

Consulte debates, estadísticas y perfiles de autores de esta publicación en: <https://www.researchgate.net/publication/341405739>

Algoritmo de búsqueda tabú (TSA): una encuesta completa

Documento de conferencia · Febrero 2020

DOI: 10.1109/ICETCE48199.2020.9091743

CITAS

27

LECTURAS

873

3 autores:



Vishnu Kumar Prajapati

Instituto Nacional de Tecnología, Hamirpur

7 PUBLICACIONES 27 CITAS

VER EL PERFIL



Mayank Kumar Jain

Universidad Manipal Jaipur

11 PUBLICACIONES 69 CITAS

VER EL PERFIL



Lokesh Chouhan

Universidad Nacional de Ciencias Forenses Gandhinagar

41 PUBLICACIONES 339 CITAS

VER EL PERFIL

Algoritmo de búsqueda tabú (TSA): un completo Encuesta

Vishnu Kumar Prajapati
Ingeniería en Ciencias de la
Computación Instituto Nacional de
Tecnología Hamirpur (HP),
India vishnu08jec@gmail.com

Mayank Jain
Ingeniería en Ciencias de la
Computación MNIT
Jaipur, India Jaipur
(Rajasthan), India mayank261288@gmail.com

Dr. Lokesh Chouhan
Instituto Nacional de Tecnología
de Ingeniería en Ciencias de la
Computación Hamirpur
(HP), India lokesh@nith.ac.in

Resumen: este artículo presenta una encuesta completa sobre algoritmos de búsqueda tabú (TSA). TSA es un tipo de algoritmo metaheurístico que trabaja en la solución óptima global para un problema determinado, como el problema de enrutamiento de vehículos (VRP), el problema de enrutamiento de vehículos abierto (OVRP), el problema de programación y enrutamiento de vehículos de viajes múltiples (MTVRSP), el problema de carga de contenedores. (CLP) y el problema del taller de trabajo, etc. en este artículo se centra en las características principales de TSA y su comportamiento.

Términos del índice : optimización combinatoria, búsqueda tabú, metaheurística, rutas para vehículos, ruta hamiltoniana, rutas y programación abiertas para vehículos.

I. INTRODUCCIÓN

Una técnica de optimización ha funcionado con diferentes campos, como redes distribuidas, minería de datos, inteligencia artificial, redes neuronales, redes de sensores inalámbricos, etc. Se puede clasificar como metaheurísticas y heurísticas, y otras metaheurísticas se pueden clasificar con base local y Algoritmos basados en la evolución como se muestra en la figura 1.1. TSA son algoritmos metaheurísticos que están vagamente conectados con la computación evolutiva. Podría ser abordar problemas NP-difíciles, es decir, problemas de optimización combinatoria. La TSA reduce las regiones críticas en el espacio de búsqueda aplicando este enfoque. Se aplican varios métodos de diversificación e intensificación.

Depende del problema específico y del tipo de problema y se puede utilizar qué tipo de soluciones son internas o externas, pero dará mejores resultados dentro del conjunto que se busca. Los TS están utilizando la memoria a corto o largo plazo, así como a mediano plazo, para lograr diversificación e intensificación. Además, los criterios de aspiración se pueden utilizar para optimizar el proceso. Similar al algoritmo TS son los trabajos descritos por Glover Fred en 1989. La metaheurística TS tiene varias variantes para resolver los problemas de optimización combinatoria que son TSP, VRP, OVRP, Job Shop Problem, MTVRSP y varios problemas relacionados con gráficos.

Fred W. Glover propuso por primera vez una búsqueda tabú (TS) en 1986, incluyó varios conceptos básicos aplicando muchas ideas que se sugirieron en el siglo XVI y se formularon en 1989. El artículo de Glover 1989 y de Glover , 1990 propuso todo el principio de búsqueda tabú, conocido como TSA básica. El algoritmo TS fue

Comenzó a superar los mínimos locales o máximos locales que se conocen como óptimos locales. Existen varios métodos que deben superar el problema de los óptimos locales. En ese momento, esta búsqueda proporcionó un cambio drástico para reducir el problema de los óptimos locales. Al principio, ni el investigador ni D. de Werra del Instituto Federal Suizo de Tecnología de Lausana entendían bien la búsqueda tabú. Descubrió que la palabra tabú o tabú proviene del tongano, que es un idioma polinesio y, según Webster, el tabú o tabú es una palabra para definir a los apartados como cargados de un poder sobrenatural peligroso y prohibido el uso o contacto profano que está asociado con regiones difíciles. .

