Optimización basada en Mallas Variables: Caso de estudio Viajante de Comercio

Byron Oviedo Bayas
Universidad Técnica Estatal de Quevedo, Quevedo, Los Ríos, Ecuador, boviedo@uteq.edu.ec

Jorge Humberto Guanín Fajardo
Universidad Técnica Estatal de Quevedo, Quevedo, Los Ríos, Ecuador, jguanin@uteq.edu.ec

Eduardo Díaz Ocampo
Universidad Técnica Estatal de Quevedo, Quevedo, Los Ríos, Ecuador, ediaz@uteq.edu.ec

AmilkarPuris Cáceres
Universidad Técnica Estatal de Quevedo, Quevedo, Los Ríos, Ecuador, apuris@uteq.edu.ec

ABSTRACT

A proposal is presented in this paper to implement the meta- heuristic Variable Mesh Optimization (VMO) at discrete Traveling Salesman Problem(TSP), where this model explores the search space from a population of solutions called mesh expands and contracts in order to find solutions of good quality. In this context the operator of expansion so as to be applicable in a discrete domain , making combinations between the solutions in order to obtain new nodes is changed. Another element that is changed is the operator of clearing , which is responsible for maintaining the diversity of the mesh in each interaction . A study of VMO model parameters using a set of TSP instances with different carcterísticas summarized in this paper, in addition, we can see that the purpose of this paper obtains competitive results when compared with others international reference algorithms mentioned in the statement of art.

Keywords: Combinatorial optimization, Meta-heuristics population, VMO

RESUMEN

En este trabajo se presenta una propuesta para aplicar la meta-heurística Optimización Basada en Mallas Variables (VMO) al problema discreto del Viajero Vendedor (TSP); este modelo explora el espacio de búsqueda a partir de una población de soluciones llamada malla que se expande y contrae con la finalidad de encontrar soluciones de buena calidad.En este contexto se modifica el operador de expansión de manera tal que sea aplicable en un dominio discreto, realizando combinaciones entre las soluciones a fin de obtener nuevos nodos. Otro de los elementos que se modifica es el operador de clearing, el cual se encarga de mantener la diversidad de la malla en cada interación. Se resume en este trabajo un estudio de parámetros del modelo VMO utilizando un conjunto de instancias de TSP con diferentes carcterísticas; además, se puede observar que la propuesta de este trabajo obtiene resultados competitivos al compararlos con otros algoritmos de referencia internacional mencionado en el estado del arte.

Palabras claves: Optimización combinatoria, Meta-heurísticas poblacionales, VMO

1. Introducción

Las meta-heurísticas (**Glover**, **2008**)basadas en población desde su aparición representan una herramienta muy apropiada para resolver problemas discretos. Estos algoritmos utilizan una población (conjunto de soluciones) para explorar el espacio de búsqueda complejo que supone la optimización combinatoria. El problema del

Viajero Vendedor (también conocido como problema del Viajante de Comercio o por sus siglas en inglés: TSP) (Johnson, 1997)es uno de los problemas más famosos (y quizás el mejor estudiado) en el campo de la optimización combinatoria.

Desde el punto de vista práctico, el problema no está resuelto y desde el punto de vista teórico, las técnicas empleadas son sólo aproximaciones. Estas no suponen una resolución real del TSP y sólo ofrecen soluciones aproximadas suficientemente aceptables. Por otra parte los algoritmos clásicos no son capaces de resolver el problema general, debido a la explosión combinatoria de las posibles soluciones. Por ello, a su solución se han aplicado distintas técnicas meta-heurísticas (Michalewicz, 2004).

