La Importancia de Saber Ubicar el Centro Absoluto de una Red en Problemas de Localización Óptima de Instalaciones

Dexmont A. Peña Carrillo
Anel B. Reyes Ramírez
Ruth M. Ávila Guerrero
Roger Z. Ríos Mercado
{dexmont, anel, marlen, roger}@yalma.fime.uanl.mx

Resumen

La investigación de operaciones es la aplicación, por grupos interdisciplinarios, del método científico a problemas relacionados con el control de las organizaciones o sistemas, a fin de que se produzca soluciones que mejor sirvan a los objetivos de la organización. Dentro del campo, existen una serie de aplicaciones, comúnmente conocidos como problemas de localización en redes, en las cuales resulta de marcada trascendencia saber encontrar en forma efectiva el centro de las mismas. Por centro nos referimos a un punto de la red en el cual la distancia a su nodo más alejado sea la menor posible. La importancia de encontrar el centro absoluto en una red es la localización óptima de un servicio de emergencia, tal como ambulancias, estación de bomberos y estación de policía en una transportación de red. En este trabajo se ilustra una implementación basada en el algoritmo de Dvir y Handler para encontrar dicho centro en una red dada. Las pruebas de la implementación se realizaron con diferentes instancias en distintas magnitudes de nodos y arcos para comprobar la eficiencia del algoritmo. Como se comprueba empíricamente, el algoritmo funciona muy eficientemente para redes relativamente grandes.

Abstract

Operations research is commonly referred as the science of decision-making, and it attempts to provide with quantitative and analytical tools to model, analyze, and solve a number of decision-making problems arising in many industrial settings. One of the operations research fields deal with location problems in networks, for which is very important to find absolute centers. The absolute center of a network is the point for which the maximum of the minimum distance from that node to all other nodes in the network is minimized. Applications of this problem include optimal location of emergency services such as ambulances, fire stations, and police stations. In this work, an implementation of the Dvir-Handler algorithm for finding absolute centers in a graph is presented and illustrated. Empirical evidence based on relatively large instances illustrates the effectiveness of the method.

Palabras clave: Investigación de operaciones; optimización de flujo en redes; localización, centro absoluto de una red.

Keywords: Operations research; network flow programming; location; absolute center of a network.

Introducción

La investigación de operaciones es la ciencia que brinda soporte a problemas de toma de decisiones que surgen en los diversos ámbitos industriales. En particular, uno de los subcampos de gran interés tanto práctico como científico es el de los problemas de localización [1, 2]. Un problema de localización típico tiene que ver con dónde ubicar/construir instalaciones para brindar un determinado tipo de servicio. El problema se modela normalmente como una red donde los nodos de la misma corresponden a clientes y/o puntos potenciales de ubicación de las instalaciones, y los arcos (que unen a estos nodos) corresponden con relaciones entre dichos puntos, típicamente relaciones de distancia o costo. En este proceso, resulta de marcada trascendencia saber encontrar en forma efectiva el *centro* de las mismas. Por centro nos referimos a un punto de la red en el cual la distancia a su nodo más alejado sea lo menor posible. El significado físico es que una instalación ubicada en este nodo centro tiene la característica que, al compararlo con cualquier otro nodo, la distancia recorrida al nodo más alejado siempre será la menor posible. Técnicamente, se dice que cumple con un criterio min-max, es decir, minimizar la peor (max) de las distancias.

En este trabajo se trata el problema de como ubicar el centro absoluto una red. El problema de encontrar un centro absoluto de una red fue introducido por Hakimi [3]. Entre las aplicaciones prácticas de dicho problema en problemas de localización, destacamos por ejemplo: localización de instalaciones de servicios de emergencia [4], localización de centros en problemas de diseño de territorios [5], por mencionar algunas. Uno de los métodos más recientes para encontrar el centro absoluto en una red es el de Dvir y Handler [6]. En nuestro trabajo llevamos a cabo una implementación computacional del algoritmo de Dvir-Handler, e ilustramos su utilidad en la solución de algunas instancias de red relativamente grandes.

Nuestro trabajo está organizado de la siguiente forma. En las primeras dos secciones introducimos algunos conceptos básicos y marco teórico con el fin de darle claridad a la explicación del algoritmo de solución, el cual se presenta posteriormente. Ese mismo capítulo contiene un ejemplo que ilustra paso a paso la aplicación del método. Finalmente presentamos una evaluación empírica del método al aplicarlo a la solución de algunas instancias del problema.