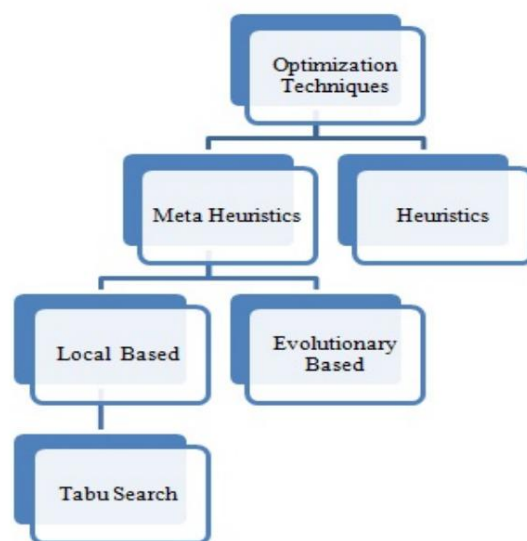


Figura 1.1. Clasificaciones de técnicas de optimización

Los algoritmos TS de las últimas dos décadas se han aplicado en varios problemas prácticos de optimización combinatoria. El problema de la optimización se puede resolver de varias maneras, como costo mínimo, distancia mínima, rendimiento máximo, etc. Existen varias aplicaciones relacionadas con la programación, las telecomunicaciones y la reorganización del carácter en el lenguaje de procesamiento natural. En este tipo de situación, la búsqueda tabú proporciona una solución óptima o casi óptima. Por Fred Glover, el tabú

Las búsquedas se aplican a una amplia clase de problemas. TS se puede aplicar directamente en varios tipos diferentes de problemas de decisión sin transformarse en formulaciones matemáticas [5].

II. MARCO DE BÚSQUEDA TABÚ

Actualmente podemos discutir los conceptos de TS. TS proporcionará la información necesaria sobre varias estrategias de diversificación e intensificación de la búsqueda que se utilizan en TS.

A. Descripción general

La característica básica del TS fue diseñada por Fred Glover para reducir el problema de los óptimos locales; en ese momento, otras técnicas de búsqueda locales se enfrentan a este problema. El autor ha utilizado funciones básicas y suficientes en ese momento, pero algunas funciones no se utilizan en ese momento. Ahora aquí usaremos el principio básico de TS descrito por Fred Glover et [2], [3] y [4]. Los TS básicos están explorando diferentes áreas para el espacio de búsqueda y la inteligencia artificial es una de las áreas que utiliza el algoritmo TS. Entonces, al usar la inteligencia, el TS se vuelve más efectivo y la resolución de problemas en la memoria adaptativa, así como la capacidad de respuesta del problema relacionado con la memoria, como la programación y el problema del taller. Utilizó una manera directa e iterativa para resolver el problema dado.

Muchos investigadores han trabajado durante la última década para mejorar la búsqueda tabú y encontrar la solución óptima o casi óptima para un problema deseable determinado. Los conceptos básicos y las características de TSA no han cambiado, incluso se han agregado algunas otras características, como la capacidad de respuesta de la memoria, razón por la cual TS es tan popular [4].

B. Memoria adaptativa

TS trabaja con una búsqueda inteligente de un problema específico, sobre la base de un proceso previo basado en datos históricos que se almacenan en la memoria, en función de acciones pasadas a realizar en el proceso de búsqueda.

El primer almacén de datos realizó una acción en el tiempo de búsqueda y utiliza la estructura para almacenar datos y aplicar TS. TS enfrenta desafíos que se utilizan cuando alguna vez se almacenan datos en el espacio de almacenamiento de la memoria. En los datos almacenados, la memoria explícita y la memoria atributiva registran la solución completa. La memoria son registros almacenados con una solución consistente y muy atractiva y no exploran a los vecinos, por lo que este tipo de memoria se conoce como memoria explícita. Este tipo de memoria se utiliza para definir el vecino y ampliar su vecindad en el algoritmo TS. En la memoria que almacena los datos o la información sobre la base de un atributo para crear la mejor solución, ese tipo de memoria se conoce como memoria atribuida. Los aspectos básicos de los atributos son los valores de funciones o variables que se utilizan para crear una solución óptima. Algunos de los atributos utilizan la función hash para almacenar los datos que pueden ser buscados de manera más eficiente.

