piscataway, New York, USA, .

Laguna, M. and R. Martí (2003). Scatter Search. Methodology and Implementation in C, Kluwer Academic Publishers

Lamiri, M., F. Grimaud, et al. (2009). "Optimization methods for a stochastic surgery planning problem." International Journal of Production Economics **120(2)**(400-410).

Lawler, E. L. C. o. N. a. m. H., Rinehart and Winston, New York. (1976). "Combinatorial optimization: Networks and matroids." Holt, Rinehart and Winston, New York.

Liao, T. W. (2010). "Two hybrid differential evolution algorithms for engineering design optimization." Applied Soft Computing **10(4)**(1188-1199).

Lourenço, H., Martin, O., & Stützle, T. ((2003)). " Iterated Local Search. In F. Glover & G. Kochenberger (Eds.)." Handbook of Metaheuristics (Vol. 57, pp. 320-353)(Springer US).



Lozano, J., P. Larrañaga, et al., Eds. (2006). Towards a New Evolutionary Computation: Advances on Estimation of Distribution Algorithms., Springer Verlag.

Luenberguer, D. (1989). Programación lineal y no lineal, Luenberguer., Addison-Wesley Iberoamericana.

Luengo, J., S. García, et al. (2009). "A Study on the Use of Non-Parametric Tests for Analyzing the Evolutionary Algorithms' Behaviour."

Martínez, Y., A. Puris, et al. (2006). Aplicación de métodos heurísticos en la solución de problemas de configuración. Cd con las memorias del V Conferencia Internacional de Ciencias Empresariales CICE 2006. Santa Clara, Cuba.

Molina Cabrera, D. (2007). Algoritmos Meméticos con Aplicación Adaptativa de la Búsqueda Local para Optimización Continua. Departamento de Ciencias de la Computación e Inteligencia Artificial. Granada, Universidad de Granada: 258.

Nelder, J. and R. Mead (1965). "A simplex method for functions minimizations." Computer Journal 7 7(4): 308–313.

Nickel, S., M. Schröder, et al. (2012). "Mid-term and short-term planning support for home health care services." European Journal of Operational Research 219(3)(574-587).

Oliveira, A. and L. Lorena (2004). "Detecting promising areas by evolutionary clustering search." Advances in Artificial Intelligence–SBIA 2004: 318-323.

Oliveira, A. and L. Lorena (2007). "Hybrid evolutionary algorithms and clustering search." Hybrid Evolutionary Algorithms: 77-99.

Papadakos, N., G. Tzallas-Regas, et al. (2011). "Risky traveling salesman problem." European Journal of Operational Research 212(1)(69-73).



Puris, A., R. Bello, et al. (2012). "Variable mesh optimization for continuous optimization problems." Soft Computing-A Fusion of Foundations, Methodologies and Applications **16**(3): 511-525.

Puris, A. Y. (2009). Desarrollo de meta-heurísticas poblacionales para la solución de problemas complejos Departamento de Inteligencia Artificial. Santa Clara, Universidad Central "Marta Abreu" de las Villas.

Schilde, M., K. F. Doerner, et al. (2011). "Metaheuristics for the dynamic stochastic dial-a-ride problem with expected return transports." Computers & Operations Research **38**(12): 1719-1730.

Shariff, S. S. R., N. H. Moin, et al. (2012). "Location allocation modeling for healthcare facility planning in Malaysia." Computers & Industrial Engineering **62**(4): 1000-1010.

Sheskin, D. (2006). Handbook of parametric and nonparametric statistical procedures. CRC. London/West Palm Beach, Chapman & Hall. **1736**.

Solis, F. J. and R. J. Wets (1981). "Minimization by Random Search Techniques." Mathematical Operation Research **6**: 19-30.

Spendley, W., G. R. Hext, et al. (1962). "Sequential Application of Simplex Designs in Optimisation and Evolutionary Operation." Technometrics **4**(4): 441-457.

Storn, R. and K. Price (1997). "Differential Evolution. A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces." Journal of Global Optimization **11**: 341-359.