Nomenclatura y conceptos básicos

Como se comenta anteriormente nuestro trabajo se basa en el algoritmo desarrollado por Dvir y Handler [4]. La implementación de este algoritmo para encontrar el centro absoluto de una red necesita como entrada la matriz de distancias más cortas de un grafo dado. En este trabajo se utiliza el algoritmo Floyd-Warshall (véase [7]) para obtener la matriz de distancias más cortas, posteriormente se utiliza el algoritmo de Dvir-Handler para encontrar el centro absoluto de la red.

A continuación reproducimos algunos conceptos introducidos por Dvir y Handler [6] y algunas conceptos básicos de redes [7] con el fin de darle claridad al material presentado. Sea $G=(V,A)$ un grafo compuesto por un conjunto de nodos $V = \{v_1, v_2, \ldots, v_n\}$ y un conjunto de arcos no dirigidos A con cardinalidad m. Denotamos la conexión de arco v_i a v_j como (v_i, v_j) o simplemente (i,j) y asociamos una longitud positiva d_{ij}. Sea $d(x,y)$ la distancia más corta entre dos puntos x, y en V, y $p(x,y)$ una correspondiente trayectoria de ruta más corta. La distancia $\max\{d(x,y)$: y en $V\}$ para un punto x en G a el nodo más lejano será denotada por $l(x)$. Se define ahora el centro vértice (VC) y el centro absoluto (AC). VC es un nodo que satisface $l(\mathrm{VC})= \min\{l(x)$: x en $V\}$, mientras AC es un punto en G tal que $l(\mathrm{AC}) = \min\{l(x)$: x en $G\}$.

El valor $r(G)= l(\mathrm{AC})$ es conocido como el radio de G. El diámetro de un grafo se define como el máximo de las distancias de las rutas mas cortas entre un par de nodos, esto es $\max\{l(x)$:x en $V\}$. Se define el diámetro $d(G)$ de la red como dos veces su radio, llamados $d(G)=2r(G)$. En general se conoce como $r(G)\leq \max\ l(x) \leq d(G)$ y para una red de árbol T, se puede ver que $2r(T) = \max\{\ l(x)=d(T)\}$. Supongamos ahora que T denota algún árbol de expansión de G. Luego, se define un mínimo diámetro de árbol (MDT) de G como un árbol de expansión de G tal que $d(\mathrm{MDT}) = \min\{d(T)$: T *en* $\Gamma(G)\}$, donde $\Gamma(G)$, es el conjunto de todos los árboles de expansión de G. Se define el árbol de trayectoria mínima enraizado a x (MPT(x)) como el árbol de expansión de trayectoria más corta de G desde un punto x en G.

Una alternativa para acelerar la solución ha sido encontrar cotas inferiores para el diámetro local de algún arco. La mejor cota, denominada $2l(x^*) \geq LB_{rs} \equiv d_{rs} + d(s,p) + d(r,q)$, donde X^* es un centro local para el arco (r,s) y v_p, v_q son los nodos mas lejanos de v_r, v_s respectivamente. El 95% de los arcos pueden ser eliminados comparando esta cota a una cota superior como UB = $2l$(VC).

La idea principal del algoritmo es tener un procedimiento iterativo exacto para identificar un MDT local asociado con un arco dado (r,s) de la red. Para el caso de una red cíclica, la cota de Halpern rinde una cota inferior para el diámetro de un árbol de expansión mínimo centrado en un arco particular *(r,s)* de la red. Comenzando con la cota de Halpern, el método genera una secuencia de cotas inferiores LB_{rs} hasta que el centro local es determinado (o hasta que $LB_{rs} \geq$ UB).

Fundamentos teóricos

Los siguientes conceptos son tomados de [6] para ayudar a la comprensión y construcción del algoritmo el cual será presentado mas adelante.

Suponga que un centro absoluto (AC) existe sobre un arco. La cota inferior de Halpern es la cota para el diámetro local de un grafo, utilizando esta definición se puede actualizar esta cota hasta que el diámetro del grafo es alcanzado, sin embargo, no conocemos el hecho de que un arco dado (r,s) incluye un AC, entonces hay dos posibilidades para abordar esto:

1) Si el arco contiene un AC. El diámetro local es el diámetro de G y el punto medio de la ruta diametral es un AC.