Las meta-heurísticas poblacionales han sido las mas aplicadas a la solucion del TSP. Estos algoritmos trabajan sobre un conjunto de soluciones potenciales, llamada población, y realizan la exploración del espacio de búsqueda a través de la cooperación y competencia entre las soluciones potenciales; ejemplos de estas técnicas son los Algoritmos Genéticos (Genetic Algorithms, GA)(Golberg, 1998), la Optimización basada en Colonias de Hormigas (Ant Colony Optimization, ACO) (Dorigo, 2004)y la Optimización basada en Enjambres de Partículas (Particle Swarm Optimization, PSO) (Eberhart, 1995).

En trabajos anteriores se presentó la Optimización basada en Malla Variable (VMO)(Puris, et al., 2012) como una meta-heurística eficiente para problemas de optimización contínua. VMO utiliza una población conocida como malla para explorar el espacio de búsqueda. Esta malla se expande (incorporando nuevas soluciones) utilizando distintas formas de generación de soluciones. Finalmente, la malla es reducida utilizando un operador de limpieza adaptativo (clearing) que elimina todo nodo o solución que se encuentre cerca de otro de mejor calidad (fitness). Este algoritmo ha presentado soluciones competitivas en problemas contínuos con diferentes características:aproximación de funciones multimodales(Puris, et al., 2012), problemas de nichos(Molina, et al., 2013), óptimos en las fronteras(Navarro, et al., 2009); pero, no se conoce de algún estudio previo sobre la aplicación de esta técnica en problemas de optimización combinatoria.

En este trabajo presentamos una adaptación de VMO orientada a explorar espacios discretos. Específicamente se aplica a la solución de diferentes instancias del TSP como una referencia obligatoria para probar la efectividad de nuevos algoritmos. Además, se incorpora un método de búsqueda local (BL) con el objetivo de explorar detalladamente los entornos de las zonas más prometedoras y hacerlas más representativas. Para finalizar el estudio, se hace una comparación entre los resultados alcanzadospor la propuesta de VMO y tres algoritmos basados en Colonias de Hormigas (ACS, AS, MMAS) (Dorigo, 2004).

El trabajo esta estructurado de la siguiente manera: En el apartado 1 se describe los aspectos fundamentales de problema de estudio TSP. Seguidamente en el segundo se explica el funcionamiento general de VMO, en el tercero se define cada uno de los operadores de expansión y contración para el problema de estudio. Posteriormente en el apartado cuarto se realiza un estudio de parámetros de la propuesta y un análisis comparativo experimental con los resultados obtenidos con otros algortimos mencionado en el estado del arte.

2. Viajante de Comercio

El Viajante de Comercio(Johnson, 1997) puede ser representado como un grafo completo G = (N, A), donde N representa la cantidad de nodos también llamados ciudades y A el conjunto de arcos bidireccionales que conectan los nodos. Cada acro a_{ij} se le asigna un valor d_{ij} que representa la distancia entre las ciudades *i* y *j*. El objetivode este problema radica en encontrar el camino (ciclo hamiltoniano) más corto desde una ciudad de partida, pasando por todas las demás ciudades solo una vez y regresando a la ciudadde origen. En el caso del TSP simétrico las distancias entre las ciudades son independientes de la direccion de la ruta ($d_{ij}=d_{ji}$). Todas las instancias utilizadas para el estudio experimental de este trabajo son tomadas de la biblioteca pública TSPLIB[1](Reinelt, 1991) y son detalladas en el apartado de los resultados experimentales.

[1]The TSPLIB Symmetric Traveling Salesman Problem Instances, http://elib.zib.de/pub/Packages/mp-testdata/tsp/tsplib/index.html

3. Optimizacion basada en Mallas Variables

VMO es una meta-heurística poblacional en la que la población se distribuye como una malla de nodos (n_1, n_2, ... , n_P) que representan soluciones en el espacio de búsqueda. La exploración se lleva a cabo por dos procesos: de expansión y de contraccion (Puris, et al., 2012). El proceso de expansión genera nuevos nodos utilizando la población en la iteración actual, por medio de los pasos siguientes:

<u>Fase 1</u> (Generación de la poblacion inicial): Se generan Psoluciones factibles de manera aleatoria que representan los nodos de la malla inicial en el proceso de búsqueda del algoritmos.

