

SOLUCIÓN AL PROBLEMA DE ENTREGA DE PEDIDOS UTILIZANDO RECOCIDO SIMULADO

RESUMEN

El objetivo de este trabajo es determinar la secuencia más favorable de entrega de pedidos y la ruta que permita a una empresa de comida rápida, cumplir con los tiempos de entrega comprometidos con los clientes. Se representó la ciudad como un dígrafo, se usó el algoritmo de Dijkstra para determinar las rutas y se empleó el algoritmo Recocido Simulado, para determinar el orden de entrega más favorable de un lote de pedidos evaluando el mínimo de pedidos tardíos (N^T). Los resultados son comparados con los arrojados por la metodología usada por el domicilio.

PALABRAS CLAVES: Entrega de pedidos, usos de técnicas heurísticas.

ABSTRACT

The purpose of this paper is to determine the most favorable sequense for delivering orders and the rute that allows the company to carry out delivery times previously agreed with the customers. The city is presented as a digrafo. The Dijkstra Algorithm is used to determine the shortest distances. The Simulated Annealing Algorithm is used to determine the most favorable delivery order within an order lot. As a performance indicator the least number of late orders is used (N^T). The results obtained are compared with the results obtained from the method used by the delivery service.

KEYWORDS: *Delivery order, Heuristic*

JORGE H. RESTREPO C.

Profesor auxiliar
Facultad Ingeniería Industrial
Universidad Tecnológica de
Pereira
jhrestrepoco@utp.edu.co

JOHN JAIRO SÁNCHEZ C.

Profesor Asistente
Escuela de Tecnología Industrial
Universidad Tecnológica de
Pereira
jasaca@utp.edu.co

MARIO HOYOS MESA

Profesor Titular
Facultad Ingeniería Mecánica
Universidad Tecnológica de
Pereira
marhoyos@utp.edu.co

1. INTRODUCCIÓN

Para una empresa de servicios de comida rápida a domicilio, es de vital importancia cumplir con los tiempos de entrega acordados con los clientes. En nuestro medio la entrega de pedidos se hace por medio de un domicilio que se transporta en una motocicleta, quien en virtud de su experiencia aplica una secuencia de entrega que considera la más adecuada; pero que en la práctica no a podido evitar un número significativo de entrega tardías que han causado pérdida de clientes y disminución en las ventas¹. Es evidente la necesidad de los empresarios de mejorar la entrega de pedidos.

La solución del problema tratado en este trabajo consiste en determinar una secuencia de pedidos que garantice un número lo menor posible de pedidos tardíos. Como el tiempo invertido en la entrega de pedidos depende de las distancias recorridas por el motociclistas es necesario considerar como parte del problema la determinación de rutas que favorezcan la entrega oportuna de los pedidos. Las mejores rutas se pueden determinar partir de un grafo que represente los puntos de entrega, y los diferentes caminos entre ellos considerando los sentidos de las vías y las longitudes correspondientes. Existen

varios algoritmos que permiten la determinación de los caminos más cortos en entre dos puntos cualquiera como el Floyd, Ford y Dijkstra entre otros. Se consideró más conveniente utilizar el algoritmo de Dijkstra para el cálculo de una matriz de distancias más cortas para cada par de puntos de entrega.

Para determinar la secuencia de entrega más favorable, la que produce el menor número de pedidos tardíos, se aplico un algoritmo de Recocido Simulado y se utilizó, para cada secuencia obtenida la matriz de distancias más cortas con el fin de calcular el tiempo de entrega de cada pedido.

El método propuesto presenta una ligera ventaja sobre el utilizado por el domicilio y para que sea más efectivo es necesario introducir reformas importantes en las técnicas utilizadas por la empresa en el manejo de los domicilios.

2. PROBLEMA DE INVESTIGACIÓN

2.1 Caracterización del problema

En una empresa que presta el servicio de comidas rápidas a domicilio a toda la ciudad, existe una persona encargada de recibir las solicitudes de los diferentes pedidos, los cuales van sucediendo cada uno en un tiempo tp_i . El primer pedido se hizo en el tiempo $tp_1=0$, y de ahí en adelante se van generando el resto de pedidos,

