

“DISEÑO E IMPLEMENTACIÓN DE UN ALGORITMO DE TIPO RECOCIDO SIMULADO PARA LA RESOLUCIÓN DEL PROBLEMA DEL AGENTE VIAJERO”

David araujo Carrión¹, Johnni Bustamante

¹Ingeniero en Estadística Informática 2006 email: earaujo@espol.edu.ec, daev75@hotmail.com

²Director de Tópico. Ingeniero Comercial, Escuela Superior Politécnica del Litoral, 1981, Maestría en Investigación de Operaciones, México, Instituto Tecnológico y de Estudios Superiores de Monterrey 1988, Maestría en Administración de Empresas, Costa Rica, INCAE 1992, Profesor de la ESPOL desde 1981., email: jfernand@espol.edu.ec

RESUMEN

En el presente trabajo resolveremos el problema del Agente Viajero haciendo uso del Algoritmo del Recocido Simulado, para determinar las rutas válidas entre dos o más nodos. El algoritmo fue aplicado en el sector céntrico de Guayaquil considerado como el sector 44 en el plano General de Guayaquil.

El algoritmo de Recocido Simulado fue diseñado para el área de termodinámica, Dada la dificultad práctica para resolver de forma exacta (simplex, “ramificación y acotación”, teoría de grafos, etc) toda una serie de importantes problemas combinatorios para los cuales, por otra parte, es necesario ofrecer alguna solución dado su interés práctico, comenzaron aparecer algoritmos que proporcionan soluciones factibles (es decir, que satisfacen las restricciones del problema), las cuales aunque no optimicen la solución objetivo, se supone que al menos se acercan al valor óptimo en un tiempo de cálculos razonables. Podríamos llamarlas en lugar de óptimas, “satisfactorias”, pues al menos es de suponer que son lo suficientemente buenas como para servirnos. Este tipo de algoritmo se denominan heurísticas.

In the present work we will solve the problem of the Travelling Agent making use of the Algorithm of the Simulated Annealing, in order to determine the valid routes between two or more nodes. The algorithm was applied in the centric sector of Guayaquil considered like sector 44 in the General plane of Guayaquil.

The algorithm of Simulated Annealing was designed for the thermodynamics area, Given the difficulty practical to solve of exact form (simplex, “ramification and landmark”, theory of graphs, etc) all a series of important combinatorial problems for which, on the other hand, its practical interest is necessary to offer some given solution, they began to appear algorithms that provide feasible solutions (that is to say, that they satisfy the restrictions with the problem), which although do not optimize the objective solution, one assumes that at least they approach the optimal value in a time of reasonable calculations. We could call them instead of optimal, “satisfactory”, then it is at least of supposition that is the sufficiently good thing like serving to us. This type of algorithm is denominated heuristic.

INTRODUCCIÓN

El propósito de esta investigación es presentar un algoritmo general para la solución de problemas de optimización combinatoria usando el método de recocido simulado. Este método tiene la propiedad de que, con una propia calibración en un tiempo polinomial produce buenas soluciones.

La técnica de recocido simulado es una técnica relativamente nueva para resolver problemas de optimización combinatoria. Fue propuesta por Kirkpatrick, Gellat y Vecchi [KIRS3] y originalmente se basó en una analogía entre la simulación de recocido de sólidos y el reto de resolver problemas de optimización combinatoria de gran escala.

La característica sobresaliente de la técnica es su aplicación general y la habilidad para obtener soluciones “arbitrariamente” cercanas a la óptima. Sin embargo, el obtener, soluciones de alta calidad puede requerir de mucho esfuerzo computacional. Una comparación importante que puede hacerse con respecto a si una técnica es general o no lo es, es la que se presenta con respecto a la programación lineal y la programación dinámica. La programación lineal es un modelo muy específico para solucionar problemas en donde la función objetivo y las restricciones deben ser lineales; si el problema no cae dentro de este esquema restringido, la programación lineal no puede aplicarse.