Suganthan, P., N. Hansen, et al. (2005). Problem Definitions and Evaluation Criteria for the CEC 2005 Special Session on Real-Parameter Optimization, Nanyang Technological University.

Voudouris, C. and E. Tsang (1995). Guided local search. University of Essex, Department of Computer Science.



Wilcoxon, F. (1945). " Individual comparisons by ranking methods." Biometrics Bulletin **1(6)**: 80-83.

Wolpert, D. H. and W. G. Macready (1997). "No Free Lunch Theorems for Optimization." IEEE Transactions of Evolutionary Computation **1**: 67-82.

ANEXOS

Anexo A. Tablas de los valores obtenidos para cada una de las funciones de prueba del CEC 2005 por los modelos estudiados.

Funciones	VMO-puro Dimensión 10	VMO-puro Dimensión 30	VMO-puro Dimensión 50
F6	7.56E+01	2.44E+03	2.64E+03
F7	1.27E+03	4.71E+03	6.24E+03
F8	2.03E+01	2.10E+01	2.12E+01
F9	1.15E+01	1.45E+02	4.17E+02
F10	8.37E+00	8.67E+01	2.54E+02
F11	1.96E+00	2.31E+01	5.19E+01
F12	4.46E+02	2.41E+04	2.39E+05
F13	1.05E+00	7.50E+00	2.62E+01
F14	2.69E+00	1.24E+01	2.21E+01
F15	2.94E+02	4.46E+02	5.25E+02
F16	1.10E+02	2.94E+02	2.85E+02
F17	1.11E+02	2.99E+02	3.24E+02
F18	5.43E+02	9.19E+02	9.86E+02
F19	5.65E+02	9.13E+02	9.94E+02
F20	6.40E+02	9.21E+02	9.93E+02
F21	4.60E+02	5.92E+02	5.52E+02
F22	7.59E+02	9.81E+02	1.03E+03
F23	5.54E+02	6.17E+02	1.01E+03
F24	2.12E+02	2.41E+02	7.32E+02
F25	2.00E+02	2.00E+02	7.26E+02

Tabla 1 Resultados del modelo VMO puro.

Funciones	VMO-BL Dimensión 10	VMO-BL Dimensión 30	VMO-BL Dimensión 50
F6	4.53E+01	2.58E+03	4.17E+03
F7	1.01E+03	4.72E+03	6.29E+03
F8	2.04E+01	2.10E+01	2.12E+01
F9	1.21E+01	1.68E+02	4.85E+02
F10	8.17E+00	1.00E+02	2.74E+02
F11	1.69E+00	2.27E+01	4.97E+01
F12	3.46E+02	2.81E+04	3.14E+05
F13	1.04E+00	1.04E+01	3.32E+01
F14	2.75E+00	1.23E+01	2.22E+01
F15	2.59E+02	4.69E+02	6.32E+02
F16	1.08E+02	1.96E+02	2.70E+02
F17	1.12E+02	2.85E+02	2.65E+02
F18	6.20E+02	9.18E+02	9.83E+02
F19	6.07E+02	9.18E+02	9.89E+02
F20	5.64E+02	9.18E+02	9.92E+02
F21	4.92E+02	5.37E+02	6.79E+02
F22	7.66E+02	9.67E+02	1.04E+03
F23	5.66E+02	6.44E+02	9.65E+02
F24	2.12E+02	2.00E+02	1.02E+03
F25	2.12E+02	2.44E+02	9.95E+02

Tabla 2 Resultados del modelo híbrido basado en VMO con BL aplicada a priori.