¹ Según comentario de la administración

y así sucesivamente se van registrando todos los tiempos en los cuales se efectúan los diferentes pedidos. En la medida que se recibe cada pedido, la empresa compromete un tiempo con el cliente (tc_i), lo que indica que el cliente espera su pedido en el momento (tpc_i), el cual está constituido por el tiempo en que se efectúa el pedido (tp_i) más el tiempo comprometido (tc_i). El tiempo en que el cliente espera el pedido, se compara con el tiempo que demora el domicilio desde que arranca con el pedido, hasta que llega al nodo y sale de allí (te_i). La empresa tiene como política formar un lote de pedidos, el primer pedido arranca en $tp_i=0$, y se esperan solicitudes hasta el minuto veinte (20). A partir de este momento el domicilio inicia el proceso de distribución de los pedidos, a este momento se le denomina (m_i). El punto de venta, lugar donde se inicia el recorrido para la entrega de los pedidos es llamado (Origen). La diferencia de la comparación entre el tiempo que el cliente espera el pedido y el tiempo que demora el domicilio, da como resultado si el pedido llegó a tiempo o tarde. Es aquí donde tiene importancia la medida de desempeño utilizada N^T (total de pedidos tardíos).

Para comprender mejor el problema, es necesario tener en cuenta las siguiente variables:

1. tp_i = Momento t en que se efectúa el pedido
2. p_i = Vector que almacena el nodo del pedido
3. tc_i = Tiempo comprometido para la entrega del pedido a partir del momento t .
4. te_i = Tiempo de entrega, que corresponde al tiempo t gastado por el domicilio desde el momento en que llega al nodo i hasta que sale de éste.
5. $tpc_i = tp_i + tc_i$, es el momento t que el cliente espera su pedido.
6. S_e = Secuencia de entrada, que corresponde al orden en que se fueron generando los pedidos.
7. n = Número de pedidos
8. S_s = Secuencia de entrega, que corresponde al orden final de entrega de pedidos, la cual depende de N^T .
9. N^T = Total de pedidos tardíos.
10. p^T = Identificación de pedidos tardíos
11. $tpcr_i$ = Tiempo t en que llega el pedido al nodo i .
12. d_{ij} = Distancia entre nodos, (donde i es el nodo origen y j es el nodo destino). $i \neq j$.
13. v = Velocidad de circulación en la ciudad. ($V = 15$ km./hora)
14. m_i = Momento t de inicio de entrega de pedidos. ($m_i = 20$ minutos).

3. CONSTRUCCIÓN FUNCIÓN OBJETIVO

$tpcr_i$ = Tiempo t en que llega el pedido i al nodo p_i
Siempre para

$$tpcr_1 = m_t + [d(\text{Origen}, p_1) / v]$$

$tpcr_i$ = Corresponde al tiempo que ha transcurrido desde que se hizo el primer pedido, hasta que el domicilio sale

del punto de venta, más el tiempo gastado desde el punto de venta (origen), hasta el nodo que identifica la primera entrega.

Y para todos los i diferente de 1, (donde $i=2,3,\dots,n$), se procede como sigue:

$$tpcr_i = m_t + [d(\text{origen}, p_1) / v] + \sum_{k=2}^i [d(p_{k-1}, p_k) / v + te_{k-1}]$$

donde $i=2,3,4,\dots$

Para la segunda entrega, el $tpcr_2$ corresponde a la suma de $tpcr_1$, más el tiempo gastado desde el nodo donde se hizo la primera entrega, hasta el nodo siguiente, más el tiempo que transcurre en la entrega del pedido anterior. Para todas las i siguientes, se aplica la misma metodología, la cual se resume en la ecuación que aparece a continuación

$$tpcr_i = tpcr_1 + \sum_{k=2}^i [d(p_{k-1}, p_k) / v + te_{k-1}]$$

2.2 Función Objetivo

La función objetivo a minimizar, es el número total de pedidos tardíos, la cual se representa matemáticamente como sigue:

$$\text{Min: } N^T = \sum_{i=1}^n U(tpcr_i - tpc_i)$$

$tpcr_i$ = Tiempo t en que llega el pedido i al nodo p_i

tpc_i = Es el momento t en que el cliente espera su pedido.