2) Si el arco no contiene un AC. Si se ha encontrado el diámetro de algún árbol de expansión de G, el cual es una cota superior de el diámetro de G, entonces si se repite este procedimiento para todos los arcos (r,s) en A (conjunto de arcos en el grafo) se garantiza encontrar el diámetro de G como el mínimo de todos los diámetros locales.

A continuación se presenta un lema y tres teoremas que servirán para el procedimiento de diámetro local basado en arco (r,s). La demostración se encuentra en [6].

Para un arco dado (r,s) suponga que v_p y v_q son los vértices mas lejanos a los vértices v_r y v_s respectivamente.

Lema: Si un AC es localizado sobre el interior de un arco (r,s) entonces la ruta mas corta entre AC y v_p no puede ir a través de v_r .

Teorema 1: si para un arco (r,s) existe v_p y v_q tales que son los mismos entonces el interior de este arco no puede contener un AC.

***Teorema 2*:** (Cota de Halpern) Sí un AC esta localizado sobre un arco (r,s) entonces

$$d(G) \geq LB_{rs} \equiv d_{rs} + d(s,p) + d(r,q)$$

donde $d(G)$ es el diámetro del grafo.

d_{rs} la longitud de la ruta mas corta de el vértice r a el vértice s.

$d(s,p)$ la longitud de la ruta mas corta de s a p.

$d(r,q)$ la longitud de la ruta mas corta de r a q.

A continuación se presenta el más importante teorema, el cual nos permite actualizar LB_{rs} hasta que el diámetro actual es alcanzado, suponiendo que el AC existe en el interior del arco.

En general, la idea básica es encontrar exitosamente un nuevo nodo, el cual es el más lejano que v_p y v_q de AC. Específicamente, se selecciona un nodo más lejano a AC en una dirección, digamos a través de v_r lo cual implica que estará conectado al AC (sobre la ruta mínima del árbol enraizado en AC) en la dirección opuesta v_s.

Ahora supongamos que $v_{c(r)}$ y $v_{c(s)}$ son el par de vértices actuales más cercanos conectados a AC vía v_r y v_s respectivamente. Esto significa que si hay un AC en el interior del arco considerado entonces hay una ruta más corta de AC a $v_{c(r)}$ que va a través de v_r, de la misma manera la ruta más corta de AC a $v_{c(s)}$ que va a través de v_s, entonces la cota inferior asociada con este par de nodos es $LB_{rs} \equiv d_{rs} + d(r,c(r)) + d(s,c(s))$.

Se supone que para un arco dado *(r,s)* y un nodo v_k se definen dos diferencias $\delta_{rk} = d(r,k) - d(r,c(r))$, $\delta_{sk} = d(s,k) - d(s,c(s))$. Posteriormente de define δ_{k*} = max{ max{ δ_{rk}, δ_{sk}}: $k \in K$} como la máxima diferencia, donde $K = \{k\text{: } \delta_{rk} > 0,\ \delta_{rs} > 0,\ k = 1, 2, 3, \ldots n\}$ y si K=Ø se define $\delta_{k*} = 0$.

Teorema 3: Suponga que un AC esta localizado en el interior de un arco (*r,s*) y suponga que se tiene un par actual de nodos mas lejanos $v_{c(r)}$, $v_{c(s)}$ de un AC asociado a una cota inferior .Construyendo los pares de la diferencia para cada nodo v_k, se distinguen tres casos:

1) Si $\delta_{k*} = 0$ entonces el diámetro $d(G) = LB_{rs}$

2) Si $\delta_{k*} = \delta_{rk*} = \delta_{sk*} > 0$ entonces $d(G) = LB_{rs} + \delta_{k*}$

3) Sin perdida de generalidad $\delta_{k*} = \delta_{rk*} > \delta_{sk*} > 0$ entonces la ruta mas corta p(AC, k*) va a través de v_s, así que *c(s)=k** y $v_{c(r)}$, v_{k*} es un par actualizado de los nodos mas lejanos con respecto a el AC con cota inferior asociada a la cota inferior $LB_{rs} \leftarrow \delta_{sk*} + LB_{rs}$

Descripción del Algoritmo

El algoritmo de Dvir y Handler [6] para encontrar un centro absoluto de una red recibe como entrada la distancia de rutas mas cortas *D* y el conjunto de arcos *A*. Como salida muestra el diámetro *d*(*G*) y un centro absoluto AC. El algoritmo utiliza los siguientes dos procedimientos:

Procedimiento **setupWT**(*r, s*)
    hacer *t*(*i, c*) := *d*(*i, c*) para *i = r, s* y *c = 1, 2, …, n*.