Funciones	HA-M2	HA-M4	HA-M6	HA-M8	HA-M10	HA-M12
F6	5.467E+01	1.766E+01	3.445E+01	6.690E+01	1.360E+02	6.698E+01
F7	8.194E+00	5.935E+00	5.783E+00	2.559E+02	1.267E+03	2.666E+02
F8	2.031E+01	2.031E+01	2.032E+01	2.032E+01	2.032E+01	2.030E+01
F9	8.711E+00	8.660E+00	7.024E+00	1.591E+01	1.373E+01	1.408E+01
F10	6.647E+00	7.803E+00	8.318E+00	7.471E+00	7.744E+00	6.861E+00
F11	9.637E-01	7.726E-01	1.142E+00	1.863E+00	1.481E+00	1.243E+00
F12	2.354E+02	2.408E+02	3.028E+02	1.854E+02	1.406E+02	6.204E+01
F13	8.486E-01	8.163E-01	9.533E-01	1.037E+00	1.090E+00	1.107E+00
F14	2.257E+00	2.014E+00	2.265E+00	2.546E+00	2.396E+00	2.434E+00
F15	2.485E+02	1.878E+02	1.706E+02	2.197E+02	2.386E+02	2.199E+02
F16	1.057E+02	1.082E+02	1.060E+02	1.089E+02	1.076E+02	1.094E+02
F17	1.108E+02	1.061E+02	1.113E+02	1.149E+02	1.147E+02	1.113E+02
F18	6.050E+02	6.782E+02	6.056E+02	6.441E+02	5.857E+02	6.801E+02
F19	6.935E+02	5.845E+02	5.900E+02	6.401E+02	6.837E+02	7.201E+02
F20	6.059E+02	5.740E+02	5.506E+02	6.701E+02	5.800E+02	7.114E+02
F21	4.120E+02	4.120E+02	4.440E+02	4.000E+02	4.280E+02	4.590E+02
F22	7.626E+02	7.565E+02	7.552E+02	6.957E+02	5.636E+02	5.288E+02
F23	6.436E+02	6.335E+02	6.242E+02	6.662E+02	6.760E+02	6.505E+02
F24	2.347E+02	2.466E+02	2.360E+02	2.001E+02	2.603E+02	2.241E+02
F25	2.000E+02	2.224E+02	2.110E+02	2.561E+02	2.241E+02	2.121E+02

Tabla 3 Resultados del modelo híbrido basado en VMO y ECS para dimensión 10 con Simplex como método de BL.

Funciones	HA-M1	HA-M3	HA-M5	HA-M7	HA-M9	HA-M11
F6	2.370E+01	2.213E+01	1.865E+01	7.006E+01	1.530E+02	1.804E+02
F7	2.238E+01	1.961E+01	1.883E+01	1.267E+03	1.268E+03	1.267E+03
F8	2.032E+01	2.033E+01	2.031E+01	2.031E+01	2.024E+01	2.032E+01
F9	8.319E+00	7.448E+00	8.682E+00	1.575E+01	1.699E+01	1.571E+01
F10	7.642E+00	6.608E+00	7.324E+00	8.391E+00	8.252E+00	8.218E+00
F11	1.045E+00	1.110E+00	1.059E+00	1.593E+00	1.857E+00	1.657E+00
F12	3.987E+02	3.141E+02	2.326E+02	1.376E+02	1.669E+02	1.633E+02
F13	8.244E-01	9.604E-01	8.307E-01	1.159E+00	1.052E+00	1.125E+00
F14	2.378E+00	2.172E+00	2.015E+00	2.586E+00	2.678E+00	2.497E+00
F15	2.380E+02	2.244E+02	2.318E+02	3.021E+02	2.763E+02	2.703E+02
F16	1.074E+02	1.096E+02	1.073E+02	1.059E+02	1.074E+02	1.078E+02
F17	1.076E+02	1.102E+02	1.112E+02	1.102E+02	1.138E+02	1.150E+02
F18	6.680E+02	6.385E+02	5.038E+02	6.487E+02	6.843E+02	6.636E+02
F19	6.097E+02	6.200E+02	6.167E+02	6.801E+02	6.892E+02	7.098E+02
F20	5.296E+02	5.657E+02	6.238E+02	6.001E+02	7.442E+02	6.201E+02
F21	4.200E+02	4.260E+02	4.360E+02	4.602E+02	3.968E+02	3.800E+02
F22	7.362E+02	7.498E+02	7.568E+02	6.904E+02	5.770E+02	5.776E+02
F23	5.983E+02	6.141E+02	6.465E+02	6.205E+02	6.741E+02	6.461E+02
F24	2.120E+02	2.000E+02	2.107E+02	2.321E+02	2.362E+02	2.241E+02
F25	2.327E+02	2.209E+02	2.240E+02	2.363E+02	2.481E+02	2.241E+02