Donde:

$$U(tpcr_i - tpc_i) = 1.$$

Si $tpcr_i > tpc_i$ (pedido tardío)

$$U(tpcr_i - tpc_i) = 0.$$

Si $tpcr_i \leq tpc_i$ (pedido cumplido)

4. CONTEXTUALIZACIÓN

4.1 Teoría de grafos

Una ciudad, presenta una red de tránsito de vehículos que si se describe mediante un esquema geométrico que no tiene en cuenta el sentido de las vías, esto representaría la estructura de un grafo. Un grafo es una "terna $G = (V, A, i)$, donde V y A son conjuntos finitos e i una aplicación que, a cada elemento de A , asocia un par de elementos de V . Los elementos de V se llaman vértices de G , los elementos de A serán las aristas de G , e i será la aplicación de incidencia, que asocia a cada arista sus extremos o vértices"². En esta red de tránsito las intersecciones de calles y las carreras o esquinas representan los vértices (nodos), y las calles y carreras que los unen se denominan aristas (distancias entre nodos). Según la clasificación anterior, dos vértices de

² TORANZOS, Fausto. Introducción a la Teoría de Grafos. OEA. 1976. Pág. 8.

un grafo son adyacentes si son extremos de una misma arista.

Al apreciar la malla vial de una ciudad hay que considerar el sentido de las rutas, y con esta connotación, se genera una nueva estructura que se denomina grafo dirigido o dígrafo.

4.2 Algoritmo de Dijkstra

El algoritmo de Dijkstra sirve para encontrar las distancias más cortas de un nodo origen a cualquier otro nodo y las rutas. En este método³ el primer paso es asegurarse que existe una distancia asociada con cada par de nodos del dígrafo. Esta distancia es la del arco incidente, si éste es un arco entre dos nodos, esta distancia será cero para el recorrido de un nodo a el mismo e infinita para un par de nodos que no tiene un arco que los una. (En la práctica el valor del infinito se sustituye por un número muy grande por efectos de la programación en computadora). No necesariamente se tiene que dar que la distancia de i a j sea igual de j a i. Si definimos d (i, j) como la distancia asociada con el arco (i, j). Esto es que d (i,j) no necesita ser igual a d(j,i). Este algoritmo asigna una etiqueta a cada nodo o vértice del dígrafo. Esta etiqueta es la distancia que hay desde el nodo de arranque(s), a lo largo de la ruta más corta encontrada. La etiqueta puede estar en uno de dos estados: La etiqueta puede ser permanente en el caso de que la distancia encontrada a lo largo de toda la ruta sea la más corta, o puede ser temporal, esto corresponde a la duda de que la ruta encontrada sea la mas corta de todas. El método gradualmente cambia las etiquetas temporales por unas permanentes.

Dado un conjunto de nodos con etiquetas temporales, el objetivo es hacer que ese conjunto sea mínimo encontrando rutas para estos nodos usando las rutas más cortas de los nodos etiquetados como permanentes, seguido por un arco de un nodo con etiqueta permanente. Una vez esto se ha logrado, el nodo con menor etiqueta temporal es seleccionado y se le coloca etiqueta permanente. Este proceso se repite hasta que se le asigne una etiqueta permanente al nodo(t) que representa el final de la ruta que se está analizando, lo que puede ocurrir eventualmente, una vez el algoritmo es usado, entonces habrá una etiqueta temporal menos, y así el número de nodos con etiquetas temporales decrece a cero.

4.3 Programación secuencial

En el problema de entrega de pedidos, implícitamente se establece un orden, el cual posiblemente no sea el más adecuado y cuando este tiene involucrado algún factor que puede causar detrimento económico, entonces se hace necesario buscar la secuencia (orden), que permita minimizar este factor. Por esto se hace necesario recurrir a técnicas que ayuden a resolver el problema de secuenciación. Hay técnicas heurísticas disponibles que

ayudan a resolver este tipo de problemas. Entre ellas se tienen las siguientes: El Simulated Annealing(SA), Algoritmo Genético(AG), Tabú Search, entre otras. Para resolver este problema se utilizará el Simulated annealing(Recocido Simulado).

4.4 Recocido simulado (rs)

A principios de los ochenta, Kirkpatrick,⁴ cuando estudiaba el diseño de circuitos eléctricos, y Cerny,⁵ cuando investigaba el problema del agente viajero, encontraron de manera independiente que los problemas que estaban tratando, dada su naturaleza combinatoria podían ser abordados a través de la aplicación del algoritmo de Metrópolis. Para ello, pensaron que era posible establecer una analogía entre los parámetros que intervienen en la simulación termodinámica, y los que aparecen en los métodos de optimización local, la cual se pueden observar en la tabla 1.

TERMODINÁMICA	OPTIMIZACIÓN
Configuración	Solución factible
Configuración fundamental	Solución óptima
Energía de la configuración	Costo de la solución
Temperatura	¿?