En los TS se utilizan ambos tipos de memoria para realizar la mejor estrategia de búsqueda.

La memoria puede estructurarse a corto o largo plazo.

TS utiliza ambos tipos de estructura de memoria y cada uno tiene su propia estrategia y característica [6]. Según el requisito, podemos usar cortos

memoria a plazo o a largo plazo y modificó la vecindad utilizando la solución actual para la siguiente nueva solución. Para encontrar la nueva solución, la vecindad modificada se contiene en datos históricos o registros y, según el tipo de estructura de memoria (a corto o largo plazo) utilizada en diferentes estados de iteración en el algoritmo TS. Entonces, los tipos de estructura de la memoria contienen básicamente cuatro principios que se basan en la calidad, la frecuencia, la influencia y la regencia [8]. La memoria basada en la calidad se usa para la capacidad, la influencia se usa para el impacto, la regencia se usa para mantener el seguimiento y la frecuencia se usa para realizar un seguimiento de la solución, que se usa muy comúnmente en el pasado y todos trabajan juntos para que TS brinde un mejor rendimiento.

C. Mecanismo de movimiento

Algoritmo de búsqueda tabú sobre la solución actual que puede aplicar este mecanismo reemplazando los bordes seleccionados. En este mecanismo, el intercambio de bordes tiene lugar cuando se intercambian el borde seleccionado actual y los nuevos bordes seleccionados. Primero, todos los bordes seleccionados actualmente se almacenan en una lista conocida como lista de candidatos. Después de eso, cada movimiento crea una nueva solución posible, selecciona la mejor de ellas y aplica el mecanismo de movimiento. Cuando se eligen mecanismos de movimiento, existen algunas restricciones que se deben a que al seleccionar bordes no hay ningún ciclo allí. Las selecciones de aristas satisfacen el requisito del problema dado. Hay dos tipos de procesos de intercambio, uno es estático y otro dinámico. Estos intercambios de bordes estáticos y dinámicos pertenecen a Yajie Tian en 2000. El ejemplo del mecanismo de movimiento como se muestra en la siguiente figura 2.2 en este ejemplo, hay una línea de puntos que muestra la eliminación del borde y una línea oscura que muestra el borde agregado. Aquí se intercambian dos bordes mediante el uso del mecanismo de movimiento, podemos seleccionar los dos bordes y conectar estos bordes seleccionados que pertenecen a la solución actual que se puede intercambiar y estos procesos se realizan sin ningún ciclo en el gráfico.

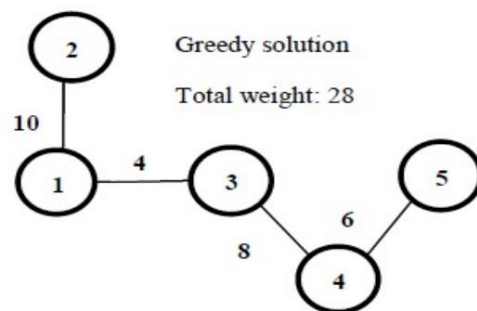
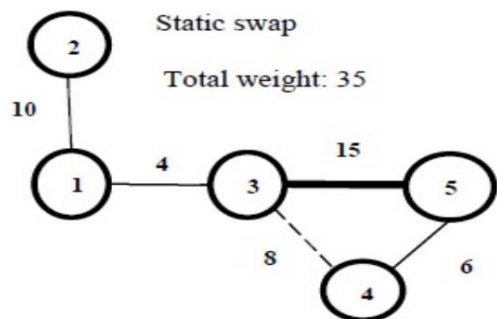


Figura 2.1. Enfoque codicioso



Primero busque todos los bordes de movimiento, selecciónelos y guárdelos en la lista de candidatos. Después de eso, mediante el mecanismo de movimiento, seleccione el mejor movimiento e intercambie los bordes seleccionados actuales por los nuevos bordes seleccionados para obtener la mejor solución aplicando TS. Aquí, para todos y cada uno de los pasos del proceso de búsqueda, necesitamos descubrir todos los movimientos que se aplicaron en la solución actual. Este proceso se realiza de forma iterativa.