Tabla 4 Resultados del modelo híbrido basado en VMO y ECS para dimensión 10 con Solis Wets como método de BL.

Funciones	HA-M2	HA-M4	HA-M6	HA-M8	HA-M10	HA-M12
F6	1.847E+03	3.365E+03	2.382E+03	4.608E+03	3.835E+03	3.659E+03
F7	8.878E+02	3.236E+03	2.799E+03	1.207E+03	4.727E+03	4.727E+03
F8	2.097E+01	2.097E+01	2.096E+01	2.098E+01	2.099E+01	2.097E+01
F9	1.499E+02	1.427E+02	1.509E+02	1.649E+02	1.668E+02	1.680E+02
F10	6.424E+01	6.474E+01	6.098E+01	6.720E+01	5.639E+01	5.906E+01
F11	1.930E+01	2.015E+01	1.860E+01	1.736E+01	1.731E+01	1.881E+01
F12	2.091E+04	1.754E+04	1.979E+04	3.512E+04	3.333E+04	2.625E+04
F13	7.728E+00	9.494E+00	8.508E+00	9.392E+00	9.947E+00	9.346E+00
F14	1.197E+01	1.190E+01	1.207E+01	1.244E+01	1.235E+01	1.243E+01
F15	3.688E+02	3.668E+02	3.600E+02	4.387E+02	4.180E+02	4.208E+02
F16	1.514E+02	1.194E+02	1.848E+02	2.213E+02	1.762E+02	1.935E+02
F17	1.907E+02	1.424E+02	1.701E+02	2.426E+02	1.754E+02	2.330E+02
F18	9.022E+02	8.998E+02	8.999E+02	9.169E+02	9.192E+02	9.180E+02
F19	9.025E+02	9.026E+02	9.006E+02	9.165E+02	9.162E+02	9.108E+02
F20	9.018E+02	9.002E+02	9.002E+02	9.170E+02	9.169E+02	9.165E+02
F21	5.521E+02	5.536E+02	5.415E+02	6.029E+02	6.437E+02	6.408E+02
F22	8.644E+02	8.746E+02	8.875E+02	9.560E+02	9.511E+02	9.377E+02
F23	5.669E+02	5.655E+02	6.571E+02	7.979E+02	7.371E+02	6.518E+02
F24	3.072E+02	2.941E+02	2.272E+02	2.421E+02	2.323E+02	2.669E+02
F25	3.468E+02	2.832E+02	3.609E+02	2.477E+02	2.132E+02	2.253E+02

Tabla 5 Resultados del modelo híbrido basado en VMO y ECS para dimensión 30 con Simplex como método de BL.

