

# *Simulated Annealing* aplicado a Triangulaciones y Pseudotriangulaciones de Peso Mínimo \*

Maria Gisela Dorzán, Edilma Olinda Gagliardi, Mario Guillermo Leguizamón<sup>1</sup>  
Gregorio Hernández Peñalver<sup>2</sup>

<sup>1</sup> Facultad de Ciencias Físico Matemáticas y Naturales,  
Universidad Nacional de San Luis, Argentina  
{mgdorzan, oli, legui}@unsl.edu.ar

<sup>2</sup> Facultad de Informática, Universidad Politécnica de Madrid, España  
gregorio@fi.upm.es

**Abstract.** Muchos problemas de optimización en configuraciones geométricas son NP-duros. En este artículo, consideramos los problemas de Triangulación de Peso Mínimo (*Minimum Weight Triangulation, MWT*) y Pseudotriangulación de Peso Mínimo (*Minimum Weight Pseudo-triangulation, MWPT*) para un conjunto dado de puntos en el plano, y mostramos el diseño para la técnica metaheurística *Simulated Annealing* (SA) que permite resolverlos de forma aproximada.

**Key words:** Geometría Computacional, Triangulaciones, Pseudotriangulaciones, Triangulación de Peso Mínimo, Pseudotriangulación de Peso Mínimo, Metaheurísticas, Simulated Annealing.

## 1 Introducción

En Geometría Computacional hay numerosos problemas que, o bien son de naturaleza NP-dura, o bien son problemas para los cuales no se conocen soluciones eficientes. De todos modos, resulta de interés encontrar soluciones a tales problemas, aunque sean aproximadas a las óptimas, por medio de métodos de naturaleza heurística. En particular, es interesante el estudio de problemas de optimización geométrica relacionados con ciertas configuraciones geométricas obtenidas a partir de un conjunto de puntos, tales como son las triangulaciones y las pseudotriangulaciones. En estos problemas se busca optimizar ciertas propiedades que miden la calidad de las configuraciones: peso, perímetro, dilación, factor de carga, etc. Los problemas de Triangulación de Peso Mínimo (*Minimum Weight Triangulation, MWT*) y Pseudotriangulación de Peso Mínimo (*Minimum Weight Pseudo-triangulation, MWPT*) minimizan la suma de las longitudes de las aristas, dando una medida de calidad para determinar cuán buena es la estructura geométrica. Mulzer y Rote [19] demostraron en 2006 que el problema MWT es de naturaleza NP-dura, eliminando dicho problema de la lista de

\* Este trabajo es parcialmente subvencionado por el Proyecto “Tecnologías Avanzadas de Base de Datos” 22/F614-UNSL; y parcialmente subvencionado por el proyecto del Ministerio Ciencia e Innovación MTM2008-05043- España.

problemas abiertos presentado por Garey y Johnson [9]. La complejidad para el problema MWPT es aún desconocido, aunque Levkopoulos y Gudmundsson [11] indicaron cómo obtener una 12-aproximación de una pseudotriangulación de peso mínimo puede ser calculada en tiempo  $O(n^3)$ . Además obtuvieron una  $O(\log n \cdot w(MST))$  aproximación de una pseudotriangulación de peso mínimo, calculada en tiempo  $O(n \log n)$ , donde  $w(MST)$  es el peso del árbol generador mínimo.

Considerando el actual estado del arte de los problemas presentados, resolverlos utilizando técnicas metaheurísticas resultan los métodos más apropiados para encontrar soluciones aproximadas a las óptimas. En [4, 6, 7, 5] se puede observar el trabajo realizado por los autores, sobre los problemas MWT y MWPT utilizando metaheurísticas, donde se describe el diseño de algoritmos y resultados experimentales obtenidos utilizando la técnica Optimización basada en Colonia de Hormigas (*Ant Colony Optimization, ACO*).

En este artículo, consideramos los problemas de MWT y MWPT para un conjunto dado de puntos en el plano, y mostramos como la técnica metaheurística *Simulated Annealing*(SA) permite resolverlos de forma aproximada. Mostramos el plan diseñado para llevar a cabo en el estudio experimental, que permita comprobar el desempeño de dicha técnica en los problemas mencionados MWT y MWPT.