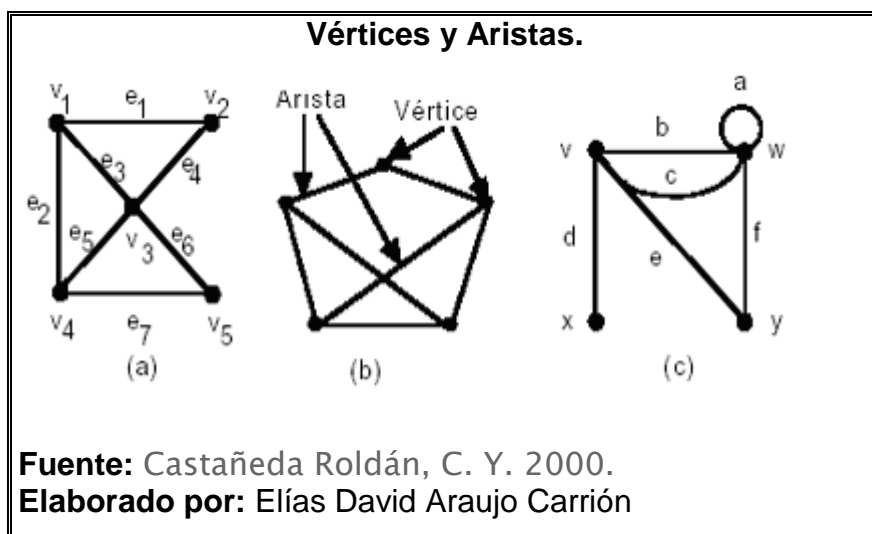
CONTENIDO

TEORÍA DE GRAFOS

En un sentido amplio, un grafo G es un diagrama que, si se interpreta en forma adecuada proporciona información. Cuando los grafos no tienen dirección alguna, se dice que son grafos no dirigidos, por lo que en esta sección se discutirá sobre ellos mencionándolos simplemente como grafos.

Lo más importante de un grafo son sus aristas y vértices.

Se denotan: Grafo (G), los vértices (v_1, v_2, \dots, v_n), y las aristas por (e_1, e_2, \dots, e_m).



MATRIZ DE INCIDENCIA Y MATRIZ DE ADYACENCIA.

Cada grafo puede ser representado con estructuras matemáticas, una de estas estructuras consiste en que cada grafo puede tener una matriz de incidencia y una matriz de adyacencia, así :

La matriz de incidencia de un Grafo es la matriz $M(G) = [m^{ij}]$, donde m^{ij} es el número de veces (0, 1 o 2) que v_i y e_j son incidentes. La matriz incidente de un grafo es justo un camino diferente de especificar al grafo.

Otra matriz asociada con G es la matriz de adyacencia, esta es la matriz $v \times v$ $A(G) = [a^{ij}]$, en donde a^{ij} es el número de aristas que unen a v_i con v_j . Se muestra a continuación en la Figura 1.4 un ejemplo de un grafo, en la tabla 1.2 su matriz de incidencia, y en la tabla 1.3 su matriz de adyacencia.

HEURÍSTICAS

Clasificación.

Las heurísticas se clasifican de acuerdo a su tiempo de resolución, cuando necesitan un tiempo polinomial para su resolución se denominan de clase P y cuando se necesita un tiempo exponencial se denomina NP estos a su vez se subdividen en NP duros y NP no duros.

EL PROBLEMA DE SATISFACTIBILIDAD

Dada la dificultad de dar solución a problemas combinatorios de tipo NP (que no tienen un tiempo polinomial de respuesta), y ante la necesidad de ofrecer alguna solución dado su interés práctico, surgen diversos algoritmos que proporcionan soluciones factibles, es decir, que satisfacen las restricciones del problema, las cuales aunque no optimicen la función objetivo, al menos se acercan al valor óptimo en un tiempo de cálculo razonable, podríamos llamarlas en lugar de optimas, "satisfactorias", pues son suficientemente buenas como para servirnos.

Este tipo de algoritmo se denomina heurísticas, del griego Heuriskein.

Una posible manera de definir estos métodos es: como "procedimientos simples, a menudo basados en el sentido común; que se supone ofrecerán una buena solución (aunque no necesariamente la óptima) a problemas difíciles, de un modo fácil y rápido [Zanakis, Evans, 1981].

Es recomendable la utilización de algoritmos heurísticos para la resolución de un problema cuando:

No existe un algoritmo exacto de resolución o este requiere mucho tiempo de cálculo o de memoria.

