



UNIVERSIDAD
NACIONAL
AUTÓNOMA DE
NICARAGUA,
MANAGUA

UNAN - MANAGUA

FACULTAD REGIONAL MULTIDISCIPLINARIA DE CHONTALES
FAREM-CHONTALES
PROGRAMA DE DOCTORADO EN MATEMÁTICA APLICADA

RESUMEN

Traveling Salesman Problem (TSP)

**Diseño de Algoritmos Heurísticos y Metaheurísticos
eficientes para resolver el Problema del Agente Viajero**

Autor: M.Sc. José Jesús Mendoza Casanova

Tutor: M.Sc. – Ph.D. Antonio Parajón Guevara

Agosto, 2017

ÍNDICE

1. INTRODUCCIÓN	3
2. PLANTEAMIENTO DEL PROBLEMA.....	5
3. JUSTIFICACIÓN.....	7
4. OBJETIVOS.....	9
4.1 Objetivo General	9
4.2 Objetivos Específicos.....	9
5. MARCO TEÓRICO	10
5.1 Heurístico del Vecino Más Cercano.....	11
5.2 Heurístico GRASP	11
10. METODOLOGÍA	14
11. RESULTADOS Y ANÁLISIS	16
11.1 Comparación con Solver y WinQSB	16
11.2 Comparación del Scatter Search con GRASP y Tabu Search.....	18
12. CONCLUSIONES	20
13. PERSPECTIVAS DE FUTURO SOBRE EL TSP	22
14. BIBLIOGRAFÍA	23

1. INTRODUCCIÓN

La investigación es una actividad orientada a la obtención de nuevos conocimiento y su aplicación para la solución de problemas sociales, o bien cuestiones de carácter científico. En ello radica su importancia.

En el caso de la Matemática, la investigación se produce en este momento en todos los países del mundo, con las limitaciones propias del potencial de cada país, pero sin los inconvenientes de las barreras ideológicas o culturales. El trabajo del matemático en este campo se da tanto a nivel teórico como a nivel aplicado. Y la importancia de este tipo de trabajos radica en el influjo de las necesidades de otras disciplinas del saber humano y los problemas surgidos de su propio contexto.

No bastaría el tamaño de este documento para mencionar los grandes aportes de la matemática a nivel aplicado. Sin embargo, se deben citar algunos ejemplos de impacto de esta ciencia en la sociedad moderna: la teoría de la computación está basada en la Matemática Discreta y de esa relación nació la Teoría de la Complejidad; modelos integro-diferenciales han sido extendidos hacia la epidemiología y genética Molecular, ello ha permitido el diseño de medicamentos y su administración. Otro ejemplo del impacto de su en la sociedad moderna, popularizado por producciones literarias y cinematográficas, son los aportes que hiciera a las teorías económicas el matemático estadounidense, y Premio Nobel de Economía 1994, John Nash; su vida es retratada en la película "A Beautiful Mind", basada en la biografía escrita por Sylvia Nasar.

En cuanto a los grandes aportes de la matemática en el ámbito teórico, el mejor ejemplo es su relación simbiótica con la Física, ello merece un capítulo aparte: son muchos los volúmenes que podrían escribirse para exponer esa relación. Un ápice de ello lo encontramos en los siguientes hechos. La Física moderna se vería en serios aprietos sin los aportes de la Matemática; por ejemplo, la Teoría de la Relatividad no tendría el desarrollo y profundo impacto científico y tecnológico sin las bases proporcionadas por las geometrías no euclidianas.

En Nicaragua, también se hacen esfuerzos con el desarrollo de investigaciones matemáticas. Las universidades son pioneras en estos tipos de trabajos científicos. Las instituciones de educación superior más importantes y más grandes del país, tienen incorporadas academias de jóvenes talentos, la

carrera de matemática, maestrías y hasta doctorados en esta disciplina. Actualmente, en la UNAN-Managua se está desarrollando el Programa de Doctorado de Matemática Aplicada, bajo la coordinación de la Universidad Central “Marta Abreu” de Las Villas, Cuba. Los grandes esfuerzos que nuestro país realiza tienen como finalidad el beneficio de la sociedad.

En esta investigación es de carácter aplicado, y se realiza un estudio sobre el Problema del Agente Viajero (TSP), uno de los problemas más estudiados en Programación Matemática. La importancia de investigar sobre este problema se debe a que es un problema abierto, difícil de resolver, pero que tiene muchas aplicaciones prácticas en la industria, mercados bursátiles, turismo, planificación, logística, fabricación de circuitos electrónicos, secuencia de ADN, etc.

Este problema de apariencia sencilla es famoso por su gran complejidad computacional. Como se verá más adelante, se encuadra dentro de la categoría NP-duro por lo que, a día de hoy, no se ha encontrado ningún algoritmo que logre resolverlo en un tiempo polinómico. Se formula de la siguiente manera: un comerciante debe visitar varios clientes y desea conocer cuál es el camino de mínima distancia que, partiendo de su lugar de trabajo, vaya a todas las ciudades y regrese a la ciudad de origen.

Por tanto, en esta investigación se van a diseñar algoritmos heurísticos y Metaheurísticos para resolver el problema del agente viajero. Para ello se ha estructurado en quince apartados.