Esta presentación está organizada de la siguiente manera. En la Sección 2 presentamos los aspectos teóricos y estado del arte de los problemas planteados. En la Sección 3 describimos la estrategia Simulated Annealing. En la Sección 4 detallamos los parámetros, incluidos los operadores de vecindad utilizados. Finalmente, en la Sección 5, damos nuestras conclusiones y visión de futuro de esta investigación.

## 2 Triangulación y Pseudotriangulación de Peso Mínimo

### 2.1 Triangulación de Peso Mínimo

Sea  $S$  un conjunto de puntos en el plano. Una triangulación de  $S$  es una partición del cierre convexo de  $S$  en triángulos cuyo conjunto de vértices es exactamente  $S$ . El peso de una triangulación  $T$  es la suma de las longitudes Euclídeas de todas las aristas de  $T$ . La triangulación que minimiza dicha suma se denomina *Triangulación de Mínimo Peso* (*Minimum Weight Triangulation*) de  $S$  y se denota por  $MWT(S)$ .

Düppe y Gottschalk [8] propusieron un algoritmo ávido (*greedy*) insertando en cada paso la arista más corta a la triangulación. Cinco años más tarde, en 1975, Shamos y Hoey [24] sugirieron utilizar la triangulación de Delaunay como la triangulación de mínimo peso. Pero Lloyd [17] mostró ejemplos donde los dos algoritmos propuestos anteriormente no calculaban la triangulación de mínimo peso. De manera independiente, Gilbert [10] y Klincsek [15] mostraron cómo calcular la triangulación de mínimo peso de un polígono simple en tiempo  $O(n^3)$  utilizando programación dinámica.

En 1993, Wu y Wainwright [25] aproximan la triangulación de peso mínimo utilizando un algoritmo genético donde los operadores de recombinación y de mutación son iguales, es decir, ambos realizan un *flip* para obtener los individuos de la próxima población. Qin et al. [21] también utilizan un algoritmo genético para lo que proponen nuevos operadores de recombinación y mutación adaptados al problema. Capp y Julstrom [1] presentan una nueva codificación ponderada de las triangulaciones para aplicar en un algoritmo genético. En los trabajos anteriores la evaluación experimental es muy limitada y no se extraen conclusiones acerca de la calidad de las soluciones encontradas. En 2001, Kolingerova y Ferko [16] presentan una optimización genética cuyo operador de recombinación, denominado DeWall y el operador de mutación realiza un *flip* en el individuo seleccionado.

La complejidad del cálculo de MWT fue uno de los problemas abiertos más interesantes en Geometría Computacional hasta que en 2006, Mulzer y Rote [19], demostraron que es un problema NP-duro.

## 2.2 Pseudotriangulación de Peso Mínimo

Sea  $S$  un conjunto de puntos en el plano. Una pseudotriangulación de  $S$  es una partición del cierre convexo de  $S$  en pseudotriángulos cuyo conjunto de vértices es exactamente  $S$ . Un pseudotriángulo es un polígono que tiene exactamente tres vértices convexos. El peso de una pseudotriangulación  $PT$  es la suma de las longitudes Euclídeas de todas las aristas de  $PT$ . La pseudotriangulación que minimiza dicha suma se denomina *PseudoTriangulación de Mínimo Peso* (*Minimum Weight Pseudo-Triangulation*) de  $S$  y se denota por  $MWPT(S)$ .