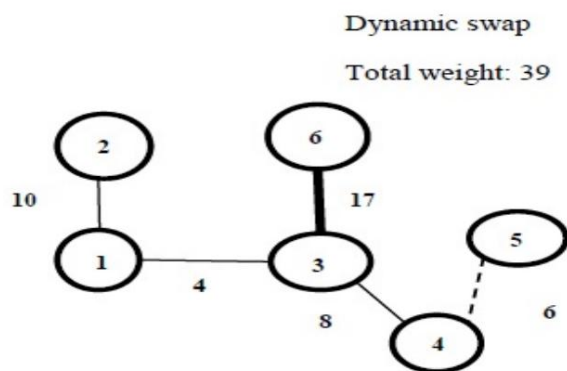
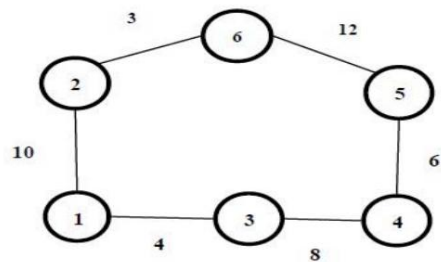
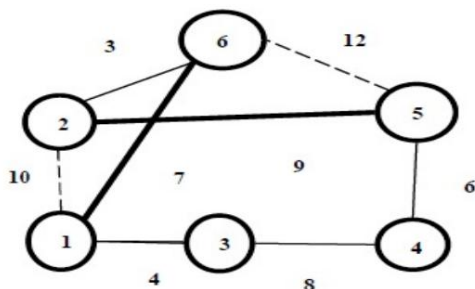


Figura 2.2. Mecanismo de movimiento

2 edge exchange
Total weight 37



Greedy Solution

Total weight 43

En la figura anterior, intercambie la arista o el proceso de intercambio se realiza para realizar 3 aristas. Aquí, los bordes de intercambio deben volver a conectarse a estos bordes en cualquier ciclo en el gráfico existente.

D. tabú

El nodo y el borde están restringidos al elemento en la siguiente iteración de la búsqueda definida por la búsqueda tabú. En TS, ignore el ciclismo y optimice el costo del camino para encontrar la solución óptima. El vecindario debe aplicarse cuando se aplican mecanismos de movimiento en la búsqueda tabú. Aquí lo más importante es que al convertirse en diversas situaciones como óptimos locales, deben resolver la búsqueda (óptimos locales) desde el seguimiento de los pasos anteriores hasta su origen.

Ese tipo de logros son posibles mediante la búsqueda tabú. Se grabó en un esquema de memoria a corto plazo para realizar el paso del estado actual al siguiente estado de solución posible. En la búsqueda Tabú, la lista de candidatos tiene información de almacenamiento limitada por un período de tiempo limitado. Podemos almacenar soluciones completas pero necesita mucha memoria de almacenamiento. Cuando se expande la memoria, el movimiento resulta costoso para verificar si el movimiento es necesario o no en la búsqueda tabú. Por eso es necesario definir el tamaño de la lista de candidatos en la búsqueda tabú. Por lo tanto, el almacenamiento se basa en técnicas de eliminación por orden de llegada. En la solución inicial podemos aplicar mover y encontrar la lista de candidatos en la búsqueda tabú [15].

E. Criterios de aspiración Tabús

Puede haber algún movimiento atractivo pero prohibido, incluso si no es un peligro o una condición crítica como andar en bicicleta, pero aún así estos movimientos dan resultados atractivos y pueden manejar el estancamiento del proceso de búsqueda de TS. Este tipo de situaciones son necesarias para algunas condiciones que se conocen como condición o criterio de aspiración. Esto también se conoce como factor clave sensible en TSA. Se utiliza para la mayoría de los criterios de aspiración simples.

Los criterios de aspiración permiten avanzar de esa manera. Es mejor avanzar utilizando estos criterios. Con una mejor solución basada en la solución más conocida que permite Tabu. Estas condiciones se denominan criterios de aspiración.