2. PLANTEAMIENTO DEL PROBLEMA

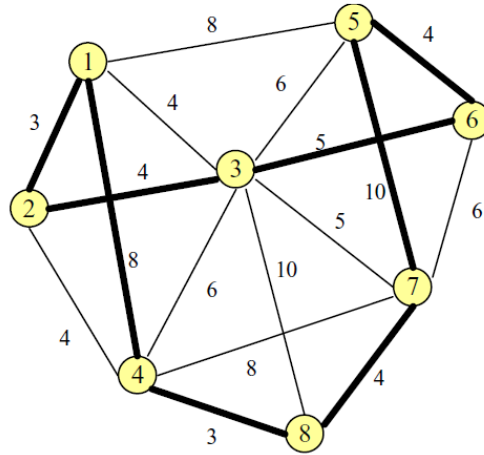
El Problema del Agente Viajero se formula de la siguiente manera:

Dado un conjunto de n ciudades, de las cuales se conoce para cada par de ellas, la distancia que las separa, un agente viajero ha de partir de una ciudad de origen y debe visitar exactamente una vez cada ciudad del conjunto, y retornar al punto de partida. Un recorrido con estas características es llamado, dentro de este contexto, ciclo hamiltoniano. El problema consiste en encontrar el ciclo para el cual la distancia total recorrida sea mínima.

No obstante, para enunciar el problema formalmente se introducen las siguientes terminologías: Sea un grafo $G = (V, A, C)$ donde V es el conjunto de vértices, A es el de aristas y C_{ij} es la matriz de costes. Esto es, C_{ij} es el coste o distancia de la arista (i, j) .

- Un **camino** (o cadena) es una sucesión de aristas (e_1, e_2, \dots, e_k) en donde el vértice final de cada arista coincide con el inicial de la siguiente. También puede representarse por la sucesión de vértices utilizados.
- Un camino es **simple** o elemental si no utiliza el mismo vértice más de una vez.
- Un **ciclo** es un camino (e_1, e_2, \dots, e_k) en el que el vértice final de e_k coincide con el inicial de e_1 .
- Un **ciclo** es simple si lo es el camino que lo define.
- Un **subtour** es un ciclo simple que no pasa por todos los vértices del grafo.
- Un **tour** o **ciclo hamiltoniano** es un ciclo simple que pasa por todos los vértices del grafo.

El Problema del Agente Viajero consiste en determinar un tour de coste mínimo. La Siguiete figura muestra un grafo de 8 vértices en el que aparece destacado un ciclo hamiltoniano.



Es decir que el TSP consiste en encontrar el camino mínimo que recorre todos los vértices de un grafo, saliendo de uno de dichos vértices y retornando a dicho vértice. El problema se puede representar como un grafo completo orientado o bien como una permutación. Los nodos representan las ciudades y los valores asociados a las aristas, son las distancias entre ellas, y por tanto, nunca pueden ser negativos.

3. JUSTIFICACIÓN

El potencial de esta investigación se precisa por la justificación y viabilidad de la misma, a través de los siguientes aspectos:

- **Conveniencia.** Los resultados obtenidos por este estudio serán de mucho provecho por su utilidad práctica para resolver diversos problemas mediante el TSP: industriales, mercados bursátiles, turismo, planificación, logística, fabricación de circuitos electrónicos, secuencia de ADN, etc.
- **Relevancia Social.** Por citar un par ejemplos, las dificultades de transporte y la construcción de un canal interoceánico son los mayores problemas por resolver en Nicaragua. Nuestro país, hoy día, carece de un banco de personal capacitado en el área de la investigación de operaciones, tanto en el ámbito de la investigación como en el académico en las universidades para emprender este tipo de proyectos.
- **Implicaciones Prácticas.** Todo el proceso de investigación y los resultados mismos tendrán implicaciones prácticas debido a que permitirán resolver diversos problemas reales del área de la investigación de operaciones, tales como dificultades o inconvenientes de transporte, asignación de recursos, líneas de espera, mantenimiento y reemplazo de equipo, etc.
- **Valor Teórico.** Esta investigación tiene una valía significativa por tratar uno de los problemas abiertos más importantes de la investigación de operaciones, y más estudiado por diferentes investigadores de muchas naciones, ya que es sumamente difícil obtener solución óptima. Por tanto implicará un valioso aporte al conocimiento científico.
- **Utilidad Metodológica.** Esta investigación tendrá mucha utilidad metodológica por el diseño de algoritmos heurísticos y metaheurísticos, cuyos resultados constituirán un valioso andamiaje para futuras investigaciones del TSP. Asimismo, a nivel universitario, ingenieros de la computación y sistemas, civiles, industriales y electrónicos tendrán suficientes herramientas metodológicas tanto para inserción laboral como para la solución de situaciones específicas.
- **Viabilidad o Factibilidad de Ejecución.** Las principales razones que motivan este estudio son la viabilidad o factibilidad de ejecución del mismo.

La posibilidad de llevar a cabo esta investigación radica, en primer lugar, en la necesidad que tiene actualmente la Facultad de Ciencias e Ingeniería de la UNAN-Managua de desarrollar investigaciones de vital importancia para el país. Asimismo, el aspecto económico no representa ningún factor negativo para el desarrollo de este estudio. Finalmente, la UNAN cuenta con el apoyo de un Doctor graduado en esta área de investigación, y cuya participación en investigaciones de esta naturaleza ha sido muy notoria.