El concepto de pseudotriangulación fue introducido por Pocchiola y Vegter in [20] en analogía a los arreglos de pseudorectas; para más información sobre resultados en pseudotriangulaciones remitirse al survey [22]. Como se mencionó en la Sección 1, existe un conjunto de puntos para los cuales cualquier triangulación tendrá un peso de  $O(n \cdot wt(M(S)))$ . La pregunta que surge entonces es si existen límites similares para el peor caso para las pseudotriangulaciones. Rote et al. [23] fueron los que se preguntaron si la pseudotriangulación de peso mínimo es un problema NP-duro, lo cual incentivó la búsqueda de algoritmos exactos o aproximados. Gudmundsson y Levcopoulos [11] consideraron el problema de calcular una pseudotriangulación de peso mínimo para un conjunto  $S$  de  $n$  puntos en el plano, presentando un algoritmo de tiempo  $O(n \cdot \log n)$  que produce pseudotriangulaciones de peso  $O(\log n \cdot wt(M(S)))$ , lo que es asintóticamente óptimo en el peor caso. Es decir, existe un conjunto  $S$  de puntos para los cuales cualquier pseudotriangulación tiene peso  $\Omega(\log n \cdot wt(M(S)))$ , donde  $wt(M(S))$  es el peso de un árbol generador mínimo de  $S$ .

## 3 Metaheurística Simulated Annealing

Una de las técnicas metaheurísticas más antiguas es la de *Recocido Simulado* (*Simulated Annealing, SA*), que se basa en la analogía con el proceso de templado

o enfriado controlado en la elaboración de ciertas sustancias como vidrio o aceros y el problema de resolver grandes problemas de optimización combinatoria. [18, 14, 2]

Para realizar dicha estrategia, SA introduce un parámetro de control  $T$ , designado por la *temperatura*, cuyo valor inicial debe ser alto y se debe decrementar durante el proceso de búsqueda. En dicho proceso se realiza la ejecución de varias iteraciones del algoritmo hasta que se alcanza la condición de finalización. El parámetro de control  $T$  permite, con cierta probabilidad, moverse a soluciones  $y$  del espacio de búsqueda  $S$  cuyos valores de función objetivo  $f(y)$  son peores que el valor de función objetivo  $f(x)$  de la solución actual  $x \in S$ . Estos movimientos se llaman ascendentes (*uphill moves*). A esta probabilidad se la denomina *función de aceptación* y se evalúa de acuerdo a:

$$p(T, x, y) = e^{-\frac{\delta}{T}} \quad (1)$$

Por la Ecuación (1), se puede observar que la probabilidad de aceptación de movimientos ascendentes es controlado por dos factores:  $T$  y  $\delta = f(y) - f(x)$ . Analizando la función podemos ver que: i) para un  $\delta$  fijo y bajas temperaturas, es menor la probabilidad de realizar movimientos ascendentes. Por lo tanto, a lo largo del proceso de búsqueda, es decir, de las iteraciones del algoritmo, es más difícil aceptar este tipo de movimientos; y ii) para una temperatura fija y altos valores de  $\delta$ , es menor la probabilidad de aceptar un movimiento de  $x$  a  $y$ . Notar que para valores iguales de  $\delta$ , es mayor la probabilidad de aceptar movimientos ascendentes para altas temperaturas. En el caso  $T = 1$ , se acepta cualquier función objetivo peor a la actual y en el caso  $T = 0$ , se rechaza cualquier valor de la función objetivo peor al de la solución actual y el algoritmo se convierte en un algoritmo iterativo de búsqueda local.

### 3.1 Algoritmo SA General

El Algoritmo 1 muestra la estructura básica de la estrategia SA.

El algoritmo comienza generando una solución inicial  $x \in S$  e inicializando la temperatura inicial  $T_0$ .  $N(T_k)$  es el número de iteraciones para cada temperatura  $T_k$  y en cada iteración se genera aleatoriamente una nueva solución  $y$  que forma parte del vecindario de  $x$ ,  $\mathcal{N}(x)$ . Si  $y$  es mejor que  $x$ , se acepta la nueva solución  $y$  y reemplaza a la solución actual. Si el movimiento de  $x$  a  $y$  es un movimiento ascendente,  $y$  se acepta con una probabilidad que se calcula de acuerdo a la Ecuación 1. Finalmente, el valor de  $T_k$  se disminuye en cada iteración  $k$  del algoritmo. Se continúa de esta manera hasta alcanzar la condición de finalización.

