

Trazados ortogonales en el plano. Resultados generales y aproximación por algoritmos genéticos

M. A. Garrido, A. Márquez, J. R. Portillo y P. Reyes

Dept. de Matemática Aplicada I
Universidad de Sevilla (España)
{vizuete,almar,josera,preyes}@cica.es

Los trazados ortogonales se utilizan en multitud de aplicaciones de Electrónica-Informática, como por ejemplo en el trazado de circuitos VLSI o diseño de bases de datos. Este interés práctico ha llevado a la obtención de multitud de resultados sobre la materia [1, 2, 3, 5, 7, 8]. En muchas ocasiones las condiciones del problema nos obligan a trabajar con coordenadas fijas en el plano, teniendo que unir diferentes conjuntos de puntos mediante trazados ortogonales y sin intersecciones entre las conexiones.

El primer problema abordado en este campo es el de estudiar si se pueden conectar entre sí n parejas de puntos, mediante trazados ortogonales, utilizando a lo sumo un codo y sin que se produzcan intersecciones entre ellos. Este problema, conocido como *Single Bend Wiring*, fue resuelto por Raghavan et al. [7] en tiempo cuadrático en el número de parejas. Así mismo prueban que el problema de encontrar el máximo número de parejas que pueden ser trazadas en estas condiciones es NP -duro.

Dado que las aplicaciones prácticas presentan situaciones más complejas, generalizaremos el estudio considerando diferentes conjuntos de puntos y variando el número de codos. Así enunciamos:

Trazados ortogonales $TO(G, k)$

ENTRADA: Un grafo G , un número entero positivo k y n conjuntos de puntos en el plano W_i ($i = 1, \dots, n$) de cardinalidad igual al número de vértices de G .

PREGUNTA: ¿Pueden unirse simultáneamente todos los puntos de cada conjunto W_i , mediante una inmersión ortogonal de G en la retícula del plano, con a lo sumo k codos y sin intersecciones?

Evidentemente el problema reseñado en [7] es el caso particular $TO(K_2, 1)$.

Con el objeto de obtener resultados generales, comenzamos estudiando el problema para grafos particulares y número de codos prefijado. En función de la naturaleza de cada caso, las técnicas empleadas en su resolución han sido distintas, en algunos casos dentro del marco de la Geometría Computacional y en otros relacionándolos con problemas de satisfacibilidad.

El siguiente cuadro nos ofrece algunos de los resultados obtenidos en casos particulares de este problema:

¹Parcialmente financiado por los grupos de investigación PB96-1374, incluido en el proyecto DGICYT y FQM—0164, adscrito al proyecto PAI

$G \setminus b$	0	1	2	3	4	5	12
K_2	$n \log n$	n^2	NP
K_3	NO	$n \log n$	n^2	NP
P_2	$n \log n$	—	NP
P_3	n^2	—	NP
$K_{1,3}$	$n \log n$	n^2	NP
C_4	$n \log n$	n^2	NP
P_4	—	—	NP
C_5	$n \log n$	n^2	NP

Tabla 1: Complejidad computacional para $\text{TO}(G, k)$

Siguiendo esta línea, se han obtenido resultados para grafos generales, tales como 2—conexos y 1—camino. Por otro lado, tal y como podemos observar, se ha demostrado la naturaleza NP —completa de varios problemas, lo que sugiere buscar métodos de aproximación para los problemas de optimización asociados.

La versatilidad de los algoritmos genéticos y la gran cantidad de resultados obtenidos como fruto de su aplicación en otros campos [4, 6] nos llevan a utilizar esta metodología en nuestro estudio.

En este sentido, el primero de los problemas de optimización que abordamos es el asociado a $\text{TO}(K_2, 1)$, cuya naturaleza NP —dura se ha probado en [7].

La formulación de este problema es la siguiente:

Problema Max—Single Bend Wiring MAX—SBW

ENTRADA: Un conjunto de n parejas de puntos en el plano $\{w_1, w_2, \dots, w_n\}$.

PREGUNTA: ¿Cuál es el número máximo de parejas que pueden unirse mediante trazados ortogonales, con a lo sumo un codo y sin que se produzcan intersecciones entre ellas?

Utilizamos en la implementación del problema un algoritmo genético simple (todas las generaciones tienen el mismo número de individuos) y en su modelización definimos un genoma (o individuo) como un conjunto de n parejas de puntos $\{w_{p_1}, w_{p_2}, \dots, w_{p_n}\}$ donde $p_1 p_2 \dots p_n$ es una permutación con repetición de orden n de los elementos $1, 2, \dots, n$.

Puesto que nuestro objetivo final es aproximar el número máximo de parejas de puntos que se pueden unir bajo las condiciones anteriores, la función objetivo sobre cada individuo debe aproximar tal número. Con este objeto elegimos como valor de la función objetivo el número de parejas de puntos consecutivas (con carácter cíclico y evitando las repeticiones) que puedan trazarse. Para el cálculo de este valor hemos diseñado e implementado un algoritmo incremental que realiza, en cada paso, una reducción a un problema de 2—satisfacibilidad.