4. OBJETIVOS

4.1 Objetivo General

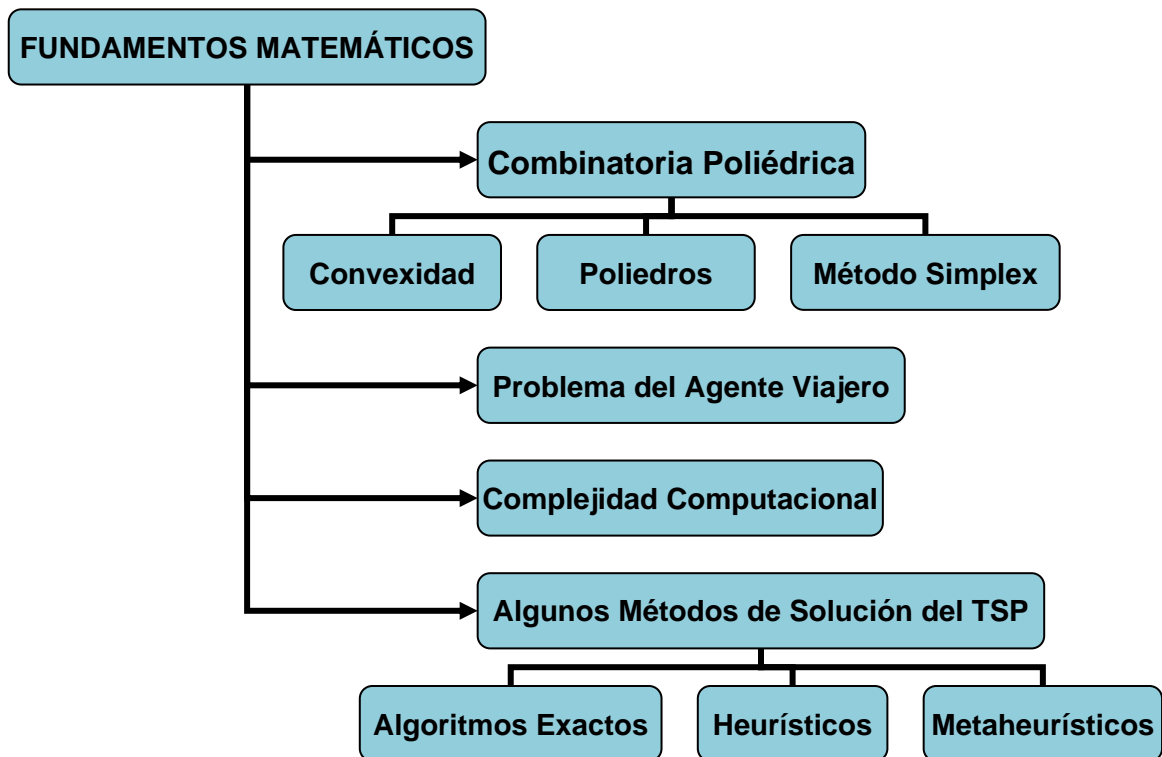
Diseñar algoritmos Heurísticos y Metaheurísticos eficientes para resolver el Problema del Agente Viajero o TSP (Traveling Salesman Problem).

4.2 Objetivos Específicos

- Revisar el estado del arte para Problema del Agente Viajero con los resultados de investigaciones que se han desarrollado al día de hoy.
- Describir la teoría matemática de convexidad para formular el TSP.
- Diseñar los algoritmos heurísticos del Vecino Más Cercano y Fredy Randomized Adaptive Search Procedures (GRASP) para resolver el TSP.
- Diseñar los algoritmos metaheurísticos de Tabu Search y Scatter Search para resolver el TSP.
- Implementar los algoritmos heurísticos y metaheurísticos desarrollados en lenguaje C++ del ambiente de programación Microsoft Visual Studio 2008, Solver de Excel y WinQSB, utilizando las instancias de Traveling Salesman Problem Library (TSPLIB).
- Comparar la eficiencia de los algoritmos propuestos mediante la librería virtual TSPLIB.

5. MARCO TEÓRICO

Los fundamentos matemáticos necesarios para la comprensión y solución del Problema del Agente Viajero (TSP), están organizados en apartados que obedecen a la siguiente estructura:



5.1 Heurístico del Vecino Más Cercano

El Vecino Más Cercano es el heurístico más sencillo e intuitivo para resolver el TSP. Este algoritmo trata de construir un ciclo Hamiltoniano de bajo coste basándose en el vértice más próximo a uno dado. Su planteamiento se debe a Rosenkrantz, Stearns y Lewis en el año 1977 y su código, en una versión standard, es el siguiente:

Algoritmo 2. Pseudocódigo del Vecino Más Cercano

Inicialización

 Seleccionar un vértice j al azar

 Hacer $t = j$ y $W = V \setminus \{j\}$

while ($W \neq \phi$)

 Tomar $j \in W / c_{tj} = \min\{c_{ti} / i \in W\}$

 Conectar t a j

 Hacer $W = W \setminus \{j\}$ y $t = j$

5.2 Heurístico GRASP

Los elementos de este procedimiento son: fase de construcción, de mejora y actualización.

Algoritmo 3. Pseudocódigo del GRASP

while (condición de parada)

Fase Constructiva

Seleccionar una lista de elementos candidatos.

Considerar una Lista Restringida de los mejores Candidatos.

Seleccionar un elemento aleatoriamente de la Lista.

Fase de Mejora

Realizar un proceso de búsqueda local a partir de la solución construida hasta que no se pueda mejorar más.

Actualización

Si la solución obtenida mejora a la mejor almacenada, actualizarla.

5.3 Algoritmo Tabu Search

El principio del Tabu Search podría resumirse como:

Es mejor una mala decisión basada en información que una buena decisión al azar, ya que, en un sistema que emplea memoria, una mala elección basada en una estrategia proporcionará claves útiles para continuar la búsqueda. Una buena elección fruto del azar no proporcionará ninguna información para posteriores acciones.