Por lo tanto, considerando un problema de optimización es necesario adaptarlo al esquema SA, el cual se obtiene especificando los siguientes parámetros: representación del espacio de soluciones, función objetivo  $f$ , vecindario de una solución  $\mathcal{N}(x)$ , solución inicial  $S_0$ , temperatura inicial  $T_0$ , regla de disminución de la temperatura  $\mathcal{R}$ , cantidad de iteraciones para cada temperatura  $N(T_k)$  y condición de finalización.

**Algoritmo 1** General-SA

---

```

Generar una solución inicial  $x \in S$ 
Determinar la temperatura inicial  $T_0$ 
 $k \leftarrow 0$ 
while No se alcance la condición de finalización do
   $i = 1$ 
  while  $i < N(T_k)$  do
    Generar  $y \in \mathcal{N}(x) \subset S$ 
    Evaluar  $\delta = f(y) - f(x)$ 
    if  $\delta < 0$  then
       $x \leftarrow y$ 
    else
       $x \leftarrow y$  con probabilidad  $p(T, x, y)$  // ver Ecuación 1
    end if
     $i \leftarrow i + 1$ 
  end while
   $k \leftarrow k + 1$ 
  Decrementar la temperatura  $T_k$ 
end while

```

---

**4 Simulated Annealing aplicado a MWT y MWPT**

Como se mencionó anteriormente, para resolver un problema de optimización utilizando la metaheurística SA es necesario identificar los parámetros de la estrategia SA. A continuación se describen estos parámetros para los problemas MWT y MWPT.

*Espacio de soluciones  $S$* : Para el problema MWT, el espacio de soluciones  $S$  está representado por triangulaciones y para MWPT está formado por pseudotriangulaciones de un conjunto  $P$  de  $n$  puntos en el plano. Para ambos tipos de soluciones se utiliza una representación de matrices de  $n \times n$  de ceros y unos, donde un 1 en la posición  $(i, j)$  significa que existe una arista que une los puntos  $i$  y  $j$ ; en caso contrario se coloca un 0.

*Función objetivo  $f$* : La función objetivo  $f : S \rightarrow \mathbb{R}$  asigna a cada elemento de  $S$  un valor real. Para cada  $S_i \in S$ , la función  $f$  se define como la suma de las longitudes euclídeas de las aristas de la solución  $S_i$ , es decir, que se suman todas las aristas representadas por un 1 en la matriz de representación de  $S_i$ .

*Solución inicial  $S_0$* : Considerando los problemas MWT y MWPT, y un conjunto  $P$  de  $n$  puntos en el plano, el algoritmo SA puede comenzar a ejecutar con distintos tipos de solución inicial: una triangulación/pseudotriangulación aleatoria, una triangulación/pseudotriangulación greedy, la triangulación de Delaunay, o el resultado del algoritmo ACO [6] aplicado al conjunto  $P$ .

*Temperatura inicial  $T_0$* : la temperatura inicial depende del número  $m$  de aristas de la solución inicial y de la medida de calidad que se está considerando en los problemas, es decir,  $T_0 = m \times l$ , donde  $l$  es la longitud media de las aristas de  $S_0$ ; y (ii) una temperatura inicial constante, es decir,  $T_0 = c$ , con  $c \in \mathbb{N}$ .

*Regla de disminución de la temperatura  $\mathcal{R}$* : Se utilizan tres tipos diferentes de reglas: (i) decremento rápido ( $T_{k+1} = \frac{T_0}{(1+k)}$ ); (ii) decremento muy rápido ( $T_{k+1} = \frac{T_0}{e^k}$ ); (iii) y decremento geométrico ( $T_{k+1} = \alpha T_k$  con  $\alpha = 0.8, 0.9$  y  $0.95$ ).

*Cantidad de iteraciones para cada temperatura  $N(T_k)$* : Se utiliza  $N(T_k) = T_k$  para asegurar que se realicen más iteraciones para temperaturas altas, y menos iteraciones cuando se está cerca de la solución óptima.