<u>Fase 2</u> (Generación de nodos hacia los mejores vecinos): Para cada solución de la malla (n_i) se buscan los k nodos más cercanos. A continuación se selecciona el vecino de mejor calidad (fitness) ($n*$),y si $n*$ es mejor que n_i se genera un nuevo nodo entre ambos; caso contrario, no se genera una nueva solución.

<u>Fase 3</u> (Generación de nodos hacia la mejor solución):Se crea un nuevo nodo para cada n_i en dirección al nodo con mejor valor de fitness (n_g, óptimo global) de la malla actual.

<u>Fase 4</u> (Generación de nodos utilizando los nodos fronteras de la malla): Los nodos fronteras se definen como las soluciones que se encuentran más cerca y más lejos del centro de la población. En esta fase las nuevas soluciones se generan desplazando cada nodo frontera en función de un valor de desplazamiento w(Puris, et al., 2012).

<u>Fase de Clearing</u>: En esta fase, se seleccionan las soluciones más representativas entre la malla actual y los nuevos nodos generados en el proceso de expansión (Fase 2, 3, 4). Para ello, se evita seleccionar soluciones demasiado cercanas entre sí. Durante el clearing se define una distancia ξ como distancia mínima entre las soluciones, y se tiene que ignorar todo nodo que tenga una distancia inferior a un nodo con mejor o igual calidad. Dicha distancia se inicializa a un 25% del rango de búsqueda, y va reduciendose durante la ejecución del algoritmo (Puris, et al., 2012).

En el próximo apartado de definirá cada uno de los operadores de generación de nodos utilizados para el caso de estudio del TSP y la forma en que se realiza la operación de clearing. Además en la Tabla 1se describe cada uno de los parámetros que utiliza el algoritmo.

4. VMO para el TSP

Aunque la definición general de la meta-heurística VMO no está dirigida exactamente para un tipo de dominio, se puede decir que sus operadores de expansión utilizan formas de generación de nuevas soluciones siguiendo un enfoque contínuo(más información en (Puris, et al., 2012)), por lo que a continuación se redefinen cada unos de ellos para adaptarlos a la generación de soluciones en dominios discretos.

4.1. Generación de la malla inicial

En este paso se generan P soluciones factibles de forma aleatoria. Cada nodo se genera considerando que no se puede asignar una misma ciudad en las diferentes posiciones, de manera que un nodo represente un camino o ciclo hamiltoniano desde cualquier ciudad de origen (también seleccionada de manera aleatoria entre todas las ciudades). A continuación se representan los casos posibles de una solución donde se representa el caso a) como favorable ya que no se repite ninguna ciudad en el trayecto, y el caso b) en el cúal se puede apreciar que la ciudad 1 se repite por lo que no representa una solución factible para el problema.

1	4	3	2	5

a) Solución factible

1	4	1	3	5

b) Solución no factible

Figura 2: Ejemplo de posibles soluciones para TSP de 5 ciudades

4.2. Generación de nodos hacia los mejores vecinos

En este paso de generación se utlizan la distancia de Hammingparaobtener los k vecinos más cercanos de cada nodo n_i de lamalla,luego se selecciona entre los k vecinos el recorrido de menor distancia n* (nodo de mejor calidad) y si este representa una mejor solución que el nodo n_i se creará una nueva solución a través de una combinacion entre ambas soluciones ($n_{i,}$n*) de la siguiente manera:

Paso1: Seleccionar como ciudad de partida de la nueva solución la que presenta la solución n*.
Paso2: Ocupar la siguiente posición por la ciudad de n_i o n* que introduzca menos distancia a la nueva solución.
Paso3: Si la ciudad seleccionada ya existe en la nueva solución, se selecciona la otra posible ciudad. Si ambas ciudades aparecen se deja la posición vacía y se pasa a completar la próxima posición con el Paso 2.
Paso4: Si no falta posición por explorar pasar a completar las posiciones vacías seleccionando aleatoriamente entre el conjunto de ciudades no asignadas.