Tabla 1. Analogía entre la termodinámica y optimización

Como se ve, al concepto físico de temperatura no le correspondería un significado real en el campo de la optimización, sino que ha de ser considerado como un parámetro, T, que habrá que ir ajustando. De este manera se podría imaginar similares los procesos que ocurren cuando las moléculas de una sustancia van colocándose en los diferentes niveles energéticos buscando el equilibrio a una determinada temperatura, y los que ocurren en los procesos de minimización en optimización local(para la maximización sería semejante). En el primer paso, fijando la temperatura, la distribución de las partículas en los diferentes niveles sigue la distribución de Boltzmann, por lo que cuando una molécula se mueve, ese movimiento será aceptado en la simulación si la energía disminuye, o bien con una probabilidad

proporcional al factor de Boltzmann ($e^{\left(\frac{-\text{delta}}{T}\right)}$) en caso contrario. Al hablar de optimización, fijado el parámetro T, producimos una perturbación, aceptando directamente la nueva solución cuando su costo disminuye, o bien con una probabilidad proporcional al factor de Boltzmann en caso contrario.

Esto último es la clave del Recocido Simulado, ya que básicamente es una estrategia heurística de búsqueda local, en la cual la elección del nuevo elemento del

⁴ DÍAZ, Adenso, GLOVER, Fred, GHARZIRI, Hassan, GONZALEZ, J, LAGUNA, Manuel, MOSCATO, Pablo. Optimización Heurística y Redes Neuronales. Editorial Parafino. 1996. P.41

⁵ Idem. P. 41

³ SMITH, David. Network Optimization Practice. John Wiley & Sons. 1982. Pág. 55

entorno $N(s)$ se realizan aleatoriamente y, como se vio con anterioridad este tipo de estrategia presenta el inconveniente que si durante el proceso de búsqueda se cae en un óptimo local, la técnica no sería capaz de salir de él. Para evitarlo, el recocido simulado permite con una cierta probabilidad (cada vez menor conforme nos acercamos a la solución óptima) el paso a soluciones peores. En efecto analizando el comportamiento del factor de Boltzmann en función de la temperatura, se observa que, como disminuye ésta, disminuye rápidamente la probabilidad de que se acepte una solución peor que la actual.

Por lo tanto, la estrategia que se sigue en el Recocido Simulado será partir de una temperatura alta (con lo cual se permite cambios a soluciones peores en los primeros pasos, cuando aún se está lejos del óptimo global), y posteriormente reduciendo la temperatura, disminuyendo la posibilidad de cambios a soluciones peores cuando ya se encuentra más cercano al óptimo buscado. De este procedimiento proviene el nombre del Algoritmo Recocido Simulado, ya que el recocido es un proceso metalúrgico (usado por ejemplo para eliminar en el acero laminado en frío las tensiones internas adquiridas) mediante el cual se somete al material a un calentamiento rápido, para posteriormente dejarlo enfriar lento y controladamente durante horas.

El algoritmo de recocido simulado podría representarse de la siguiente forma:

```

Input ( $T_0, \alpha, L, T_f$ )
 $T \leftarrow T_0$ 
 $S_{actual}$  = genera solución inicial
While  $T \geq T_f$  Do
  Begin
    For cont  $\leftarrow 1$  to  $L(T)$  Do
      Begin
         $S_{candidata}$   $\leftarrow$  selecciona solución  $N(S_{actual})$ 
         $\delta = Costo(S_{candidata}) - Costo(S_{actual})$ 
        If  $((U(0,1) < e^{\left(\frac{-\delta}{T}\right)})$  or
          ( $\delta < 0$ ) then  $S_{actual} \leftarrow S_{candidata}$ 
      End
    End
     $T = \alpha(T)$ 
  End

```

(Escribe como solución, la mejor de las S_{actual} Visitadas)

Es decir, se selecciona la temperatura inicial T_0 , la velocidad de enfriamiento (o sea, la forma de cálculo α del valor T_{i+1} a partir del T_i cuando se disminuye la temperatura tras haber estado $L(T)$ iteraciones en esa T), y la temperatura final T_f . Se genera una solución inicial perteneciente a los espacios de solución Ω y, mientras no se llegue al final del proceso, para cada T se calcula el número $L(T)$ de veces (antes de disminuir la temperatura) una solución que esté en el entorno $N(S_{actual})$ de la actual, la cual sustituirá a ésta bien si tiene menor costo, o bien

con una probabilidad $e^{\left(\frac{-\delta}{T}\right)}$. Para el cálculo de esta probabilidad se genera un número aleatorio uniforme

(0,1). Finalmente la solución ofrecida será la mejor de todas las S_{actual} visitadas.