En resumen, la búsqueda tabú es una combinación de búsqueda local con un mecanismo de memoria a corto plazo. Siendo sus elementos claves los siguientes:

- Restricciones Tabú: restringir la búsqueda al clasificar ciertos movimientos como prohibidos (tabú), para evitar caer en soluciones recientemente generadas.
- Criterio de aspiración: liberar la búsqueda por medio de una función de memoria a corto plazo (olvido estratégico).

En la siguiente tabla se muestra el funcionamiento del algoritmo:

Algoritmo 4. Pseudocódigo del Tabu Search
<ol style="list-style-type: none"> 1. Selecciona un estado $x \in X$ inicial y sea $x^* := x$, $k = 0$ (contador de iteración) y $T := \phi$. 2. Si $S(x) - T = \phi$ ve a 4. Sino $k := k + 1$ y selecciona $s_k \in (S(x) - T)$ tal que $s_k(x) = \text{OPTIMO}\{s(x) : s \in (S(x) - T)\}$. 3. Sea $x := s_k(x)$. Si $c(x) < c(x^*)$ (Donde x^* es la mejor solución encontrada hasta el momento), sea $x^* := x$. 4. Si se agotó el número de interacciones o si $S(x) - T = \phi$; entonces para, Sino, actualiza T (añade el movimiento actual a la lista tabú y posiblemente elimina el elemento más viejo) y regresa a 2.

5.3.2 Algoritmo Scatter Search

La metodología del Scatter Search es muy flexible, ya que cada uno de los elementos puede ser implementado en una variedad de maneras y grado de sofisticación.

Su algoritmo puede resumirse de la siguiente manera:

Algoritmo 5. Pseudocódigo del Scatter Search

1. Comenzar con $P = \phi$. Utilizar el método de generación de la diversificación para construir una solución x . Si $x \notin P$ entonces añadir x a P (es decir, $P = P \cup x$), en caso contrario, descartar x . Repetir esta etapa hasta que P tenga un tamaño prefijado, $|P| = Psize$.
2. Construir el conjunto de referencia $Re\ fSet = \{x^1, \dots, x^b\}$ con soluciones diversas de P . (con las mejores $b/2$ soluciones de P y las $b/2$ soluciones de P más diversas a las ya incluidas.)
3. Evaluar las soluciones en $Re\ fSet$ y ordenarlas de mejor a peor respecto a la función objetivo, tal que x es la mejor solución y x^b la peor.
4. Hacer $NuevaSolución = TRUE$
5. **While** ($NuevaSolución$) **do**
 Generar nuevos subconjuntos, el cual consiste de todos los pares de soluciones en $Re\ fSet$ que incluye al menos una nueva solución.
 Hacer $NuevaSolución = FALSE$
While ($NuevosSubconjuntos \neq \phi$) **do**
 Seleccionar el nuevo subconjunto s en $NuevosSubconjuntos$.
 Aplicar el método de combinación de la solución a s para obtener una o más nuevas soluciones x .
If (x no está en $Re\ fSet$ y $f(x) < f(x^b)$) **then**
 Hacer $x^b = x$ y reordenar $Re\ fSet$
 Hacer $NuevaSolución = TRUE$
end if
 Eliminar s de $NuevosSubconjuntos$
end While
6. **end While**