*Condición de finalización*: Se termina el proceso de búsqueda cuando la temperatura es menor o igual a 0.005, es decir,  $T_f = 0.005$ . Las soluciones obtenidas en temperaturas bajas se aproximan a las óptimas, pero el tiempo de respuesta del algoritmo se incrementa considerablemente.

*Vecindario de cada solución*: Para cada solución  $S_i \in S$  se debe obtener un elemento  $S_j \in S$ , llamado *vecino*. En nuestro caso, para los problemas MWT y MWPT, tenemos diferentes operadores para generar un vecino para una solución, los cuales se describen en las siguientes subsecciones.

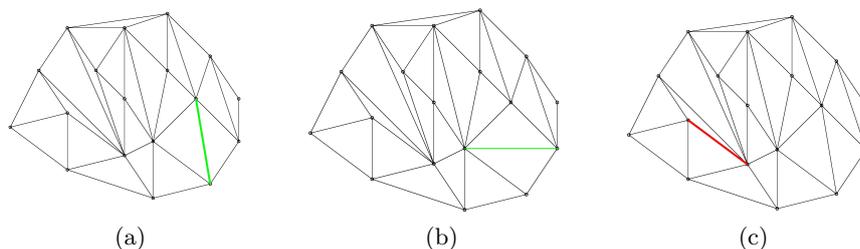
#### 4.1 Cálculo de Vecindario para MWT

*Intercambiar (Flip)*: Dados dos triángulos  $t_1$  y  $t_2$  que comparten una arista  $a$  en una triangulación  $T$ , la operación de intercambio sobre la arista  $a$  cambia  $T$  por otra triangulación  $T'$  donde: los triángulos fuera del cuadrilátero  $t_1 \cup t_2$  son los mismos y en  $t_1 \cup t_2$  se toma la diagonal opuesta a  $a$ .

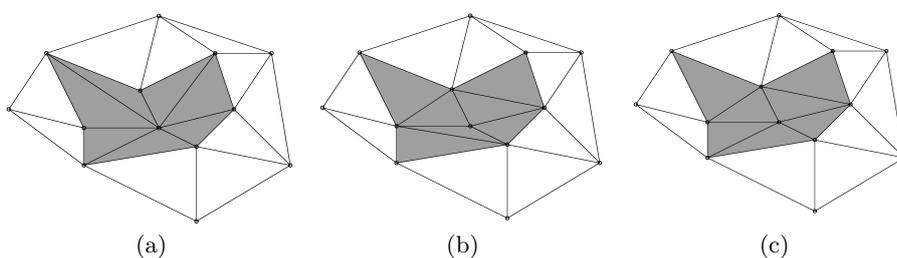
Para MWT se elige aleatoriamente una arista  $a$  de la solución actual y se realiza la operación de *flip* sobre  $a$ , siempre que sea posible. En caso de no poder realizar el *flip*, debido a que la arista es ilegal, se elige nuevamente una arista de manera aleatoria y se repite dicha operación. (ver Figura 1)

Además, en este cálculo tomamos la idea de los  $\beta$ -*skeleton*: sean  $p$  y  $q$  dos puntos distintos del conjunto de puntos  $S$  con  $\beta > 1$ . Siguiendo a Kirkpatrick y Radke [13], la arista  $pq$  se incluye en el  $\beta$ -*skeleton*  $\beta(S)$  de  $S$  si los dos círculos de diámetro  $\beta \cdot |pq|$  y que pasan a través de ambos puntos,  $p$  y  $q$ , no contienen ningún otro punto de  $S$ . Para nuestro trabajo es interesante notar que  $\beta(S)$  es un subgrafo de cualquier  $MWT(S)$ , para  $\beta$  demasiado grande. La cota original  $\beta \geq \sqrt{2}$  en Keil [12] fue mejorada por Cheng y Xu [3] a  $\beta > 1.1768$ . Por lo tanto, se sabe que las aristas del *Grafo de Vecindad Relativa (Relative Neighborhood Graph, RNG)*, llamado *2-skeleton*, están contenidas en MWT. En consecuencia, no se permiten intercambiar aristas que pertenecen al RNG.