Funciones	HA-M1	HA-M3	HA-M5	HA-M7	HA-M9	HA-M11
F6	2.250E+03	3.272E+03	2.473E+03	4.833E+03	2.952E+03	2.849E+03
F7	3.097E+03	3.037E+03	2.863E+03	4.734E+03	4.723E+03	4.724E+03
F8	2.097E+01	2.096E+01	2.097E+01	2.097E+01	2.098E+01	2.097E+01
F9	1.449E+02	1.452E+02	1.343E+02	1.632E+02	1.597E+02	1.543E+02
F10	7.125E+01	6.688E+01	6.992E+01	6.092E+01	5.351E+01	5.595E+01
F11	1.996E+01	1.880E+01	1.996E+01	1.857E+01	1.733E+01	1.701E+01
F12	1.860E+04	2.473E+04	1.873E+04	3.661E+04	2.902E+04	3.401E+04
F13	8.530E+00	8.423E+00	8.356E+00	1.016E+01	9.150E+00	9.196E+00
F14	1.189E+01	1.199E+01	1.203E+01	1.226E+01	1.219E+01	1.235E+01
F15	3.710E+02	3.639E+02	3.989E+02	3.972E+02	4.249E+02	4.023E+02
F16	1.442E+02	2.381E+02	1.810E+02	2.128E+02	2.086E+02	2.157E+02
F17	1.782E+02	1.679E+02	1.740E+02	2.740E+02	1.909E+02	2.289E+02
F18	9.008E+02	8.992E+02	9.013E+02	9.175E+02	9.157E+02	9.196E+02
F19	9.001E+02	9.004E+02	8.999E+02	9.185E+02	9.174E+02	9.168E+02
F20	9.011E+02	8.997E+02	8.995E+02	9.162E+02	9.180E+02	9.146E+02
F21	5.640E+02	5.246E+02	5.401E+02	6.929E+02	6.169E+02	5.743E+02
F22	8.844E+02	8.779E+02	8.794E+02	9.619E+02	9.518E+02	9.408E+02
F23	5.843E+02	5.677E+02	6.159E+02	7.655E+02	6.704E+02	6.271E+02
F24	3.371E+02	4.162E+02	3.642E+02	2.673E+02	2.333E+02	2.408E+02
F25	3.484E+02	4.073E+02	4.511E+02	2.125E+02	2.003E+02	2.794E+02

Tabla 6 Resultados del modelo híbrido basado en VMO y ECS para dimensión 30 con Solis Wets como método de BL.

Funciones	HA-M2	HA-M4	HA-M6	HA-M8	HA-M10	HA-M12
F6	3.929E+03	1.910E+03	3.658E+03	8.861E+03	6.592E+03	5.108E+03
F7	2.440E+03	5.094E+03	5.074E+03	2.744E+03	6.315E+03	6.306E+03
F8	2.116E+01	2.115E+01	2.116E+01	2.117E+01	2.117E+01	2.116E+01
F9	3.404E+02	3.962E+02	3.767E+02	4.019E+02	3.854E+02	3.772E+02
F10	1.456E+02	1.430E+02	1.652E+02	1.653E+02	1.697E+02	1.576E+02
F11	4.401E+01	4.693E+01	4.584E+01	4.021E+01	3.927E+01	4.116E+01
F12	1.400E+05	1.210E+05	1.620E+05	2.156E+05	2.388E+05	1.931E+05
F13	2.675E+01	2.725E+01	2.773E+01	2.536E+01	2.403E+01	2.303E+01
F14	2.154E+01	2.145E+01	2.117E+01	2.179E+01	2.170E+01	2.166E+01
F15	4.135E+02	4.325E+02	4.362E+02	4.931E+02	4.772E+02	4.630E+02
F16	1.933E+02	2.013E+02	1.927E+02	2.171E+02	2.260E+02	1.678E+02
F17	1.776E+02	1.786E+02	2.231E+02	3.239E+02	3.003E+02	2.991E+02
F18	9.464E+02	9.521E+02	9.564E+02	9.189E+02	9.745E+02	9.542E+02
F19	9.525E+02	9.477E+02	9.547E+02	9.359E+02	9.665E+02	9.579E+02
F20	9.569E+02	9.574E+02	9.534E+02	9.334E+02	9.642E+02	9.634E+02
F21	6.715E+02	6.753E+02	6.507E+02	7.099E+02	8.355E+02	7.788E+02
F22	9.614E+02	9.615E+02	9.566E+02	1.023E+03	1.010E+03	1.016E+03
F23	7.189E+02	7.680E+02	7.529E+02	9.691E+02	9.020E+02	7.888E+02
F24	3.388E+02	2.633E+02	3.511E+02	1.091E+03	8.706E+02	1.007E+03
F25	3.701E+02	3.801E+02	2.361E+02	8.776E+02	1.040E+03	8.754E+02

Tabla 7 Resultados del modelo híbrido basado en VMO y ECS para dimensión 50 con Simplex como método de BL.