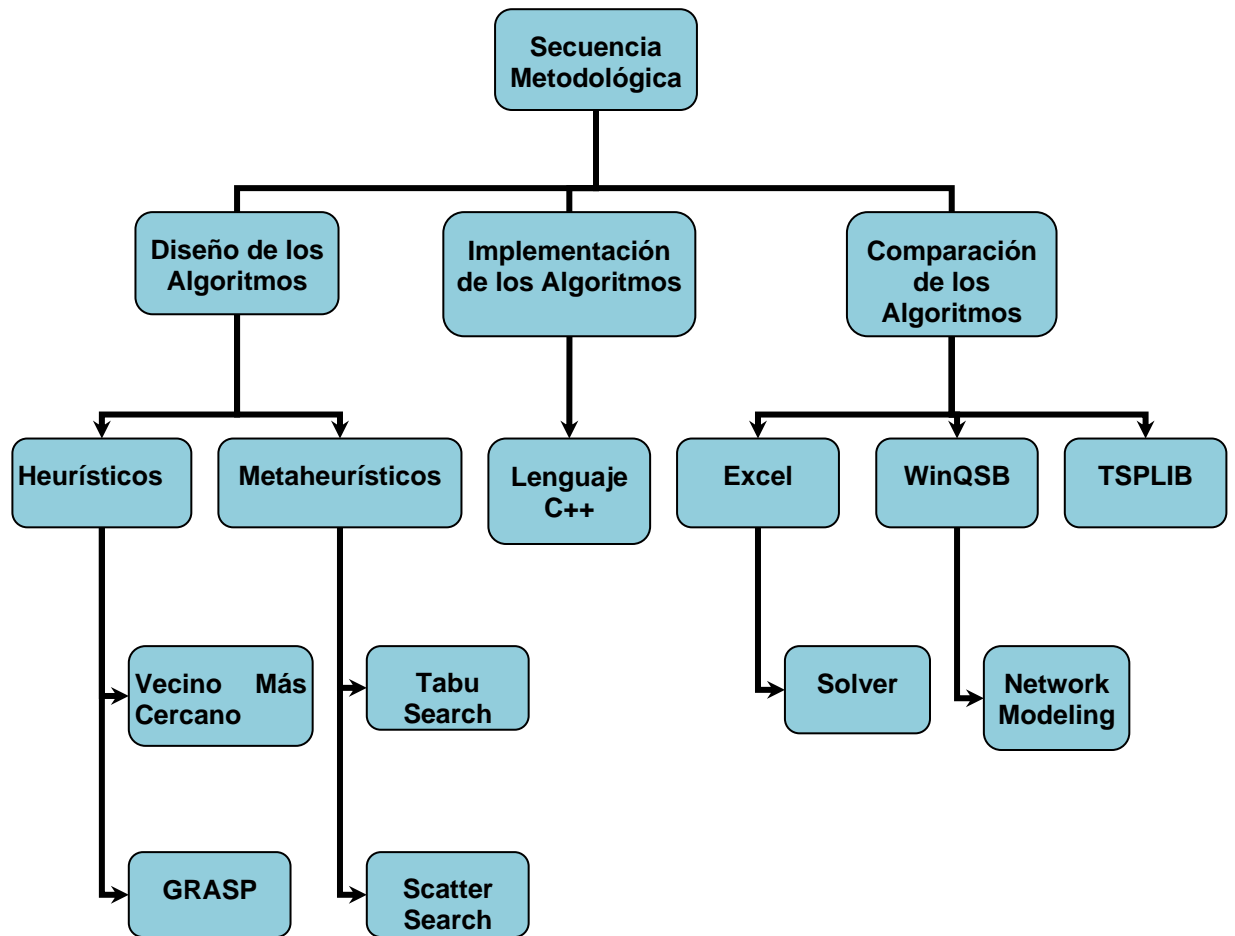
10. METODOLOGÍA

Para alcanzar los propósitos de este trabajo se realizó una revisión del estado del arte, centrándonos en la profundidad y alcance con ha sido abordado el problema del TSP (Traveling Salesman Problem).

Durante esta revisión se tomaron en cuenta las investigaciones con grandes aportes teóricos y prácticos sobre el Problema del Agente Viajero, así como los artículos clásicos y más importantes publicados recientemente según la literatura relevante del TSP. Pérez (2011) propone el estudio de las diez publicaciones más citadas, entre 2006-2010, que tuvieron al TSP como eje central (tema o problema principal de prueba). De igual manera se tomaron en cuenta las tendencias para avanzar alrededor del conocimiento actual sobre el TSP y para mover las puntas de conocimiento hacia otras esferas.

Con la revisión del estado del arte, se pudo encontrar la trayectoria de científicos que han hecho importantes aportes a este problema con el desarrollo de los llamados heurísticos modernos o metaheurísticos. El primero de ellos es el profesor Fred Glover de la Universidad de Colorado, con un total de 57911 citas según el Google Académico. En segundo lugar el profesor de Investigación de Operaciones, Manuel Laguna, también de la Universidad de Colorado con 21758 citas. En tercer lugar se encuentra el profesor de Investigación de Operaciones y Estadísticas Rafael Martí de la Universidad de Valencia con 8150 citas. Y siguiendo esta línea, llegamos a encontrar a mi tutor de Tesis Doctoral el Doctor Antonio Parajón Guevara con aproximadamente 200 referencias en el índice de citas. Esta ha sido en esencia la bibliografía consultada para el desarrollo de este trabajo.

El diseño metodológico de esta investigación se puede apreciar en la siguiente figura. Inicia con el diseño de algoritmos heurísticos y metaheurísticos para resolver el TSP, continua con la implementación en lenguaje C++ y finaliza con la comparación de los resultados con las mejores soluciones de la TSPLIB (Traveling Salesman Problem Library) publicada desde 1991 por Gerhard Reinelt de la Universidad de Heidelberg, Alemania.



11. RESULTADOS Y ANÁLISIS

11.1 Comparación con Solver y WinQSB

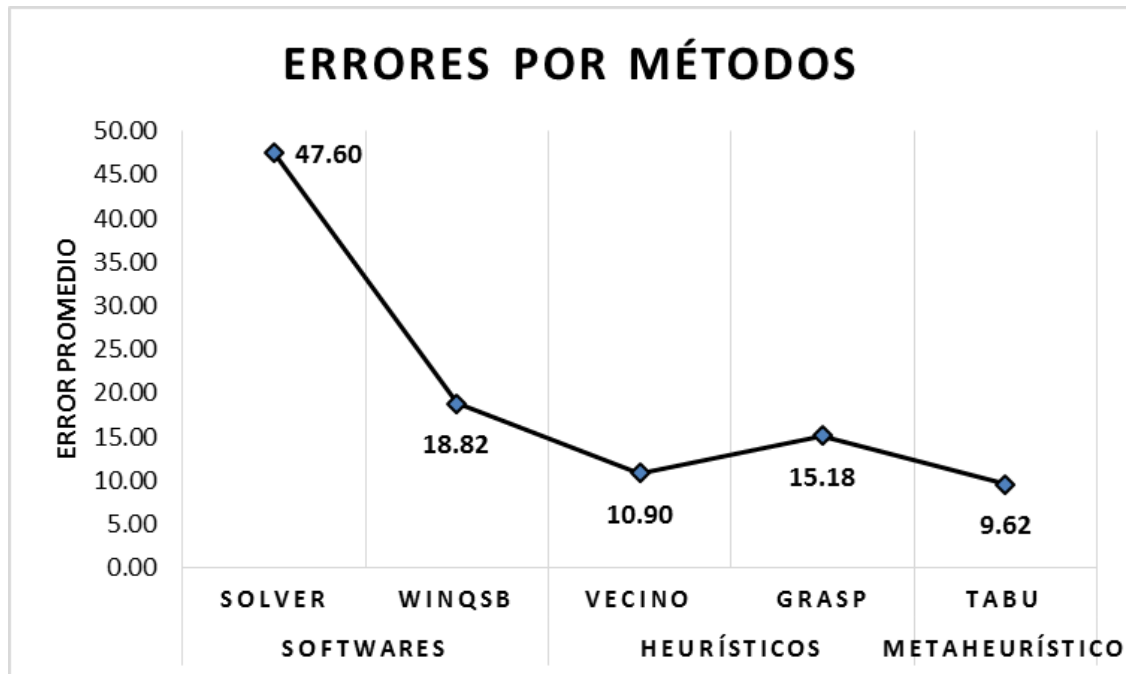
En esta parte de la investigación se presentan los resultados obtenidos para 12 instancias de tamaño medio de la TSPLIB. En la siguiente tabla se muestran los resultados obtenidos con el Solver de Excel, WinQSB, Vecino Más Cercano, GRASP y el Tabú Search:

Tabla de la Función Objetivo

FUNCIÓN OBJETIVO							
NO	TSPLIB		SOFTWARES		HEURÍSTICOS		METAHEURÍSTICO
	INSTANCIA	MEJOR SOLUCIÓN	SOLVER	WINQSB	VECINO 1000-it	GRASP 100000-it	TABU 200-it
1	brazil58	25395	47918	29380	27384	28607	27243
2	dantzig42	699	699	794	864	802	853
3	fri26	937	953	1046	965	937	937
4	gr17	2085	2090	2157	2178	2085	2085
5	gr21	2707	2803	3119	3003	2707	3031
6	gr24	1272	1388	1372	1553	1355	1490
7	gr48	5046	6557	5845	5840	5968	5800
8	gr120	6942	29261	7822	8438	9836	8280
9	hk48	11461	16829	13015	12137	13837	12002
10	si175	21407	26361		22000	33292	22043
11	swiss42	1273	1864	1481	1437	1427	1466
12	tsp8	34	34	34	34	34	34

 Óptimo Alcanzado

El mejor método que se presenta en esta parte de la investigación es el Tabu Search. Su error promedio al resolver las instancias es de apenas 9.62%, como lo muestra la siguiente gráfica:



Y este error podría ser mucho menor en caso de tener una computadora con más memoria y mayor capacidad de procesamiento. El Tabu Search fue ejecutado con solamente el análisis de 200 entornos, con un mayor número genera enormes archivos de escritura que contienen las soluciones. Pero estos archivos no solo contienen las soluciones sino que grandes cantidades de iteraciones por entornos y enormes Lista Tabú, lo que los hace muy pesados e imposibles de abrir para obtener la solución que contienen.

11.2 Comparación del Scatter Search con GRASP y Tabu Search

Se puede observar en la siguiente tabla que el algoritmo que obtuvo mejores resultados fue el Scatter Search con parámetros Psize = 100 y RefSet = 20. Además, de tener la media y desviación típica más baja respecto a todos los demás, consigue un mayor número de mínimos valores. Aquí no se habla de valores óptimos porque las instancias tienen las distancias con números decimales y no se pudo leer perfectamente estos valores decimales por tal razón se habla de valor mínimo conseguido con los algoritmos propuestos.

FUNCIÓN OBJETIVO									
INSTANCIA	SCATTER			GRASP			TABU		MÍNIMO
	100;10	100;20	100;30	4	2	1	40;1000	80;2000	
bier127	118288	118256	118336	118613	119016	119116	124831	124831	118256
d198.tsp	15823	15793	15809	15816	15843	15814	16175	16189	15793
eil101.tsp	619	611	611	613	618	625	699	657	611
fl417.tsp	11798	11842	11845	11779	11867	11844	12255	12227	11779
gil262.tsp	2350	2388	2368	2344	2358	2374	2452	2452	2344
kroA100.tsp	21343	21247	21247	21247	21247	21247	22172	22198	21247
kroA150.tsp	26675	26485	26517	26747	26707	26911	28412	28364	26485
kroA200.tsp	29702	29604	29487	29521	29620	29804	31421	31400	29487
kroB100.tsp	22207	22086	22126	22091	22086	22150	23236	23236	22086
kroB150.tsp	26499	26063	26079	26078	26142	26279	29120	28900	26063
kroB200.tsp	30000	29530	29654	29665	30017	30077	30992	30857	29530
kroC100.tsp	20723	20703	20703	20703	20723	20703	23128	23137	20703
kroD100.tsp	21526	21249	21249	21301	21265	21331	22336	22343	21249
kroE100.tsp	22053	22093	22075	22075	22029	22033	22831	22831	22029
lin105.tsp	14392	14354	14354	14354	14354	14354	15062	15062	14354
lin318.tsp	43235	43170	43451	42982	43232	43200	44386	44154	42982
pr107.tsp	44302	44369	44280	44280	44280	44280	45729	45781	44280
pr124.tsp	59011	59011	59011	59011	59011	59011	60232	59882	59011
pr136.tsp	97444	96967	96977	98481	98130	98103	105500	105209	96967
pr144.tsp	58572	58521	58521	58521	58521	58521	59106	59106	58521
pr152.tsp	73780	73759	73622	73622	73622	73626	75196	75196	73622
pr226.tsp	80803	80431	80640	80506	80546	80909	81038	81038	80431
pr264.tsp	49697	49861	49685	49428	49459	50605	53918	53894	49428
pr299.tsp	48580	49376	49366	48892	49418	49990	52686	52306	48580
rat195.tsp	2357	2329	2352	2291	2341	2401	2432	2444	2291
rd100.tsp	7893	7858	7858	7858	7864	7858	8361	8358	7858
rd400.tsp	15883	16037	15971	15555	15768	15968	16362	16320	15555