Se utiliza para guardar la tabla WT en la cual se realiza la actualización de las filas *r* y *s*.

Procedimiento **update(** *i, j, k* **)**
    restar *t*(*j*, *k*) de todas las entradas en la fila *j*
    si no quedan columnas positivas en WT, entonces
        UB := LB_{rs} y CAC se localiza a .5 UB – *d*(*j, k*) unidades de v_j en *(i, j)*
        ir a fin de ciclo

Se utiliza para realizar las actualizaciones en las filas *r, s,* en el caso de que no existan columnas positivas en la tabla WT se actualiza el UB y se encuentra el CAC.

Inicio
    Dado D, encontrar $\{k(i)\}_{i=i}^{n}$, VC
    Hacer UB := 2*l*(VC), CAC := VC

    Para todo (*r, s*) $\in$ *A* hacer
    Inicio

Si $k(r) = k(s)$ entonces ir a fin de ciclo
calcular $LB_{rs} := d_{rs} + d(r, k(s)) + d(s, k(r))$
si $LB_{rs} \geq UB$ entonces ir a fin de ciclo
set upWT(r, s**)**
update($s, r, k(s)$ **)**
update($r, s, k(r)$ **)**
cotas: (suponga que ocurre una entrada máxima de WT en la fila i, $i \in \{r, s\}$ y en la columna k, y sea j la otra fila).
buscar una entrada máxima $t(i, k)$ en WT
calcular $LB_{rs} := LB_{rs} + t(j, k)$
si $LB_{rs} \geq UB$ entonces ir a fin de ciclo
update(i, j, k **)**
ir a cotas;
Fin de ciclo:
Fin
$d(G) :=$ UB, AC := CAC;
Fin

Para la aplicación del algoritmo propuesto por Dvir y Handler para encontrar el centro absoluto se requiere la matriz de distancias más cortas D, la cual es obtenida por el método Floyd-Warshall. Sea $k(i)$ la columna en D que contiene un valor máximo para la fila i. Entonces, $k(i)$ representa un nodo mas lejano del nodo i y $l(i) = d(i, k(i))$. De entre todos los lazos se selecciona uno arbitrariamente. Sea UB la cota superior actual en $d(G)$, y CAC, el candidato actual correspondiente a UB. Una inicialización natural es hacer CAC = VC y UB = $2l$(VC), donde $d(\text{VC}, k(\text{VC})) = \min\{d(i, k(i))\}, i = 1,2,\ldots,n\}$. El algoritmo trabaja sobre un arco (r, s) dado. Los cálculos se realizan en una matriz $WT_{2 \times n}$, la cual inicialmente consiste en el par de filas r, s de D, esta es actualizada de manera iterativa. Cuando se completa el análisis del arco (r, s), se forma una nueva tabla para el siguiente arco a ser investigado.

Tal como se documentó en [6], el algoritmo se justifica por los resultados de los teoremas 1, 2 y 3, los cuales se reproducen en el apéndice 1. Por ejemplo el Teorema 1 trata el caso donde v_p, v_q coinciden con $k(r) = k(s)$. El Teorema 2 establece la cota inferior inicial LB_{rs}, mientras el Teorema 3 establece el procedimiento iterativo de actualización para LB_{rs}. En resumen, la inspección del arco (r, s) termina en una de tres maneras: cuando $k(r) = k(s)$, $LB_{rs} \geq UB$, o WT no contiene mas columnas positivas, en cuyo caso se mejora UB y CAC.