Figura 3: Pasos para la generación de un nuevo nodo a partir de la combinación de dos

4.3. Generación de nodos hacia la mejor solución global

Para generar nuevas soluciones utilizando este paso primero se selecciona el nodo que representa el recorrido de menor distancia (n_g, óptimo global), luego se aplica el algoritmo de la figura 3 (cambiando n* por n_g) para generar nuevas soluciones entre cada nodo de la malla inicial y el nodo n_g.

4.4 Generación de nodo a partir de las fronteras de la malla

En este paso se seleccionan los nodos o soluciones que representan las fronteras de la malla inicial, esto se lleva a cabo utilizando la distancia de Hamming para crear una matriz simétrica de distancias entre nodos. Luego se selecciona como frontera interior al nodo que más cerca se encuentre de los demás y como frontera exterior el que más lejano se encuentre. Luego a estos nodos se les realiza una permutación de dos posiciones seleccionadas de manera aleatoria.

En la propuesta original de VMO se seleccionan como frontera interior y exterior una cantidad de nodos necesarios para completar la cantidad de soluciones de la malla total. En nuestra propuesta solo se selecciona una solución por cada conjunto.

4.5 Operador de clearing

Para el caso de estudio, en este operador se ordena de manera ascendente los nodos en función de la calidad del recorrido que representan, luego se eliminan aquellas soluciones que esten a una distancia de Hamming menor que ξ de otro nodo de mejor calidad. La ecuación 1 presenta la forma de calcula ξ donde size(TSP) representa la cantidad de ciudades de la instancia que sera computada, C es la cantidad total de iteraciones que realizará el algoritmo (parámetro de VMO) y c la iteracción que se está ejecutando en el momento del cálculo de la distancia, cumpliendo que $c \leq C$.

$$\xi = \begin{cases} size(TSP) - \dfrac{size(TSP)}{2}, si\ c \leq 15\%C \\ size(TSP) - \dfrac{size(TSP)}{4}, si\ 15\%C < c \leq 30\%C \\ size(TSP) - \dfrac{size(TSP)}{6}, si\ 30\%C < c \leq 60\%C \\ size(TSP) - \dfrac{size(TSP)}{8}, si\ c \leq 80\%C \end{cases} \quad (1)$$

El conteo aplicado por la distancia de Haminng no se realiza posisción a posición como lo propone el algoritmooriginal; sino, por secuencias de tamaño 2.

4.6. Aplicación de la búsqueda local

Los procedimientos de búsqueda local comienzan con una solución del problema y la van mejorando progresivamente. El procedimiento realiza en cada paso un movimiento de una solución a otra con mejor valor. El método finaliza cuandoen un estado no existe ninguna solución accesible que la mejore. Para el caso del TSP existen diferentes algoritmos de búsqueda local entre los que destaca el Recocido Simulado(Gellat, et al., 1983), Búsqueda Tabú(Glover & Laguna, 1997),Lin y Kernighan y los conocidos k-intercambios(StützleH & Hoos, 1997)**.** Entre este último el algoritmo 2-intercambio (2-opt)(StützleH & Hoos, 1997)es de los más simples y utilizados en la literatura para hibridar con Meta-heurística poblacionales. En la presente investigación se propone utilizar este algoritmo para aplicarlo a cada una de las soluciones obtenidas en los operadores de generación de nodos del algoritmo VMO.