*Retriangular localmente*: Se elige aleatoriamente un vértice  $u$  de la solución actual y se recuperan todos los vértices adyacentes a  $u$ . Luego se recupera el polígono que forman los vértices adyacentes a  $u$  y se retriangula el interior de ese polígono junto con el vértice  $v$ . Esta zona se puede retriangular de manera aleatoria o de manera ávida (*greedy*). En la primera, se van insertando al azar las aristas de dicha zona mientras no se intersequen con las agregadas anteriormente y en la segunda manera, se ordenan todas las aristas de la zona considerando su longitud y se van insertando en ese orden mientras no intersequen con las agregadas anteriormente. (ver Figura 2)



**Fig. 1.** En la figura se muestra una triangulación a) sobre la cual se realiza un flip (arista verde) y se muestra la triangulación resultante b). En c) se muestra una arista ilegal (en rojo) a la cual no se le puede realizar un flip.



**Fig. 2.** En la figura se muestra un ejemplo a) donde se indica la zona de color gris que es retriangulada de manera b) aleatoria y c) greedy.

#### 4.2 Cálculo de Vecindario para MWPT

También en el cálculo de vecindario para una pseudotriangulación se utiliza el concepto de flip adaptado a este tipo de configuraciones geométricas. Se elige aleatoriamente una arista  $e$  de la solución actual y se realiza la operación de *flip* sobre  $e$ , siempre que sea posible. En caso de no poder realizar el flip, se elige nuevamente una arista de manera aleatoria y se repite dicha operación. Dados dos pseudotriángulos  $t_1$  y  $t_2$  que comparten una arista  $e$  en una pseudotriangulación  $T$ , la operación de intercambio sobre la arista  $e$  cambia  $T$  por otra pseudotriangulación  $T$ . Antes, hay que referirse a camino geodésico: en un polígono simple, el camino más corto entre cualquier par de vértices,  $p$  and  $q$ , el cual no interseca con ninguna arista se llama camino geodésico entre  $p$  y  $q$ . Este camino tiene la propiedad de ser único. En un polígono con cuatro vértices convexos, el camino geodésico entre un par de vértices es o una arista presente o es una arista ausente. Para una pseudotriangulación  $T$ , en el caso de dos pseudotriángulos  $t_1$  y  $t_2$  que comparten una arista  $e$ , se tiene que los pseudotriángulos fuera del pseudocuatrilátero  $t_1 \cup t_2$  son los mismos y en  $t_1 \cup t_2$  es un polígono simple, que puede ser un pseudocuatrilátero (cuatro vértices convexos solamente), o es un pseudotriángulo.

Considerando  $u$  el vértice convexo de  $t_1$  opuesto al lado que contiene a  $e$ , y  $v$  el vértice convexo de  $t_2$  opuesto al lado que contiene a  $e$ , se tiene: *i*) si el camino

geodésico entre  $u$  y  $v$  existe entonces se trata de un pseudotriángulo y la arista  $e$  no tiene dual (Caso eliminación de arista). Una arista de una pseudotriangulación es eliminable si su eliminación resulta una pseudotriangulación. En este sentido, hallar una pseudotriangulación vecina se puede hacer mediante la operación eliminación de arista siempre que su resultado sea una pseudotriangulación. *ii*) en otro caso, el camino geodésico completo no está presente, y por tanto una arista  $e'$  no presente es la dual a  $e$ , por lo que el flip consiste en reemplazar dicha arista  $e$  por la arista  $e'$ , resultando la nueva configuración geométrica en una nueva pseudotriangulación.