ts225.tsp	126936	126809	127129	127335	126850	127129	128381	127580	126809
tsp225.tsp	3925	3901	3912	3849	3924	3935	4057	4017	3849
u159.tsp	42499	42045	42045	42045	42118	42289	44362	44355	42045
Media	37963.8	37891.6	37909.3	37920.1	37965.9	38082.9	39562.3	39477.5	
Desv. Típica	32750.6	32717.5	32756.0	32884.8	32846.2	32900.8	33906.5	33812.2	

 **Mejor Valor Encontrado**

12. CONCLUSIONES

- Se realizó una revisión exhaustiva del estado del arte para el Problema del Agente Viajero (TSP). Este es uno de los problemas más importantes de la optimización combinatoria, tanto por su complejidad computacional, como por sus múltiples aplicaciones. Actualmente hay disponibles en las revistas indexadas una cantidad enorme de propuestas científicas para resolver el TSP, desde nuevos teoremas sobre su estructura, así como aproximaciones muy buenas como planos de corte, Simulated annealing, GRASP, Tabu Search, Scatter Search, etc. Dentro de los científicos que más tiempo y esfuerzo le han dedicado a este problema están Fred Glover y Manuel Laguna de la Universidad de Colorado (EEUU), así como Rafael Martí de la Universidad de Valencia (España).
- Se estructuraron los resultados teóricos de Análisis Convexo para formular y resolver el Problema del Agente Viajero. Dicha estructura era realmente necesaria dado que el TSP se puede plantear como un problema de grafos, y estos a su vez tienen formas convexas.
- Se diseñaron los algoritmos Heurísticos del Vecino Más Cercano y Greedy Randomized Adaptive Search Procedures (GRASP), y los Metaheurísticos Tabu Search y Scatter Search para resolver el TSP. Estos funcionaron muy bien sobre las instancias de la librería TSPLIB, alcanzando algunos de los óptimos y en promedio, el algoritmo Tabu Search que es el más prometedor para instancias de tamaño pequeño y mediano alcanza un índice del 90% de calidad. En el caso de este algoritmo la fase de mejora funciona utilizando ideas de inteligencia artificial, en cuanto a la memoria a corto plazo que implementa mediante estructuras simples con el objetivo de dirigir la búsqueda teniendo en cuenta la historia o registro computacional de ésta, a través de la Lista Tabú evitando quedarse atrapado en un óptimo local. Con respecto al Scatter Search, es el mejor para instancias grandes, tiene el mismo orden de eficiencia que el Tabu Search. Realiza la búsqueda de mejores soluciones mediante exploración sistemática sobre una serie de buenas soluciones llamadas conjunto de referencia que se ha de ir actualizando.
- Se comparó la calidad de los algoritmos diseñados para el TSP, al implementar y obtener resultados usando lenguaje C++ (ambiente de programación Microsoft Visual Studio 2008), Solver de Excel y WinQSB,

utilizando las instancias de Traveling Salesman Problem Library (TSPLIB). Se hace notar que las computadoras disponible para esta investigación no eran ni de las más modernas ni las más rápidas, de haber contado con hardware disponibles como en universidades desarrolladas y de primer mundo se habría llegado al óptimo de la mayoría de las instancia, en un tiempo razonable.

- El análisis de los resultados obtenido con los algoritmos heurísticos y metaheurísticos, indica que el Vecino más Cercano, GRASP y Tabu Search, son más eficientes que los optimizadores WinQSB y Solver de Excel. No obstante, el mejor algoritmo diseñado para instancias pequeñas y medianas fue el Tabu Search y para instancias grandes fue el Scatter Search.
- Tomando en cuenta la eficiencia que debe tener un método para resolver el Problema del Agente Viajero, se concluye que el orden decreciente de eficiencia es el siguiente: Scatter Search, Tabu Search, GRASP, Vecino más Cercano, WinQSB y Solver.

13. PERSPECTIVAS DE FUTURO SOBRE EL TSP

- Para investigaciones futuras se planea trabajar en las fases de mejora de los algoritmos diseñados, incorporando nuevas ideas de inteligencia artificial. También se harán las adecuaciones necesarias para tratar con las instancias asimétricas de la TSPLIB.