Cuando no se necesita la solución optima.

Cuando los datos son poco fiables.

Cuando limitaciones de tiempo, espacio (para almacenamiento de datos), etc. Obliguen al empleo de métodos de rápida respuesta, aún a costa de la precisión.

Como paso intermedio en la aplicación de otro algoritmo.

El problema de satisfactibilidad consiste en:

Sea:

$$\text{Min } Z = f(x) \rightarrow \text{función objetivo}$$

$$\forall x \in \Omega \rightarrow \text{Restricción}$$

Se define Ω como el rango donde se encuentra todos los puntos posibles x_i para que la solución sea satisfactoria. Pero para hacer más satisfactorio la solución del problema es necesario acotar al máximo el

rango Ω a Ω_1 , lo cual se lo puede hacer por varios métodos de aproximación, relajamiento, etc. Los cuales se detallan a continuación:

PROBLEMAS DE DECISIÓN

Un problema de decisión es aquel problema que se modela matemáticamente con variables binarias $\{0,1\}$; si o no, aceptado, no aceptado, etc.

Muchos de los problemas que estudiamos pueden ser descritos como un problema de decisión o como problemas de optimización; por ejemplo: ¿Cuál es la longitud del camino mas corto de estas ciudades?, ¿Cuál es el tamaño del conjunto independiente mas largo de vértices en G?.

La teoría de NP-completitud es un intento, con un éxito parcial, para proveer que ciertos problemas son intrínsecamente difíciles de resolver. Por lo que se dará una breve, e informal introducción a esta área teórica tan importante de la ciencia de la computación.

Así, un problema de decisión es especificado por describir una instancia típica del problema, y haciendo la pregunta de ¿cuál es... ? y ser contestada con un "si" o "no".

PROBLEMAS DE OPTIMIZACIÓN

En la resolución de problemas de optimización es frecuentemente importante encontrar máximos y mínimos globales. La programación lineal nos permite afrontar este tipo de problemas en el caso de funciones lineales. En general, la programación lineal permite encontrar máximos o mínimos globales para cualquier función lineal objetivo sujeta a un conjunto de restricciones definidas normalmente por un conjunto de desigualdades lineales.

PROGRAMACIÓN HEURÍSTICA.

Muchos problemas de optimización no pueden ser abordados por métodos exactos, ya sea, por su alto grado combinatorio o por la dificultad de generar un modelo basado en programación matemática que represente exactamente una situación real. Para situaciones de ésta naturaleza se han venido generando desde la década de los sesenta, métodos conocidos como heurísticos, capaces de encontrar soluciones de buena calidad pero en muchos casos aproximada a la solución óptima.

En el primer tiempo se generaron métodos orientados específicamente a la resolución de cada problema, gran parte de estos métodos fueron generados inspirándose en la resolución de problemas de fácil representación pero de muy difícil solución como lo son: el Problema del Agente Viajero; el Problema de la Mochila; el Problema de los Conjuntos de Cobertura; etc.

Por la naturaleza diferente de estos problemas los métodos que se generaron eran útiles apenas para el problema en el cual habían sido inspirados, a partir de los años 80` se han generado una familia de métodos conocidos como meta-heurísticos que ahora tienen la capacidad de ser aplicables a problemas de diversa naturaleza. Es decir, una misma plantilla algorítmica puede ser utilizada para resolver problemas que provienen de diversos sectores.

RECOCIDO SIMULADO.

El Recocido Simulado es el algoritmo que usaremos para resolver nuestro problema, se basa en los conceptos descritos originalmente por la mecánica estadística que describe el proceso físico sufrido por un sólido al ser sometido a un baño térmico (Kirkpatrick y Gelatt, 1983). Este puede ser simulado por el algoritmo de Metrópolis, basado en las técnica de Monte Carlo. Así pues, los estados del sistema se corresponden con las soluciones del problema, la energía de los estados con los criterios de evaluación de la calidad de la solución, el estado fundamental con la solución óptima del problema, los estados metaestables serán los equivalentes a los óptimos locales, y la temperatura correspondería a una variable de control.