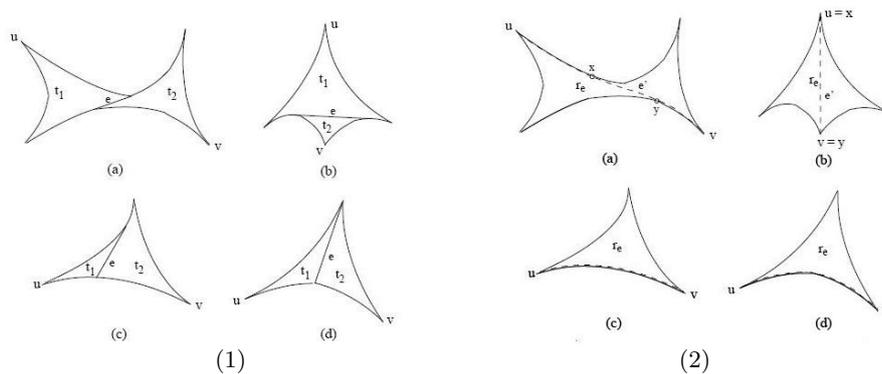


Fig. 3. (1) Casos de flip y (2) sus resoluciones

## 5 Conclusiones y trabajos futuros

Luego de realizar una revisión de los artículos sobre los problemas planteados en este trabajo no se han presentado colecciones de instancias (conjunto de puntos) sobre las cuales se pueda comprobar el desempeño de la estrategia SA. Por lo tanto, se diseñó un *generador de instancias* con el cual se generó una colección de 10 instancias de tamaño 40/80/120/160/200 respectivamente, es decir, un total de 50 instancias de problema. Para mayores detalles sobre el generador remitirse a [6].

Hasta la actualidad, el diseño se ha completado, iniciando la etapa de evaluación experimental. Considerando este aspecto, los resultados iniciales han arrojado perspectiva acerca del posible comportamiento de la técnica, mostrando que se requiere calibrar los parámetros tales como temperaturas iniciales y restricciones sobre las operaciones que generan los vecindarios.

A continuación, se muestran resultados preliminares para MWT, considerando cuatro lotes de 40 puntos, y utilizando el operador vecindario Intercambio.

En este artículo hemos presentado el diseño de algoritmos SA para la obtención de soluciones aproximadas a los problemas MWT y MWPT, mostrando

**Table 1.** Resultados para LD401.

Config.	Media	Mediana	Mejor	Varianza
T1-D1-V1	5566047	5568660	5536413	11935
T1-D4-V1	5724891	5728380	5636856	32787
T1-D5-V1	5708934	5717689	5646367	30775
T1-D3-V1	5736384	5749597	5654222	39227
T1-D2-V1	5775527	5779485	5684696	41487

**Table 3.** Resultados para LD403.

Config.	Media	Mediana	Mejor	Varianza
T1-D1-V1	5615611	5619349	5574341	18899
T1-D5-V1	5776851	5787095	5693195	40950
T1-D3-V1	5821193	5828344	5721135	48539
T1-D4-V1	5793932	5801214	5734212	37654
T1-D2-V1	5865095	5874604	5740253	45472

**Table 2.** Resultados para LD402.

Config.	Media	Mediana	Mejor	Varianza
T1-D1-V1	4746014	4749184	4710618	13889
T1-D4-V1	4916429	4923230	4808472	43400
T1-D5-V1	4912487	4922510	4846340	34179
T1-D2-V1	4994352	4998591	4856144	48050
T1-D3-V1	4947066	4946020	4856144	38256

**Table 4.** Resultados para LD404.

Config.	Media	Mediana	Mejor	Varianza
T1-D1-V1	5909888	5915225	5862851	17807
T1-D4-V1	6156295	6174940	6015541	52692
T1-D5-V1	6133936	6140354	6037112	38510
T1-D3-V1	6199571	6200206	6058921	61216
T1-D2-V1	6257826	6257978	6103159	77139

cuál será el escenario paramétrico con el cual se realiza la etapa experimental para luego poder verificar con qué valores de parámetros se obtienen mejores soluciones.