F. Intensificación

Si necesitamos una solución más eficaz y mejor admisible, entonces existen algunas características y componentes, uno de ellos la intensificación de la búsqueda tabú. Es como explotación. Esta estrategia se basa en mejorar las reglas de elección. Está basado en

entre la memoria a corto plazo y la memoria perdida que se conoce como memoria intermedia. La memoria intermedia es una memoria reciente que se aplica a las soluciones o combinaciones iniciales o mejores. Utiliza un algoritmo tradicional sin interrupción. La intensificación puede ser generar vecinos mediante estrategias de injerto o evaluación para una buena solución. Se utiliza en muchas implementaciones de TS, pero no es necesario examinar cuándo el proceso de búsqueda normal proporciona la mejor solución posible o lo suficientemente exhaustiva. En este caso no es necesario examinar más detenidamente estos nodos que ya hemos visitado. En la búsqueda tabú se puede utilizar la resolución óptima de problemas local cuando sea necesario y lograr una buena solución. Por lo tanto, siempre que no logramos explorar las otras partes interesantes de un espacio de búsqueda determinado, se utiliza una estrategia de diversificación para abordar dicho problema.

G. Diversificación

Siempre que no podamos explorar las otras partes interesantes (no explorar usando la intensificación) de un espacio de búsqueda dado, podemos explorarlo usando una estrategia de diversificación. Se utiliza en la memoria a largo plazo, que es la memoria de frecuencia. Diversificar significa ampliar la búsqueda hacia una nueva dirección. La estrategia de diversificación puede registrar todas las iteraciones desde el estado inicial de la búsqueda que proporciona la solución actual. También se realizan movimientos cuando sea necesario para obtener la mejor solución. Es aplicable en la búsqueda tabú cuando se diseña un tema crítico. También se trata del paso inicial y debería realizar más iteraciones si los resultados obtenidos no están a la altura de las expectativas. Esta estrategia demostró la dirección de la nueva región del espacio de búsqueda. En TSA, la intensificación y la diversificación van de la mano. Si parte del espacio está restringido a un espacio de búsqueda y también es difícil guiar la búsqueda a diferentes regiones de búsqueda, estos casos en su mayoría reinician la TSA con otras soluciones iniciales en lugar de elegir una solución inicial aleatoria. La diversificación se utiliza más eficientemente cuando es simétrica. Algunos mecanismos que se utilizan para implementar estrategias de intensificación y diversificación son: oscilación estratégica, soluciones evaluadas pero no visitadas, reenlace de caminos, refuerzo por restricción, creación de nuevos atributos, etc.

H. Mantenimiento Preventivo (PM)

Se requiere PM para aumentar la disponibilidad del equipo y también mejorar la eficiencia general. Por tanto, la gestión del mantenimiento asume un papel activo en la fabricación del equipo. PM decide qué tareas hacer y cuándo, por lo que se puede clasificar tanto en programación como en asignación de recursos. Por McCall (1965), Pierskalla & Voelker (1976) mencionan dos aspectos, el primero dicotómico y el segundo aspectos proactivos. En dicotomía, permite que el equipo falle estocásticamente y descubre que el estado real se conoce o no en función del estado real. Proactivo, se ocupa de la vida útil de los equipos para que ambos puedan trabajar con probabilidad de ocurrencia de fallas y averías. PM_TABU, programación de la estructura de la memoria, como la búsqueda de información almacenada más allá del óptimo local. PM_TABU, el algoritmo brinda una muy buena solución al problema de programación de PM mediante varios estados, como la solución inicial, la función objetivo, la intensificación probabilística y la diversificación, son movimientos para TS, vecindad, tabú, mueven diferentes atributos y el

longitud de la lista de candidatos (tabu_list), nivel de aspiración, regla de detención, descripción del TS básico y descripción de PM_TABU.