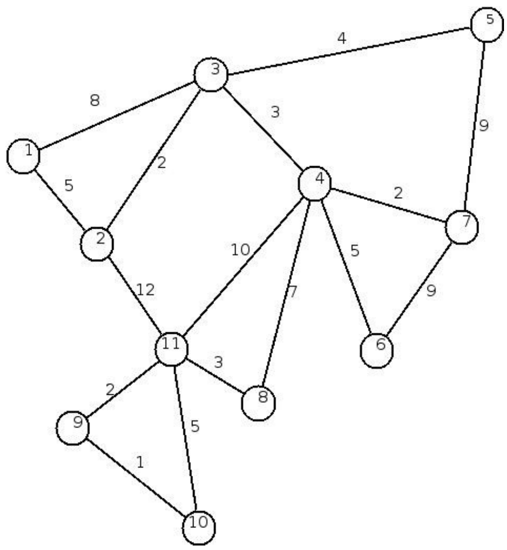

Figura 2

Ejemplo: A continuación se ilustra el uso del algoritmo encontrando el centro absoluto en una red con 11 nodos y 16 arcos, mostrados en la Figura 2. La matriz de rutas más cortas está dada en la Tabla 1.

El algoritmo inicializa y supone que VC = 4, l(VC) = d(4, 10) = 13, UB = 26, CAC = VC.

El algoritmo descarta los arcos con $k(r) = k(s)$, al eliminarlos el grafo resultante se muestra en la Figura 3.

	1	2	3	4	5	6	7	8	9	10	11	K(i)
1	0	5	7	10	11	15	12	17	19	20	17	10
2	5	0	2	5	6	10	7	12	14	15	12	10
3	7	2	0	3	4	8	5	10	15	16	13	10
4	10	5	3	0	3	5	2	7	12	13	10	10
5	11	6	4	3	0	8	1	10	15	16	13	10
6	15	10	8	5	8	0	7	12	17	18	15	10
7	12	7	5	2	1	7	0	9	14	15	12	10
8	17	12	10	7	10	12	9	0	5	6	3	1
9	19	14	15	12	15	17	14	5	0	1	2	1
10	20	15	16	13	16	18	15	6	1	0	3	1
11	17	12	13	10	13	15	12	3	2	3	0	1

Tabla 1

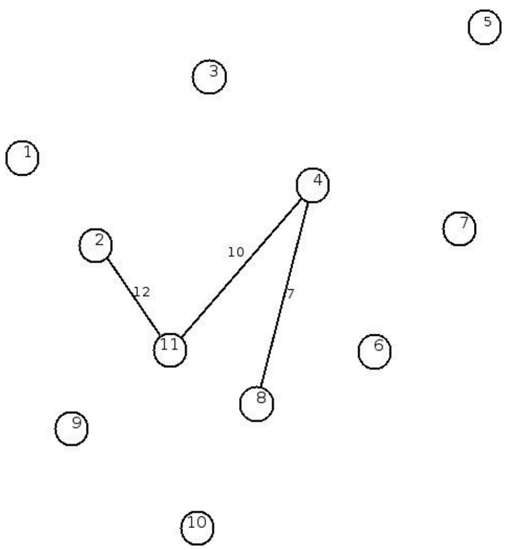

Figura 3

Al procesar el arco (2,11) se calcula $LB_{2,11}$= 12+5+3 =20, noté que es menor que UB, por lo que este arco no se descarta, así que este arco necesita ser investigado. Se guarda WT con las filas 2 y 11 como se muestra en la Tabla 2.

	1	2	3	4	5	6	7	8	9	10	11
2	5	0	2	5	6	10	7	12	14	15	12
11	17	12	13	10	13	15	12	3	2	3	0

Tabla 2

Se realiza el procedimiento **update( )** dos veces lo que resta $t(2, 1) = 5$ de la fila 2 y $t(11, 10) = 3$ de la fila 13. Las columnas que permanecen estrictamente positivas son mostradas en la Tabla 3.

	5	6	7
2	1	5	2
11	10	12	9

Tabla 3

	5	6	7
2	-4	0	-3
11	10	12	9

Tabla 4

El elemento mayor en la Tabla 3 es 12, ahora se actualiza $LB_{2,11} = 20 + t(2,6) = 25$ y nuevamente se realiza el procedimiento **update( )** en WT, restando $t(2,6) = 5$ de la fila 2. Ya que no quedan columnas positivas (como se muestra en la Tabla 4) se actualiza el $UB = LB_{2,11} = 25$ y CAC se localiza a .5 UB – $d(2, 6) = 3$ unidades de v_2 en el arco (11, 2). Se continúa investigando en los dos arcos restantes y se tiene que $LB_{11,4} = 10 + 3 + 10 = 23$ por lo que se tiene que revisar este arco ya que es menor que UB, se guarda WT con las filas 4 y 11 como se muestra en la tabla 5.