4.7. Modelo general de aplicación de VMO al TSP

A continuación en la figura 5 se describe el pseudocódigo para aplicar VMO al problema TSP en el cual se hace referencia de algunos pasos del algoritmo descritos con anterioridad en este capítulo.

```
begin
Generar de modo aleatorio P nodos para la malla inicial
Encontrar el extremo global (nodo de mejor calidad)
repeat
  for each nodo en la malla inicial do
     Encontrar sus k nodos más cercanos utilizando la distancia de Hamming
     Seleccionar el mejor vecino según su calidad (camino más corto)
     if el nodo actual no es el mejor local then
        Generar un nuevo nodo entre el nodo actual y el extremo local utilizando la Figura 2
     end if
  end for
  for each nodo en la malla inicial (exceptuando el mejor global) do
     Generar un nuevo nodo entre el nodo actual y el extremo global utilizando la Figura 2
  end for
  Seleccionar el nodo frontera inferior y frontera superior
  Generar nodos a partir de las fronteras de la malla, Aplicar Paso 4.4
  Ordenar los nodos según su calidad
  Aplicar el operador de limpieza adaptable según Paso 4.5
  Seleccionar los P mejores nodos para construir la malla inicial de la próxima iteración
  De ser preciso, generar de modo aleatorio nuevos nodos para completar dicha malla
until C
end
```

Figura 5: Pseudocódigo de la propuesta de VMO para TSP

5. Resultados experimentales

En esta sección se ha preparado un estudio experimental para probar el desempeño de la propuesta VMO para la solución de diferentes instancias del TSP (bays29, st70, rd100, ch150, kroA200, tsp225, a280, lin318, pcb442)(Reinelt, 1991), donde el número que aparece en el nombre de estas, representa la cantidad de ciudades que tiene la instancia.

A continuación se detallan los experimentos realizados para ajustar los parametros del algoritmo VMO los cuales son P tamaño de la malla inicial, k cantidad de nodos que conforman la vecindad y C la cantidad de iteraciones que ejecutara el algoritmo. Luego una comparación con los resultados obtenidos por tres algoritmos de la meta-heuristica ACO (Ant Colony System (ACS), Ant System (AS) y MaxMin Ant System (MMAS))(Dorigo, 2004)como uno de los modelos poblacionales que mejores soluciones ha presentado al TSP.

Para comprobar los experimentos se ha utilizado un conjunto de técnicas estadísticas no paramétricas propuestas en (Garcia, et al., 2007) como una herramienta para comparar resultados obtenidos por métodos aproximados. En la discusión de los resultados se describe el test utilizado y la forma de interpretar los resultados de estos.

5.1. Tamano de la malla inicial

A diferencia de la propuesta original de VMO donde se estudia la relación directa entre el tamaño de la población inicial y el tamaño máximo de la expansión, en este caso solo se estudiarála influencia del tamaño de la población inicial en la evolución del algoritmo (8, 12, 24) .

Tabla 2:Resultados del test Iman-Davenport con α=0.05

Método	Valor del Test	Valor p	Hipótesis
Iman-Davenport	5.066	1.76E-13	Rechazada

Como se puede apreciar en la Tabla 2, el valor p obtenido por el test de Iman-Davenport para detectar diferencias en el grupo formado por las tres variantes estudiadas (VMO(8), VMO(12), VMO(24)) es menor que el valor α de significancia del test, por lo que la hipótesis de semejanza es rechazada, concluyéndose que existen diferencias significativas entre los resultados computados.

Para el caso a continuación se aplica un test de Holm, tomando como muestra de control los resultados alcanzado por VMO(24). Esta selección se lleva a cabo porque es el que mejor valor medio obtuvó.

Tabla 3. Resultados del test de Holm para un valor α=0.05

Algoritmo	Z	Valor p	α/i	Hipótesis
VMO(8)	2.999	7.74E-6	0.025	Rechazada
VMO(12)	1.999	0.02534	0.05	Rechazada

Como se puede apreciar en la Tabla 3, el test de Holm detecta diferencias significativas entre la muestra de control (VMO(24)) y cada una del otras variantes comapradas. Esto se evidencia porque los valores p de cada algoritmo es menor que su correspondiente valor α/i, rechazandose la hipótesis de semejanza en cada una de las comparaciones.