## References

1. K. Capp and B. Julstrom. A weight-coded genetic algorithm for the minimum weight triangulation problem. In *SAC*, pages 327–331, 1998.
2. V. Černý. Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm. *Journal of Optimization Theory and Applications*, 1985.
3. Siu-Wing Cheng and Yin-Feng Xu. Approaching the largest  $\beta$ -skeleton within a minimum weight triangulation. In *SCG '96: Proceedings of the twelfth annual symposium on Computational geometry*, pages 196–203, New York, NY, USA, 1996. ACM.
4. M. Dorzán, E. Gagliardi, M. Leguizamón, and G. Hernández Peñalver. Algoritmo *ACO* aplicado a la obtención aproximada de triangulaciones de peso mínimo. In *XXXV Conferencia Latinoamericana de Informática*, 2009.
5. M. Dorzán, E. Gagliardi, M. Leguizamón, and G. Hernández Peñalver. Approximations on minimum weight pseudo-triangulations using ant colony optimization metaheuristic. In *XV Congreso Argentino de Ciencias de la Computación*, 2009.
6. M. Dorzán, E. Gagliardi, M. Leguizamón, and G. Hernández Peñalver. Approximations on minimum weight triangulations and minimum weight pseudo-triangulations using ant colony optimization metaheuristic. In *I Workshop on Emergent Computing, Jornadas Chilenas de Computación*, 2009.
7. M. Dorzán, E. Gagliardi, M. Leguizamón, and G. Hernández Peñalver. Soluciones aproximadas para el problema de triangulación de peso mínimo utilizando *ACO*. In *XV Congreso Argentino de Ciencias de la Computación*, 2009.
8. R. Düppe and H. Gottschalk. Automatische interpolation von isolinien bei willkürlichen stützpunkten. *Allgemeine Vermessungsnachrichten*, 77:423–426, 1970.

9. M. Garey and D. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979.
10. P. Gilbert. New results in planar triangulations. Technical Report R-850, Univ. Illinois Coordinated Science Lab., 1979.
11. J. Gudmundsson and C. Levcopoulos. Minimum weight pseudo-triangulations. *Comput. Geom.*, 38(3):139–153, 2007.
12. J. Mark Keil. Computing a subgraph of the minimum weight triangulation. *Comput. Geom. Theory Appl.*, 4(1):13–26, 1994.
13. David G. Kirkpatrick and John D. Radke. A framework for computational morphology. In G. T. Toussaint, editor, *Computational Geometry*, pages 217–248. NH, Amst, 1985.
14. S. Kirkpatrick, C. Gelatt, and M. Vecchi. Optimization by simulated annealing. *Science*, 220:671–680, 1983.
15. G. Klincsek. Minimal triangulations of polygonal domains. *Ann. Disc. Math.*, 9:121–123, 1980.
16. I. Kolingerová and A. Ferko. Multicriteria-optimized triangulations. *The Visual Computer*, 17(6):380–395, 2001.
17. E. Lloyd. On triangulations of a set of points in the plane. In "Proc. 18th IEEE Symp. Found. Comp. Sci.", pages 228–240, 1977.
18. N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller. Equation of state calculations by fast computing machines. *Journal of Chemical Physics*, 21:1087–1092, 1953.
19. W. Mulzer and G. Rote. Minimum weight triangulation is np-hard. In *In Proc. 22nd Annu. ACM Sympos. Comput. Geom.*, pages 1–10. ACM Press, 2006.
20. M. Pocchiola and G. Vegter. Pseudo-triangulations: Theory and applications. In *Symposium on Computational Geometry*, pages 291–300, 1996.
21. K. Qin, W. Wang, and M. Gong. A genetic algorithm for the minimum weight triangulation. In *In Proceedings of 1997 IEEE International Conference on Evolutionary Computation*, 1997.
22. G. Rote, F. Santos, and I. Streinu. *Pseudo-triangulations — a survey*. Contemporary Mathematics. American Mathematical Society, December 2008.
23. G. Rote, C. Wang, L. Wang, and Y Xu. On constrained minimum pseudotriangulations. In *Proc. 9th Intern. Comp. Comb. Conf.*, 2003.
24. M. Shamos and D. Hoey. Closest-point problems. In *Proc. 16th IEEE Symp. Foundations of Comp. Science*, pages 151–162, 1975.
25. Y. Wu and R. Wainwright. Near-optimal triangulation of a point set using genetic algorithms. In *Proceedings of the Seventh Oklahoma Conference on AI.*, 1993.