Funciones	HA-M1	HA-M3	HA-M5	HA-M7	HA-M9	HA-M11
F6	4.213E+03	5.727E+03	4.235E+03	5.018E+03	4.395E+03	5.680E+03
F7	5.208E+03	5.140E+03	4.971E+03	6.299E+03	6.288E+03	6.293E+03
F8	2.117E+01	2.118E+01	2.117E+01	2.117E+01	2.117E+01	2.119E+01
F9	3.692E+02	3.181E+02	3.822E+02	3.896E+02	3.835E+02	3.883E+02
F10	1.778E+02	1.667E+02	1.752E+02	1.824E+02	1.836E+02	1.720E+02
F11	4.479E+01	4.463E+01	4.547E+01	4.131E+01	4.170E+01	4.074E+01
F12	1.775E+05	1.433E+05	1.020E+05	2.774E+05	2.070E+05	2.000E+05
F13	2.540E+01	2.800E+01	2.442E+01	2.274E+01	2.462E+01	2.260E+01
F14	2.145E+01	2.144E+01	2.126E+01	2.171E+01	2.186E+01	2.163E+01
F15	4.318E+02	3.942E+02	4.370E+02	5.027E+02	4.477E+02	4.783E+02
F16	1.903E+02	1.905E+02	1.968E+02	2.141E+02	1.981E+02	1.960E+02
F17	2.304E+02	1.663E+02	2.165E+02	2.876E+02	2.909E+02	2.585E+02
F18	9.567E+02	9.503E+02	9.504E+02	9.617E+02	9.714E+02	9.662E+02
F19	9.514E+02	9.569E+02	9.578E+02	9.635E+02	9.661E+02	9.617E+02
F20	9.578E+02	9.607E+02	9.560E+02	9.619E+02	9.604E+02	9.591E+02
F21	6.617E+02	6.589E+02	6.029E+02	8.072E+02	8.607E+02	8.277E+02
F22	9.578E+02	9.550E+02	9.528E+02	1.017E+03	1.016E+03	1.014E+03
F23	7.423E+02	7.628E+02	7.048E+02	9.631E+02	7.916E+02	8.142E+02
F24	2.605E+02	2.430E+02	3.110E+02	1.058E+03	9.716E+02	9.958E+02
F25	3.007E+02	3.335E+02	2.718E+02	1.122E+03	1.026E+03	8.864E+02

Tabla 8 Resultados del modelo híbrido basado en VMO y ECS para dimensión 50 con Solis Wets como método de BL.



Anexo B. Breve Descripción de las funciones utilizadas como prueba

➤ Siete funciones básicas

- ✓ F_6 : Función de Rosenbrock desplazada, $x \in [-100, 100]^D$
- ✓ F_7 : Función de Griewank desplazada y rotada sin fronteras, $x \in [0, 600]^D$
- ✓ F_8 : Función de Ackley desplazada y rotada con óptimo local en las fronteras, $x \in [-32, 32]^D$
- ✓ F_9 : Función de Rastrigin desplazada, $x \in [-5, 5]^D$
- ✓ F_{10} : Función de Rastrigin desplazada y rotada, $x \in [-5, 5]^D$
- ✓ F_{11} : Función de Weierstrass desplazada y rotada, $x \in [-0.5, 0.5]^D$
- ✓ F_{12} : Problema 2.13 de Schwefel, $x \in [-\pi, \pi]^D$

➤ Dos funciones expandidas

- ✓ F_{13} : Función de Griewank expandida y extendida mas la función de Rosenbrock $x \in [-3, 1]^D$
- ✓ F_{14} : Función de Scaffer desplazada, rotada y expandida $x \in [-100, 100]^D$

➤ Once funciones híbridas

- ✓ $F_{15}, \dots, F_{25}$: $x \in [-5, 5]^D$