14. BIBLIOGRAFÍA

- Álvarez, R., Corberán, A., Tamarit, J. (1985). *La combinatoria poliédrica y el problema del viajante. Aplicación al caso de ciento tres ciudades Españolas*. *Qüestió*, v. 9, n. 3, p. 199-213, 1985.
- Álvarez-Valdés, R., & Parajón, A. (2002). *A tabu search algorithm for large-scale guillotine (un) constrained two-dimensional cutting problems*. *Computers & Operations Research*, 29(7), 925-947.
- Alvarez-Valdes, R., Martí, R., Tamarit, J. M., & Parajón, A. (2007). *GRASP and path relinking for the two-dimensional two-stage cutting-stock problem*. *INFORMS Journal on Computing*, 19(2), 261-272.
- Alvarez-Valdes, R., Parajón, A., & Tamarit, J. M. (2001, July). *A computational study of heuristic algorithms for two-dimensional cutting stock problems*. In *4th metaheuristics international conference (MIC'2001)* (pp. 16-20).
- Alvarez-Valdes, R., Parajón, A., & Tamarit, J. M. (2002). *A computational study of LP-based heuristic algorithms for two-dimensional guillotine cutting stock problems*. *OR Spectrum*, 24(2), 179-192.
- Applegate, D. Bixby, R. Chvatal, V. Cook. (2006). *The Traveling Salesman Problem*.
- Avila, J. (1995). *Método de Karmarkar* *Revista de Matemática: Teoría y Aplicaciones* 2(1): 45-55.
- Bazaraa, M.S., Jarvis, J.J., Sherali, H. (1990). *Linear Programming and Network Flows, 2nd Edition*. John Wiley & Sons, New York.
- Bertazzi, L., Paletta, G., Speranza, M.G. (2004). *An improved heuristic for the period traveling salesman problem*. *Computers & Operations Research*. Vol. 31, 2004, p.1215-1222.
- Calviño, A. (2011). *Cooperación en los problemas del viajante (TSP) y de rutas de vehículos (VRP): una panorámica*. Santiago de Compostela.

- Cassini, L., Righini, G. (2004). Heuristic algorithms for the TSP with rear-loading. In: ANNUAL CONFERENCE OF THE ITALIAN OPERATIONAL RESEARCH SOCIETY (AIRO XXXV), 35, 2004, Lecce, Italy.
- Chao, I.M., Golden, B.L., Wasil, E.A. (1995). *A new heuristic for the period traveling salesman problem*. *Computers & Operations Research*. Vol.22, 1995b, p.553-565.
- Chen S., Chien, C. Solving the traveling salesman problem based on the genetic simulated annealing ant colony system with particle swarm optimization techniques. *Expert Systems with Applications*, v. 38, n. 12, p. 14439-14450, 2011. <http://dx.doi.org/10.1016/j.eswa.2011.04.163>
- Cook S.A. (1971). *The complexity of theorem-proving procedures*. ACM Press, New York, USA. 151-158.
- Dantzig, G., Fulkerson, R., Johnson, S. (1954). Solution of a large-scale traveling-salesman problem. *Journal of the operations research society of America*, v. 2, n, 4, p. 393-410.
- Feo, T. and Resende, M.G.C. (1989), A probabilistic heuristic for a computational difficult set covering problems, *Operations research letters*, 8, 67–71.
- Feo, T. and Resende, M.G.C. (1995), Greedy Randomized Adaptive Search Procedures, *Journal of Global Optimization*, 2, 1–27.
- Glover, F. (1986) "Future Paths for Integer Programming and Links to Artificial Intelligence", *Computers and Operations Research*. 5. 533-549.
- Glover, F. and Laguna, M. (1997). *Tabu Search*, Kluwer Academic Publishers.
- Glover, F., Laguna, M. and Martí R. (2006) *Principles of Tabu Search*. To appear in *Approximation Algorithms and Metaheuristics*, Chapman & Hall/CRC.
- González, C., Sánchez, M. (1991). *El método de Karmarkar: un estudio de sus variantes*.
- Guevara, R. A. P. (2001). *Algoritmos heurísticos eficientes para problemas de corte en dos dimensiones* (Doctoral dissertation, Universitat de València).

- Held, M., Karp, R.M. (1962). *A dynamic programming approach to sequencing problems*. Journal of the Society of Industrial and Applied Mathematics 10, 196-210.
- Jimenez, G. (2009). *Optimización*. Universidad Nacional de Colombia.
- Martí, R. (2003). Procedimientos metaheurísticos en optimización combinatoria. *Matemáticas*, v. 1, n. 1, p. 1-60.
- Osman, I.H. and Kelly, J.P. (eds.) (1996), *Meta-Heuristics: Theory and Applications*, Ed. Kluwer Academic, Boston.
- Parajón Guevara, A. (2009). El Álgebra y su Tratamiento Metodológico y sus Aplicaciones. *Módulo III. Managua, Nicaragua*.
- Parreño Torres, F. (2004). Algoritmos heurísticos y exactos para problemas de corte no guillotina en dos dimensiones.
- Pérez, J. (2011). *Heurística inspirada en el análisis sistémico del "Vecino más cercano", para solucionar instancias simétricas TSP, empleando una base comparativa multicriterio*. Universidad Nacional de Colombia, Medellín.
- Pérez, J., Jaramillo, G. (2011). *Espacio literario relevante sobre el problema del vendedor viajero (TSP): contenido, clasificación, métodos y campos de inspiración*.
- Reeves, C.R. (1995), *Modern Heuristic Techniques for Combinatorial Problems*, Ed. McGraw-Hill, UK.
- Ríos, R., González, J. (2000). *Investigación de operaciones en acción: Heurísticas para la solución del TSP*.
- Rosenkrantz, D. J., Stearns, R. E., & Lewis, II, P. M. (1977). An analysis of several heuristics for the traveling salesman problem. *SIAM journal on computing*, 6(3), 563-581.
- Taha, H. (2011). *Investigación de operaciones*. University of Arkansas, Fayetteville. Pearson. Novena edición.
- Zanakis, S. H. y Evans, J. R. (1981). Heuristic Optimization: Why, When and How to Use It. *Interfaces*. Vol. 11(5) .