De este análisis se puede concluir que los resultados alcanzados utilizando una malla inicial de tamaño 24 obtiene resultados significativamente mejores que las otras variantes de configuración de malla inicial.

5.2. Cantidad de vecinos utilizados en la Face 2

Este parámetro es utilizado en la propuesta para determinar la cantidad de soluciones que conforman la vecindad de un nodo n_i. En este experimento se probaron diferentes valores 3 y 5 que son los propuestos por los autores en el estudio desarrollado en (Puris, et al., 2012)

A continuación la Tabla 3 presenta los resultados del test de Wilcoxon para comparar ambas propuestas, donde se puede apreciar que existen diferencias significativas a favor de la propuesta que utiliza 5 vecinos para la face 2 de VMO. Esto se puede evidenciar a partir de que el valor p es menor que el valor de significancia aplicado al test rechazando la hipótesis de igualdad

Tabla 4: Resultados del test de Wilcoxon con α=0.05

Algoritmo	R+	R-	Valor p	Hipótesis
VMO(5) vs VMO(3)	8.00	47.00	0.047	Rechazada

5.3. Comparación con Sistema de Colonia de Hormigas

A continuación se realiza una comparación entre los resultados obtenidos en esta investigación y uno de los algoritmos de la meta-heuristica ACO (Sistema de Colonia de Hormigas; ACS) que más ha sido estudiado en la solucion del TSP. En ambos casos fueron ejecutados los algoritmos para la misma cantidad de iteraciones 1000*F, donde F representa el tamaño de la instancia del TSP utilizada y además ambos algoritmos utilizaron la busqueda local 2-opt. La configuración de parámetros y los resultados de ACS, MMAS y AS fue la utilizada en (Puris, et al., 2010)

Tabla 5: Resultados del test de Wilcoxon con α=0.05

Algoritmo	R+	R-	Valor p	Hipótesis
VMO-ACS	34.0	32.00	0.87	Aceptada
VMO-MMAS	31.0	35.00	0.73	Aceptada
VMO-AS	5.0	54.0	0.024	Rechazada

La Tabla 5 muestra los resultados del test de Wilcoxon dende se puede apreciar que el valor p es mayor que α para las comparaciones con los algoritmos ACS y MMAS por lo que la hipótesis de igualdad es aceptada. De estos se concluye que no esisten diferencias significaticas entre la calidad de los resultados obtenidos por ambos algoritmos y VMO para el TSP. Para el caso de la comparación con el algoritmo AS se puede aprecias que la hipótesis es rechazada a favor VMO demostrando que los resultados alcanzados por este algoritmo son significativamente mejores que los obtenidos por AS.

6. Conclusiones

Como resultado de esta investigación se desarrolló una propuesta para aplicar la meta-heurística VMO al problema discreto TSP que difiere del modelo original en los siguientes aspectos:

- Se aplicaron otros operadores de generación de nuevos nodos en el proceso de expansión, donde se realiza una conbinación de soluciones de manera que cumpla con las restricciones impuestas por el problema.
- Se aplicó una variante de la distancia de Hamming donde se comparan secuencias de tamaño 2 para realizar el operador de clearing con el objetivo de eliminar soluciones que compartan la mayor cantidad de secuancias iguales.
- Solo dos soluciones forman el conjunto frontera donde se identifican como las más y menos semejantes a los demás nodos de las mallas. Luego se perturban las soluciones a traves de simples permutaciones internas en dichos nodos.
- Se aplicó el algoritmo de búsqueda local 2-opt a cada solución generada en el proceso de expansión con el fin de mejorar las soluciones obtenidas.
- Se realizó un estudio de los parámetros, tamaño de la malla (P) y tamaño de la vecindad (k), donde se obtuvieron los mejores resultados para un valor de P=24 y k=5 entre los valores probados.