Con este artículo se ha querido presentar el concepto de la optimización, analizando las principales herramientas de las que se disponen para poder resolver los distintos tipos de problemas, en función del número de variables que lo componen y de su complejidad. Por último se presentan algunas las últimas técnicas más relevantes para resolver problemas de optimización, siendo éstas tema principal de trabajo en numerosas líneas de investigación.

EL ALGORITMO:

1. $i = i_0$
2. $T = T_0$
3. $K = K_0$
4. while (condicion de STOP)
5. while ($k < K$ && $a < A$)
6. generar j en $N(i)$
7. if ($c(j) - c(i) < 0$)
8. $i = j$
9. $a = a + 1$
10. else
11. generar un numero r al azar
12. if ($r < \exp [(c(i) - c(j))/T]$)
13. $i = j$
14. $a = a + 1$
15. $k = k + 1$
16. $T = \alpha T$
17. $K = \rho K$
18. $k = 0$
19. $a = 0$
20. mostrar $i, c(i)$

PROBLEMA DEL AGENTE VIAJERO

Dado un grafo de N nodos, determinar la ruta más corta que parta de un nodo predeterminado N_i y visite todos los nodos solamente una vez y regrese al nodo de partida.

Este problema parece ser muy simple y puede ser que lo sea para ciertos tamaños de N .

Sin embargo existen instancias de este problema, con valores ni siquiera demasiado grandes de N , para los cuales encontrar la solución es un trabajo terrible. Tan solo basta pensar en una red de computadoras de unos 500 nodos con cierta topología para que sea muy difícil determinar la ruta más corta de un mensaje enviado desde un nodo i determinado y que debe visitar todos los nodos solamente una vez, trayendo de regreso el mensaje enviado y la traza de los nodos visitados.

Para cualquier instancia del problema del agente viajero uno podría pensar en seleccionar todos y cada uno de los diferentes recorridos posibles y luego compararlos para encontrar el de menor longitud. Sin embargo, la ejecución de este algoritmo en una computadora requerirá de aproximadamente $n!$ pasos, lo que para instancias relativamente pequeñas lo vuelve inútil, pero es posible realizarlo.

LEVANTAMIENTO DE INFORMACIÓN.

La recolección de datos partió con la selección de dos puntos estratégicos entre los cuales optimizaríamos las rutas válidas para llegar del punto 1 (P1) al punto 2 (P2).

Para esto se hizo necesario la obtención del plano georeferenciado de la Ciudad de Guayaquil, el mismo que fue facilitado por la empresa Busonil S.A. y que lo tenemos en la Figura 5.1.

Se ha seleccionado como P1 a la Piscina Olímpica Ubicada en las calles Hurtado y Avenida Del Ejercito, la selección de este punto se realizó por ser un punto céntrico en cuyas rutas incluye calles como Clemente Ballén, Cristóbal Colon, Av. Machala, Av. Quito, Esmeraldas, Los Ríos, las más importantes calles del centro de la Ciudad de Guayaquil; Como P2 se ha seleccionado a la Iglesia Catedral de la Ciudad de Guayaquil, por tratarse de un sitio bien visitado con dirección específica, vale recalcar esta información por cuanto en primera instancia se considero al malecón 2000 como punto estratégico, pero vario esta consideración ya que al hablar de Malecón 2000 estamos hablando de varios puntos turísticos debido a su extensión, a la belleza y a la variedad del mismo que lo hacen visitado e interesante en cada punto geoespacial.

El siguiente paso fue determinar rutas Válidas entre P1 y P2 las mismas que están detalladas en la Tabla 1.1.

Tabla 1.1
Rutas entre P1 y P2.

Ruta 1.- Hurtado y Av. Del Ejercito por Av. Del Ejercito sigo hasta Nueve de Octubre sigo hasta girar en Boyaca avanzo hasta Clemente Ballén luego a Chimborazo.

Ruta 2.- Hurtado y Av. Del Ejercito por Av. Del Ejercito sigo hasta Nueve de Octubre sigo hasta girar en Los Ríos avanzo hasta Clemente Ballén luego a Chimborazo.

Ruta 3.- Hurtado y Av. Del Ejercito por Hurtado sigo hasta Los Ríos de allí hasta girar en Clemente Ballén hasta llegar a Chimborazo.