	1	2	3	4	5	6	7	8	9	10	11
4	10	5	3	0	3	5	2	7	12	13	10
11	17	12	13	10	13	15	12	3	2	3	0

Tabla 5

Se realiza el procedimiento **update( )** dos veces lo que resta $t(11, 10) = 3$ de la fila 11 y $t(4, 1) = 10$ de la fila 4. Nótese que ya no quedan columnas positivas en WT como se muestra en la Tabla 6.

	1	2	3	4	5	6	7	8	9	10	11
4	0	-5	-7	-10	-7	-5	-8	-3	2	3	0
11	14	9	10	7	10	12	9	0	-1	0	-3

Tabla 6.

Entonces $UB = LB_{11,4} = 23$ y CAC se localiza a .5 UB – $d(4, 1) = 1.5$ unidades de v_4 en el arco (11, 4). Enseguida se investiga el arco restante y se tiene que $LB_{4,8} = 7 + 10 + 6 = 23$, nótese que $LB \geq UB = 23$ por lo que ya no se tiene que revisar este arco. Dado lo anterior se tiene que $d(G)$=UB=23 y AC=CAC.

Complejidad computacional del algoritmo: El orden de complejidad del algoritmo para encontrar el centro absoluto de una red es $O(mn^2)$. Sin embargo debido a que el algoritmo recibe como parámetro de entrada una matriz de rutas mas cortas, esta debe ser calculada, para realizar esta operación se utilizó el algoritmo Floyd-Warshall, el cual tiene una complejidad computacional de $O(n^3)$, lo que produce un algoritmo de orden $O(n^3 + mn^2)$.

Experimentación Computacional

Las características del equipo en donde se realizaron las ejecuciones del algoritmo son las siguientes: Laptop Dell Inspiron con procesador Intel Centrino Duo de 1.66 GHz, 1 GB RAM, disco duro de 5400 rpm, utilizando como sistema operativo Windows XP. La implementación se realizó en lenguaje C con compilador GNU de Dev C++.

Para realizar las pruebas del algoritmo se utilizaron instancias en las cuales los nodos representan coordenadas de ubicación de instalaciones y los costos de los arcos representan la distancia entre los nodos. Además se tiene la característica de que un nodo solo se puede comunicar con sus vecinos, es decir, con los nodos que se encuentran cercanos a él. Esta cercanía se definió como una zona en donde se encuentran los nodos vecinos. Los resultados reportados al correr las pruebas se encuentran en la Tabla 7.

Nodos	Redes	Arcos	Arcos Eliminados Coincidencia	Halpern's Bound	Tiempo ejecución (seg.) Floyd-Warshall	ACNet
250	10	436-460	217-229	0	0-1*	0*
1000	10	1774-1814	886-906	0	13-14	0-1*
5000	1	9024	4465	46	1669	16

* 0 seg. se refiere a que el tiempo de ejecución es menor a un segundo.

Tabla 7

Nótese en la tabla 7, que las redes generadas con 250 nodos, tuvieron alrededor de 200 arcos de coincidencia, es decir se están eliminando casi la mitad de los arcos que se generaron, lo cual ayuda al algoritmo en el tiempo de ejecución para encontrar el centro absoluto. Obsérvese que para las redes generadas con 1000 nodos y 5000 nodos pasa algo similar, los arcos que coinciden son alrededor de la mitad de los arcos de cada una de las instancias en ambos casos. De esta manera el algoritmo encuentra los centros absolutos en tiempo de ejecución muy buenos para instancias relativamente grandes.

Cabe mencionar que estas pruebas se realizaron en instancias de mayor tamaño (mayor cantidad de nodos y arcos) que las realizadas por el autor del algoritmo obteniendo resultados muy buenos con respecto a tiempo de cómputo.

Conclusiones

En este trabajo hemos llevado a cabo una implementación computacional de un algoritmo propuesto en [6] para encontrar el centro absoluto de una red, además de realizar una implementación del algoritmo Floyd-Warshall. Se realizaron pruebas con una cantidad de nodos mayor que las probadas por Dvir y Handler, obteniendo muy buenos resultados respecto a tiempo de cómputo, con lo cual se cumple con el objetivo de mostrar que se tienen buenos tiempos de ejecución para instancias de mayor tamaño que las utilizadas por los autores del algoritmo.