I. Vecindad En la

búsqueda heurística la vecindad es el aspecto más importante. El vecindario se puede aplicar según los requisitos del problema dado. Se define aplicando la función de movimiento; la función de movimiento se realiza reemplazando o intercambiando la lista de candidatos que proporciona la mejor solución. Según el mundo matemático, se define como mover el punto que es un conjunto de distancias sin salir del conjunto. Se demostró cuando las nuevas soluciones necesitan intercambiar el nodo o borde por otro nodo o borde. La solución de vecindario se realiza intercambiando y reemplazando. Tenemos varias rutas, pero la cantidad de nodos sigue siendo la misma. En la heurística, el vecindario realiza el mejor movimiento para intercambiar nodos. Sus conceptos son diferentes en la búsqueda tabú de otros algoritmos de búsqueda local. Aquí es de naturaleza dinámica porque el movimiento vecinal depende de si el vecindario es constructivo o destructivo. En constructivo, la solución y el resultado son un proceso constructivo, entonces se llama vecindad constructiva y si la solución o el resultado toman un proceso destructivo, entonces se conoce como vecindad destructiva. En TS se utiliza la naturaleza dinámica para evitar el ciclo en el espacio de búsqueda. Aquí usamos la memoria a corto plazo para almacenar y eliminar la solución visitada, pero la memoria a largo plazo se usa para expandir el vecindario y también para examinar las regiones no visitadas.

III. ALGORITMO DE BÚSQUEDA TABÚ
Algoritmo de búsqueda tabú
1. comenzar TS 2. TS_list = []; 3. S = solución inicial; 4. S* = S; 5. Repita del paso 2 al paso 4 6. Encuentre la mejor solución admisible S1 pertenece al vecindario de S 7. si f(S1) > f(S*) entonces S*:= S1; 8. S:= S1; 9. Actualizar la lista TS TS_list; 10. Hasta el proceso, criterio de detención; 11. Fin;

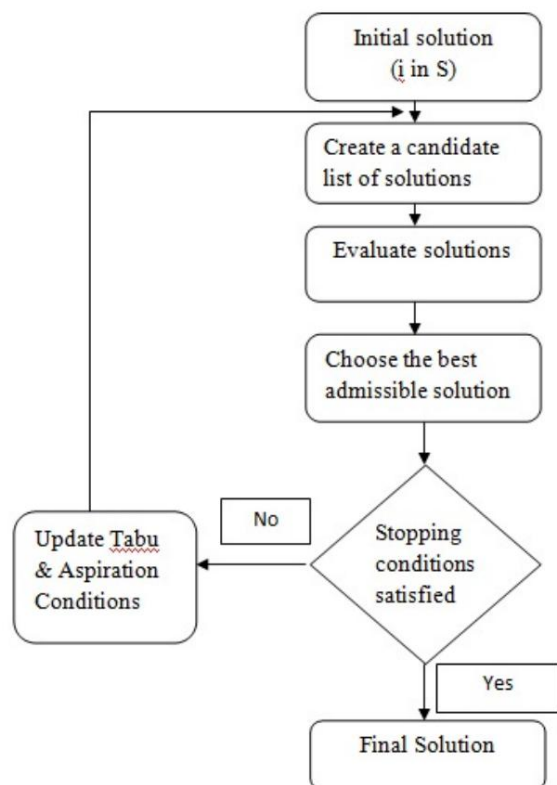


Figura 3.1. Diagrama de flujo de la TSA

IV. APLICACIÓN DEL ALGORITMO DE BÚSQUEDA TABU

En TSA tenemos varias aplicaciones, algunas de ellas el problema del viajante (TSP), el problema del árbol K, VRP, OVRP, el problema de programación y enrutamiento de viajes múltiples (MTVRSP), el problema de carga de contenedores (CLP), la programación preventiva (PM) y el taller de trabajo. problema (JSP), etc.

A. El problema del viajante TSP es el

problema más famoso en matemáticas computacionales para la investigación de optimización, porque es el más popular y simple de explicar y comprender, pero no es difícil encontrar una posible solución óptima. TSP es un tipo de problema NP-difícil, por lo que es muy difícil resolver todas las instancias posibles del problema del viajante para encontrar la optimización en un tiempo mínimo de ejecución.

TSP se define como calcular la ruta de costo óptima en un árbol o gráfico (gráfico no dirigido) desde una fuente o nodo inicial y cubrir todos los nodos en el gráfico o árbol exactamente una vez (excepto el nodo inicial) y luego detenerse en el punto inicial. nodo. Puede ser asimétrico o simétrico. Primero, estrategia asimétrica, el costo del borde depende de la dirección del viaje, por ejemplo, el costo de viajar de la ciudad X a la ciudad Y puede ser igual o puede ser diferente del costo de viajar de la ciudad Y a la ciudad X. Segunda estrategia simétrica El costo de un enlace o borde es independiente de la dirección del viaje, como el costo de viajar de la ciudad X a la ciudad Y es el mismo cuando se viaja de la ciudad Y a la ciudad X debido a la independencia de la dirección en esta estrategia. Aquí hay un ejemplo de muestra, que es solo

considerando la estrategia simétrica Problemas de los viajeros.