Para este problema se hicieron muchas pruebas con el parámetro temperatura inicial (T_0), pero los resultados encontrados fueron pobres. Por último se ensayó con una temperatura inicial de 194 por 3, donde 194 es el número de nodos que representan la malla vial y los resultados mejoraron en forma importante, conservando un tiempo de procesamiento prudente (segundos). La tasa de enfriamiento con que se trabaja el RS, es de 0.99 y el control de parada es cuando la temperatura toma valores iguales o menores a 1. Se utilizó como mecanismo de perturbación para generar nuevas soluciones el intercambio aleatorio de elementos del vector que identifica la secuencia de entrega.

5. DESCRIPCIÓN DE LA INVESTIGACIÓN

Para cumplir el propósito de esta investigación, fue necesario representar la malla vial del Municipio de Santa Rosa de Cabal como un dígrafo geométrico, donde a partir de éste se construye una matriz de pesos de arcos, la que utiliza el algoritmo de Dijkstra para determinar las distancias más cortas desde un nodo origen a cualquier otro nodo y la ruta entre nodos. Al ejecutar el algoritmo de Dijkstra para todos los nodos de la malla vial como nodo origen, se construye una matriz de distancias mínimas, la cual es utilizada por el algoritmo Recocido Simulado para evaluar cada secuencia de lotes de pedidos con la medida de desempeño (número de pedidos tardíos- N^T).

6. VALIDACIÓN

6.1 Validación de la heurística, con relación al método utilizado por el domicilio

Para realizar la validación, se tomó la información de la empresa SUPERPOLLO de Santa Rosa de Cabal (Rda) y se tuvo en cuenta lo siguiente:

- Velocidad promedio de circulación en la ciudad de Santa Rosa de Cabal = 15 km/h
- Tiempo a partir del cual se inicia la entrega de pedidos = 20 minutos
- El tiempo t_i , corresponde al tiempo en el cual fue ordenado el pedido.
- Tiempo comprometido, es el tiempo de entrega que compromete el administrador al cliente.
- Tiempo de entrega definitivo, corresponde al tiempo t_i más el tiempo comprometido
- Tiempo por nodo, es el tiempo de entrega (te_i), que corresponde al tiempo t gastado por el domicilio desde el momento en que llega al nodo i hasta que sale de éste. Para determinar este tiempo, se tomaron 38 datos de campo, los cuales después de

procesados se comportaron como una distribución normal. A partir de la media y la desviación estándar que arrojó la información de campo, y con la ayuda de la función estadística de Excel (distribución normal), se hallaron todos los valores utilizados de t_{α} en la validación.

Las secuencias que se validaron, están compuestas entre 11 y 20 pedidos.

El cuadro 2, corresponde al resumen de treinta secuencias de lotes de pedidos, donde se hace una comparación del domicilio frente al Recocido Simulado. La columna secuencia identifica el número de la secuencia evaluada. En las columnas del domicilio y Recocido Simulado aparece el número de pedidos entregados a tiempo.

SECUENCIA	DOMICILIO	RECOCIDO SIMULADO
1	1	3
2	1	3
3	1	3
4	1	3
5	1	3
6	3	4
7	4	4
8	4	4
9	4	6
10	4	4
11	3	4
12	2	3
13	2	3
14	1	4
15	2	4
16	3	3
17	1	2
18	3	4
19	3	3
20	2	3
21	2	4
22	1	4
23	2	4
24	2	4
25	3	4
26	3	4
27	2	3
28	2	3
29	2	4
30	2	4
TOTAL ÉXITOS	4/30	26/30

Tabla 1. Resumen validación

De las treinta secuencias realizadas, las diez primeras corresponden a lotes reales y las veinte siguientes son simulaciones. Se comparó el método del Domicilio con el Recocido Simulado(RS). El RS, obtuvo 26 éxitos de

las treinta secuencias, mientras el Domicilio, obtuvo 4 éxitos de las 30. Es necesario aclarar que el domicilio nunca a la heurística propuesta, los éxitos que aparecen del domicilio son dados por empates que se presentaron.

7. CONCLUSIONES

A pesar que se logró construir un procedimiento para determinar las rutas óptimas, éste desde el punto de vista práctico no tiene utilidad en el momento, debido que para el motociclista(domicilio)es complejo realizar dos funciones a la vez, conducir e interpretar la ruta.