Ruta 4.- Hurtado y Av. Del Ejercito por Av. Del Ejercito sigo hasta Clemente Ballén luego a Chimborazo.

Ruta 5.- Hurtado y Av. Del Ejercito por Hurtado sigo hasta Gabriel García Moreno giro en Clemente Ballén avanzo hasta Chimborazo.

Ruta 6.- Hurtado y Av. Del Ejercito por Hurtado sigo hasta José Mascote giro en Clemente Ballén avanzo hasta Chimborazo.

Ruta 7.- Hurtado y Av. Del Ejercito por Hurtado sigo hasta José Mascote sigo y giro en Cristóbal Colón avanzo hasta Boyaca giro en Clemente Ballén termino en Chimborazo.

Ruta 8.- Hurtado y Av. Del Ejercito por Av. Del Ejercito sigo hasta Nueve de Octubre sigo y giro en Machala avanzo hasta Clemente Ballén luego a Chimborazo.

Ruta 9 .- Hurtado y Av. Del Ejercito por Hurtado sigo hasta José Mascote sigo y giro en Luque avanzo hasta Boyaca giro en Clemente Ballén termino en Chimborazo.

Ruta 10 Hurtado y Av. Del Ejercito por Av. Del Ejercito sigo hasta Nueve de Octubre sigo y giro en José de Antepara avanzo hasta Clemente Ballén luego a Chimborazo.

Ruta 11 Hurtado y Av. Del Ejercito por Hurtado sigo hasta José Mascote sigo y giro en Luque avanzo hasta José de Antepara giro en Clemente Ballén termino en Chimborazo.

Ruta 12 Hurtado y Av. Del Ejercito por Hurtado sigo hasta José Mascote sigo y giro en Cristóbal Colón avanzo hasta Chimborazo luego a Clemente Ballén.

Ruta 13 Hurtado y Av. Del Ejercito por Hurtado sigo hasta José Mascote sigo y giro en 10 de Agosto avanzo hasta Chimborazo luego a Clemente Ballén.

Ruta 14 Hurtado y Av. Del Ejercito por Hurtado sigo hasta Los Ríos sigo y giro en 10 de Agosto avanzo hasta Chimborazo luego a Clemente Ballén

Fuente: Elías David Araujo Carrión

Elaborado por: Elías David Araujo Carrión

Una vez determinadas las rutas se ha procedido a establecer los nodos generados de estas rutas, los que están detallados en la tabla 5.2.

**Tabla 5.2.
Nodos de las rutas Preestablecidas.**

NODOS	DIRECCION
1	PISCINA OLIMPICA
2	Av. Del Ejercito y Nueve de Octubre
3	Hurtado y Los Ríos
4	Av. Del Ejercito y Clemente Ballén
5	Hurtado y Gabriel García Moreno
6	Hurtado y José Mascote
7	Av. Del Ejercito y Nueve de Octubre
8	Nueve de Octubre y Boyaca
9	Nueve de Octubre y Los Ríos
10	Los Ríos y Clemente Ballén
11	Clemente Ballén y Chimborazo
12	Gabriel García Moreno y Clemente Ballén
13	José Mascote y Clemente Ballén
14	José Mascote y Cristóbal Colon
15	Nueve de Octubre y Machala
16	José Mascote y Luque
17	Nueve de Octubre y José de Antepara
18	José Mascote y 10 de Agosto
19	Los Ríos y 10 de Agosto
20	Boyaca y Clemente Ballén
21	Cristóbal Colón y Boyaca
22	Machala y Clemente Ballén
23	Luque y Boyaca
24	José de Antepara y Clemente Ballén
25	Luque y José de Antepara
26	Cristóbal Colón y Chimborazo
27	10 de Agosto y Chimborazo
28	IGLESIA CATEDRAL