El cruce utilizado es el “cruce en un sólo punto”: Dados dos padres $\{w_{p_1}, w_{p_2}, \dots, w_{p_n}\}$ y $\{w'_{p_1}, w'_{p_2}, \dots, w'_{p_n}\}$ y un número r ($1 \leq r \leq n - 1$), elegidos arbitrariamente, los hijos se obtienen de la siguiente forma:

$$\begin{array}{|c|c|c|c|c|c|} \hline a_1 & \dots & a_r & a_{r+1} & \dots & a_n \\ \hline b_1 & \dots & b_r & b_{r+1} & \dots & b_n \\ \hline \end{array} \longrightarrow \begin{array}{|c|c|c|c|c|c|} \hline a_1 & \dots & a_r & b_{r+1} & \dots & b_n \\ \hline b_1 & \dots & b_r & a_{r+1} & \dots & a_n \\ \hline \end{array}$$

La mutación consistirá en la sustitución de uno de los genes del genoma por otro cualquiera (mutación por sustitución).

$$\begin{array}{|c|c|c|c|c|} \hline a_1 & \dots & a_r & \dots & a_n \\ \hline \end{array} \longrightarrow \begin{array}{|c|c|c|c|c|} \hline a_1 & \dots & b_r & \dots & a_n \\ \hline \end{array}$$

El soporte utilizado ha sido el lenguaje C++, usando en algunos casos librerías existentes sobre algoritmos genéticos y en otros librerías propias.

En la siguiente tabla mostramos los resultados obtenidos para configuraciones de parejas de puntos generadas aleatoriamente, experimentando con poblaciones de distintos tamaños y realizando diferente número de generaciones. En función de estos dos parámetros, indicamos el número máximo de parejas de puntos obtenido y el tiempo de ejecución, con un procesador Pentium III-500 Mhz., expresado en segundos.

Valor máximo y tiempo de ejecución					
Gen. \ Parejas	20	40	60	80	100
0	11 (0)	12 (0)	14 (0)	15 (1)	18 (2)
200	13 (17)	19 (73)	25 (247)	22 (430)	27 (722)
400	13 (36)	20 (153)	27 (516)	26 (905)	30 (1535)
600	13 (55)	20 (238)	27 (792)	27 (1394)	30 (2366)
800	13 (75)	20 (328)	27 (1078)	27 (1901)	33 (3255)
1000	13 (94)	20 (411)	27 (1331)	27 (2426)	33 (4141)
2000	13 (189)	20 (860)	28 (2740)	28 (4989)	34 (8619)
3000	13 (283)	20 (1323)	28 (4156)	30 (7673)	34 (13218)
4000	13 (380)	20 (1734)	28 (5560)	30 (10383)	34 (18013)

Tabla 2: Resultados experimentales para diferentes poblaciones y generaciones.

Actualmente continuamos realizando mejoras al algoritmo, considerando otro tipo de operadores genéticos, aplicando elitismo, etc.

Como problema abierto más general nos planteamos la comparación de los resultados obtenidos mediante técnicas genéticas, con otros métodos como heurísticas deterministas, búsqueda tabú, etc.

Referencias

- [1] T. C. Biedl. Optimal orthogonal drawings of triconnected plane graphs. Proc. of SWAT'96, LNCS **1097** Springer—Verlag (1996), 333—344.
- [2] M. A. Garrido y A. Márquez. Embedding a graph in the grid of a surface with the minimum number of bends is *NP*—hard. En G. DiBattista, editor, Graph Drawing (Proc. of GD'97), LNCS **1353** Springer—Verlag (1998), 124—133.
- [3] M. A. Garrido, A. Márquez, A. Morgana y J.R. Portillo. Single bend wiring on surfaces. *submitted* (1998), 12 páginas.
- [4] A. Kapsalis, V.J. Rayward-Smith y G.D. Smith. Solving the graphical Steiner tree problem using genetic algorithms. Journal Operations Research **44**(4) (1993), 397—406.
- [5] Y. Liu, A. Morgana y B. Simeone. A linear algorithm for 2—bend embeddings of planar graphs in the two—dimensional grid. Discrete Applied Mathematics, **8** (1998), 69—91.
- [6] Z. Michalewicz. Genetic Algorithms + Data Structures = Evolution Programs. Springer, (1999).
- [7] R. Raghavan, J. Cohoon y S. Sahni. Single Bend Wiring. Journal of Algorithms, **7** (1986), 232—257.
- [8] R. Tamassia. On embedding a graph in the grid with the minimum number of bends. SIAM J. Comput. **16**(3) (1987), 421—444.