El programas de cómputo, Recocido Simulado, debe realizar su procesamiento en un tiempo no mayor a tres minutos, debido a que un tiempo mayor aumentaría el retraso en la entrega de los pedidos. El Recocido Simulado(RS) realiza el procesamiento en menos de tres minutos.

Un aspecto importante, fue la construcción de la matriz de distancias mínimas entre todos los nodos que representan la ciudad, por medio del algoritmo de Dijkstra, la cual sirvió para resolver el problema de pedidos de SUPERPOLLO, pero además queda como una fuente de datos para resolver otros problemas de la ciudad que requieran distancias mínimas como: La entrega del periódico, la entrega de encomiendas, la ruta más corta que debe seguir el vehículo del cuerpo de bomberos para atender una emergencia.

La medida de desempeño, mínimo de pedidos tardíos, utilizada para seleccionar la secuencia de entrega de un lote de pedidos es rígida, debido a que ella evalúa de forma puntual si el pedido se entregó en el tiempo comprometido o no. Lo que hace que se rechacen tareas con un pequeño retardo, el cual puede ser tolerado por el cliente, dando lugar a que el número de pedidos tardíos no se pueda disminuir más.

Al aplicar la heurística del Recocido simulado en treinta secuencias distintas, ella superó en el 86% a las secuencias propuestas por el domicilio.

A pesar que la heurística superó al método utilizado por el domicilio, es claro que la ventaja no es tan importante, mostrando siempre superioridad de los pedidos tardíos sobre los cumplidos.

Se realizó un incremento de treinta minutos a la variable que representa el momento t en que el cliente espera su pedido y se observó que el número de pedidos tardíos sigue teniendo prácticamente el mismo comportamiento presentado en la validación original, lo que hace pensar que a pesar que el algoritmo mejora el orden de entrega propuesto por el domicilio, parece que existen otras causas que impiden disminuir el número de pedidos tardíos, como el caso del tamaño del lote, la asignación

del tiempo comprometido para la entrega (hacer sensibilidad sobre esta variable como un trabajo futuro), la dispersión geográfica de los pedidos, entre otras.

Debido a que la resolución del problema implica realizar muchos cálculos, se hace necesario diseñar y construir unos programas de cómputo que faciliten el procesamiento de la información, para entregar una respuesta oportuna a la administración.

8. BIBLIOGRAFÍA

[1] AYDÍN, M Emín, FOGARTY, Terence. A Modular Simulated Annealing Algorithm For Multiagent Systems: A Job-Shop Scheduling Application. South Bank University. [En línea]

[2] Congreso de Producción. Universidad Autónoma de Occidente. Cali Colombia. Agosto 2002.

[3] DÍAZ, Adenso, GLOVER, Fred, GHARZIRI, Hassan, GONZALEZ, J, LAGUNA, Manuel, MOSCATO, Pablo. Optimización Heurística y Redes Neuronales. Editorial Paraninfo. 1996.

[4] GLOVER, Fred, LAGUNA, Manuel. Tabú Search. Universidad de Colorado. Julio de 1997

[5] GLOVER, Fred. Optimización Heurística y Redes Neuronales. ED. Paraninfo. 1996. Pág. 37

[6] HOYOS, Mario. Notas sobre Programación Secuencial. Maestría Investigación Operativa y Estadística. UTP. Junio 30/02.

[7] IEEE Transactions on Systems, Man, and Cybernetics. Part B: Cybernetics. Vol. 32. N°5P. 686-691.

[8] IEEE Transactions on Systems, Man, and Cybernetics. Vol. 25, n° 7. Julio de 1995. 1102-1109

[9] IEEE. Transactions on Power Systems. Vol. 6 n° 2. Maintenance Scheduling by Using Simulated Annealing Method. P. 850-857

[10] INTERNACIONAL JOURNAL OF SYSTEMS AUTOMATION: Research and Applications. Vol. 2 n°3. 1992. P. 227- 243.

[11] PEHA, Jon M. Heterogeneous-Criteria Scheduling: Minimizing weighted number of tardy Jobs and weighted completion time. [En línea]. 7 nov, 2003

[12] RESTREPO, Jorge Hernán, SÁNCHEZ, John Jairo. Solución al problema de entrega de pedidos con técnicas heurísticas. Tesis Maestría. Universidad Tecnológica de Pereira 2003.

[13] SMITH, David. Network Optimization Practice. John Wiley & Sons. 1982. Pág. 55

[14] TORANZOS, Fausto. Introducción a la Teoría de Grafos. OEA. 1976.