Fuente: Elías David Araujo Carrión

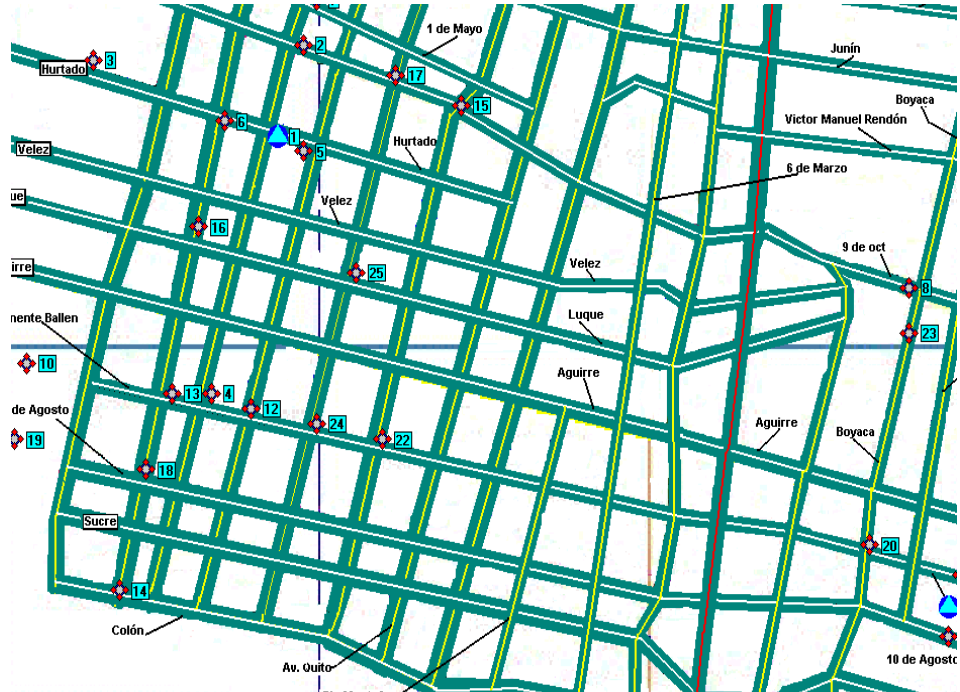
Elaborado por: Elías David Araujo Carrión

Con las dirección de los nodos, procedemos a levantar la información georeferenciada con la ayuda de un receptor GPS (Global Position System), esta recolección de datos se la realizo desde el 11 de Octubre del 2005 al 23 de Octubre del 2005, Esto se detalla en la Tabla 5.3.

Con las Georeferencias obtenidas desde el GPS, ubico cada nodo en el plano de Guayaquil, obteniendo la Figura 5.1.

Figura 5.1.

Ubicación Georeferenciada de los nodos en el Plano de Guayaquil.



Fuente: Busonil S.A.

Elaborado por: Ing. Eduardo Moyano

partir del plano de la Figura 5.1 podemos obtener la Tabla 5.4 (Anexo 1.), que es la matriz de adyacencia entre nodos, en esta matriz las rutas entre nodo y nodo no probables ya sea por que no están conectadas o por ser una sola vía, están representados por valores de 1000 que es cuantificablemente alto e impide optimizar cualquier ruta garantizando que esa ruta no será tomada por el sistema. De igual forma la diagonal Principal de la matriz de adyacencia esta representada por ceros, lo que significa que el costo de estar en el sobre su propio eje es 0. Las rutas validas y probables tienen la matriz de adyacencia asociado su propio costo.

RESULTADOS DEL SISTEMA.

Luego de varias corridas, y de verificar con los parámetros estimados podemos analizarlo siguiente:

- A medida de que se aumente el rango en el costo aumenta el número de iteraciones.
- A medida de que se disminuye el decremento aumenta el número de iteraciones
- Mientras mayor sea el número de iteraciones mayor será el tiempo requerido para que el sistema termine el proceso.
- En este caso se ha comprobado que el número de terminales juega un papel importante ya que existe manera de llegar del P1 al P2 con 2 y 3 terminales, haciendo que el mayor número de terminales provoca un mayor número de combinaciones disminuyendo la probabilidad de llegar al satisfactorio e incluyendo nodos intermedios que son innecesarios, por esta razón se recomienda que el número de terminales no exceda mas de 5.
- La bodega siempre debe ser 1, de acuerdo con la definición de la misma; se refiere a la cantidad de puntos de partida.
- Porque algoritmo usa números aleatorios cada corrida es diferente a las demás, en nuestro caso el resultado depende de los nodos y el costo por ejemplo el sistema es probable que muestre un valor que no sea el mejor sin embargo es posible, esto se produce de acuerdo a como esta llenada la matriz. Los valores imposibles generan rutas con costos superiores a 1000.