Finalmente se realizó un estudio comparativo entre los resultados alcanzado por la propuesta VMO y tres de los principales algoritmos de la meta-heurística poblacional ACO. De este estudio se pudo concluir que el algoritmo VMO obtiene resultados semejantes a los encontrados por los algoritmos ACS y MMAS y mejores que la variante AS.Estos resultados son muy importantesal serla primera investigación que se realiza con VMO para un problema discreto y de tanta complejidad como el TSP, quedando abiertaslineas de trabajo en este tema.

7. Autorización y Renuncia

Los Autores autorizan a LACCEI para publicar el documento en las actas del congreso. LACCEI o los editores no son responsables ni por el contenido ni por las implicaciones de lo que se expresa en el documento.

8. Entrega

Todos los manuscritos han sido enviados a través del enlace de presentación de papers en la página web de LACCEI: http://www.laccei.org en formato Microsoft Word. La fecha de envío es 10 de marzo del 2014.

9. Referencias

Bonabeau, E., 1999. *From natural to Artificial systems, , in Swarm Intelligence.* s.l.:Oxford University Press.

Dorigo, M., 2004. *Ant Colony Optimization.* s.l.:MIT Press.

Eberhart, R., 1995. *A new optimizer using particle swarm theory.* Nagoya, Japan, Kennedy, p. 10.

Friedman, M., 1940. A Comparison of Alternative Tests of Significance for the Problem of m Ranking. *Annals of Mathematical Statistics,* p. 86–92.

Garcia, S., Molina, D. & Herrera , F., 2007. *An experimental study about the use of non-parametric tests for analyzing the behaviour of evolutionary algorithms in optimization problems.* Spain, s.n., p. 10.

Glover, F., 2008. *Handbook of meta-heuristics.* Massachusetts: Kluwer Academic Publisher.

Glover, F. & Laguna, M., 1997. Tabu Search. *Kluwer Academic Publishers,* pp. 73-82.

Golberg, D., 1998. *Genetic Algorithms in Search in Optimization and Machine Learning.* University of Alabama USA: Addison-Wesley.

Gellat, C., Vecchi, M. & Kirkpatrick, S., 1983. Optimization by simulated annealing. *Science,* 220(4598), p. 671–680.

Johnson, D., 1997. *The traveling salesman problem: a case study in local optimization.* New York, Wiley, p. 24.

Kennedy, A., 2001. *Swarm Intelligence.* s.l.:Morgan Kaufmann.

Michalewicz, F., 2004. *How to solve It: Modern heuritics.* s.l.:Springer.

Molina, D., Puris, A., Bello, R. & Herrera , F., 2013. *Variable Mesh Optimization for the 2013 CEC Special Session Niching Methods for Multimodal Optimization.* Mexico, IEEE, pp. 87-94.

Muhlenbein, H., 1998. *Parallel genetic algorithms, population genetics and combinatorial optimization.* Viña del Mar, Springer, p. 10.

Navarro, R., Puris, A., Bello, R. & Herrera, F., 2009. Estudio del desempeño de la optimización basada en mallas variables en problemas con óptimos en las fronteras del espacio búqueda. *Revista Cubana de Ciencias Informáticas,* 3(4), pp. 24-30.

Puris, A., Bello, R., Molina , D. & Herrera, F., 2012. Variable mesh optimization for continuous optimization problems. *Soft Computing,* pp. 511-525.

Puris, A., Bello, R., Molina, D. & Herrera, F., 2010. Analysis of the efficacy of a two-stage methodology for ant colony optimization: Case of study with TSP and QAP. *Expert Systems with Applications,* Volume 37, p. 5443–5453.

Reinelt, G., 1991. TSPLIB - A traveling salesman library. *ORSA Journal on Computing,* Volume 3, pp. 376-384.

StützleH, T. & Hoos, H., 1997. *The MAX–MIN ant system and local search for the traveling salesman problem.* USA, IEEE, p. 309–314.