Agradecimientos: Los primeros tres autores agradecen al CONACYT por su apoyo económico como becarios del Programa de Maestría en Ciencias en Ing. de Sistemas de la FIME.

Referencias

1. H. A. Eiselt y C.-L. Sandblom. *Decision Analysis, Location Models, and Scheduling Problems*. Springer, Berlín, Alemania, 2004.
2. P. B. Mirchandani y R. L. Francis, editores. *Discrete Location Theory*. Wiley, New York, EUA, 1990.
3. S.L. Hakimi. Optimum locations of switching centers and the absolute centers and medians of a graph. *Operations Research*, 12:450-459, 1964
4. C. Toregas, R. Swan, C. ReVelle y L. Bergman. The location of emergency service facilities. *Operations Research*, 19:1366-1373, 1971.
5. R. Z. Ríos-Mercado y E. Fernández. A reactive GRASP for a sales territory design problem with multiple balancing requirements. Reporte técnico PISIS-2006-12, Programa de Posgrado en Ingeniería de Sistemas, FIME, UANL, San Nicolás de los Garza, México, Septiembre 2006.
6. D. Dvir y G.Y. Handler. The absolute center of a network. *Networks*, 43(2):109-118, 2004.

7. R. Ahuja, T. Magnanti y J. Orlin. *Network Flows*. Prentice-Hall, Englewood Cliffs, EUA, 1992.

Fichas biográficas

Dexmont Alejandro Peña Carrillo

Es egresado de la carrera de Lic. en Ciencias Computacionales de la FCFM de la UANL. Actualmente realiza su trabajo de tesis de maestría en la División de Posgrado en Ingeniería en Sistemas de la FIME sobre un nuevo modelo de pronósticos de series de tiempo basado en aprendizaje estadistico y procesos estocásticos. Anteriormente realizó su tesis de licenciatura sobre algoritmos de filtrado lineal para sistemas con presencia de ruido blanco con distribucion de Poisson.

Anel Berenice Reyes Ramírez

Lic. Anel Berenice Reyes Ramirez es actualmente estudiante de tiempo completo en el Programa de Maestría en Ciencias en Ing. de Sistemas de la División de Posgrado en Ingeniería de Sistemas de la FIME, UANL. Recibió su título de Licenciada en Matemáticas de la Universidad Autónoma de Nuevo León, por medio de la tesis "Diseño de una red minimizando el máximo valor del flujo y el número de conexiones utilizadas". Su área de interés es la investigación de operaciones aplicada a la optimización de los problemas multicriterio y recientemente su principal interés se basa en problemas de distribución de productos en cadenas de suministro orientados a objetivos múltiples.

Ruth Marlen Ávila Guerrero

Es egresado de la carrera de Ing. Mecánica Administrador de la FIME de la UANL. Actualmente es estudiante de maestría en la División de Posgrado en Ingeniería en Sistemas de la FIME.

Roger Z. Ríos Mercado

El Dr. Roger Z. Ríos Mercado labora actualmente como Profesor Titular en la División de Posgrado en Ingeniería de Sistemas de la FIME de la UANL. Recibió sus títulos de Doctor y Maestro en Ciencias en Investigación de Operaciones e Ingeniería Industrial de la Universidad de Texas en Austin, y su título de Lic. en Matemáticas de la UANL. Sus áreas de interés se enfocan en el campo de la investigación de operaciones como soporte científico a los problemas de toma de decisiones, en particular, a la investigación y desarrollo de algoritmos eficientes para la solución de problemas relacionados con el diseño óptimo de territorios de ventas en el ramo logístico, la secuenciación de operaciones en procesos de manufactura y la operación eficiente de redes de transporte de gas natural. Es miembro del Sistema Nacional de Investigadores (Nivel II), de la Academia Mexicana de Ciencias y del Cuerpo Académico consolidado de Ing. de Sistemas. Más sobre su obra científica, distinciones, estudiantes y oportunidades para colaborar con él, puede encontrarse en:
< *http://yalma.fime.uanl.mx/~roger/* >.