RESULTADOS REALES.

El sistema comprobado realmente funciona y es bueno a pesar de sobreestimar el costo satisfactorio en un valor difícil de determinar por ser muy variable, esto se ha determinado recorriendo cada ruta proporcionada por el sistema . Sin embargo, en pocos casos excede de 300 metros.

Mentalmente podemos establecer una ruta optima debido a nuestra racionalidad y conocimiento de la Ciudad, pero por tratarse de un sistema que no piensa y únicamente desarrolla instrucciones usando combinaciones aleatoriamente generadas, puede darse el caso que la ruta por nosotros considerada la mejor nisiquiera se haya generado, recordemos que el algoritmo nos muestra las mejores rutas dándonos la opción de escoger la que mas nos conviene de acuerdo a nuestro criterio o a condiciones externas.

CONCLUSIONES

Se ha podido comprobar que el algoritmo funciona, dando como respuesta resultados válidos en cuanto a recursos computacionales, tiempo, etc.

Como resultados de las corridas se obtienen rutas válidas y no válidas, y se puede crear un criterio adecuado y hacer su elección rápida o automática, las mismas que queda al criterio del agente viajero identificarlas, en nuestro trabajo es muy fácil debido a que las mismas están identificadas porque toman un valor superior a 1000.

El Recocido Simulado V.2 nos presenta en una hoja de Excel varias rutas, haciendo posible la elección de alguna de ellas de acuerdo a criterio del Agente Viajero, atendiendo a requerimiento externo que no estén considerado en el desarrollo del software.

El Recosido simulado V.2 es una versión fácil de utilizar, sencilla en codificación, ocupa poco espacio en disco; haciéndolo un software manejable y adaptable a cualquier necesidad.

El rendimiento del algoritmo se hace visible y útil cuando estamos manejando retas con un número superior a 100 terminales o nodos, ya que en este caso hemos podido comprobar que al traer con rutas de hasta 5 terminales el resultado no suele ser muy satisfactorio. Según estudios realizados por Cerny [1985], considerando que mientras mayor sea el número de estados de energía el criterio de evaluación de la calidad de la solución será mejor.

Se ha comprobado que al tratarse con 3 terminales, el número de combinaciones que debe generarse es $C(28,5)$, haciendo poco probable que aleatoriamente se obtenga nuestra ruta preestablecidas.

El algoritmo en si, nos permite decidir en cuanto a sus parámetros como por ejemplo: si el factor de decremento es exponencial o geométrico, estos sin dejar de ser heurística.

El algoritmo de Recocido Simulado V2. es una estrategia heurística que ha resultado ser muy eficiente en la búsqueda de una solución óptima dentro de un conjunto de soluciones posibles ya que ofrece alguna ventajas como por ejemplo:

BIBLIOGRAFÍA

1. **CASTAÑEDA ROLDAN CAROLINA YOLANDA. (1991)**“Teoría de Grafos”
2. **COLECCIÓN L.N.S. (1992)**, “Metodología De La Investigación Científica”, 3ra Edición.
3. **COLECCIÓN NOVEDADES SOCIALES. (2003)**, “Sobre Tesis y Tesistas.”
4. **DÍAZ FERNÁNDEZ ADENSO, GONZÁLES JOSÉ LUÍS GONZÁLES, LAGUNA MANUEL LAGUNA, MOSCATO PABLO, T. TSENG FAN T. TSENG, FRED GLOVER; HASSAN M. (1996)**, “Optimización Heurísticas y Redes Neuronales “, 1era Edición, España
5. **EDUARDO MORALES MANZANARES**, “**Algoritmo de Metrópolis**”, <http://dns1.mor.itesm.mx/~emorales/Cursos/Busqueda04/node78.html#tmetropolis>.
6. **GONZALES JORGE LUIS GONZALES, Z. RIOS ROGER. INVESTIGACION D EOPERACIONES EN ACCION.** “ Aplicación del TSP en problemas de Manufactura y Logística”
7. **MELIÁN BELÉN, MORENO PÉREZ JOSÉ A, MORENO VEGA J. MARCOS.** “Metaheurísticas: una visión Global”