Mostrando la siguiente Figura 4.1 que muestra un ejemplo de gráfico TSP con diez nodos. Este gráfico es un gráfico simétrico de TSP. En el gráfico que muestra los números en los enlaces están los pesos de los bordes que están conectados de un nodo a otro nodo.

Solución inicial EI

Gráfico 4.1 muestra la solución inicial donde el nodo inicial 1 y aplicando el TSP, el proceso de viaje comienza con el nodo 1 y luego se visita solo una vez para cada nodo excepto el nodo 1. El proceso de visita funciona de tal manera que existe una solución óptima. El costo encuentra la solución sin ningún ciclo (excepto el nodo inicial). Aplicando cualquier búsqueda local, podemos encontrar el camino más largo con un coste óptimo. Después de eso, aplique TSA para encontrar la solución óptima y descubra que proporciona la mejor solución óptima, como se muestra en la figura 4.3. Ahora, en el siguiente ejemplo, utilizamos el primer algoritmo codicioso para encontrar el camino más largo recorrido para obtener la solución inicial. El efecto de encontrar la solución óptima para TSP aumenta como se muestra en la figura 4.2.

La solución inicial para TSP simétrico basada en la

solución de cola heurística es 81, usaremos como solución inicial para la entrada de búsqueda tabú (TSA).

Búsqueda tabú

En el borde inicial (1, 2) mecanismo de movimiento de intercambio

Al principio en TSA, tabu_list está vacío. Se seleccionan dos bordes al azar y los bordes seleccionados no habrá restricción para intercambiarlos. En la etapa inicial, los movimientos se realizan sin restricciones y se selecciona la mejor lista de candidatos moviéndose entre sí.

Después de eso, se agregan e intercambian dos nuevos bordes moviendo la mejor solución posible con alguna restricción. Aquí se intercambian los mejores tabu_list, que son el nodo (1, 2) y el nodo (4, 5), y estos dos bordes se eliminan y se agregan los dos mejores bordes nuevos, que son (1, 4) y (2, 5). Después de eliminar dos bordes, la nueva longitud del recorrido es 77, como se muestra en la figura 4.3.

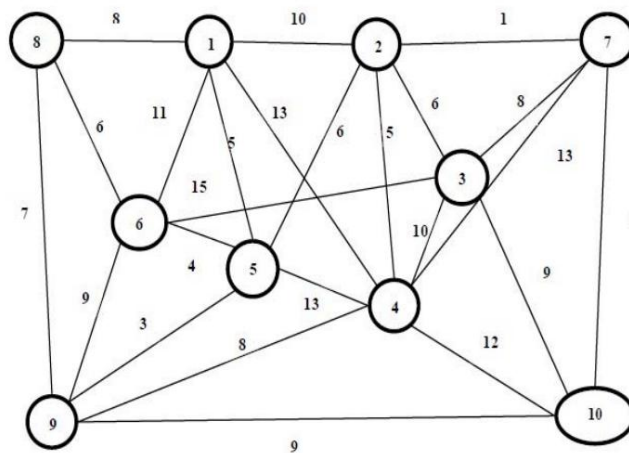


Figura 4.1. Gráfico del problema del vendedor ambulante

Tour length 81

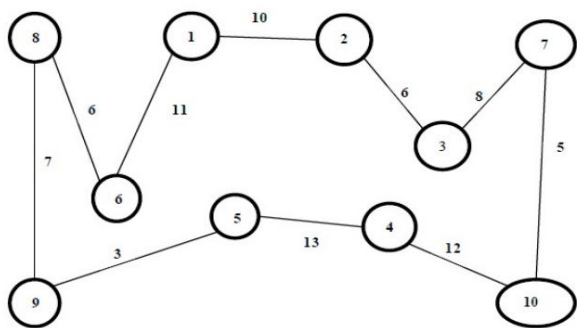


Figura 4.2. Solución por algoritmo de codicia.

Tour Length 78

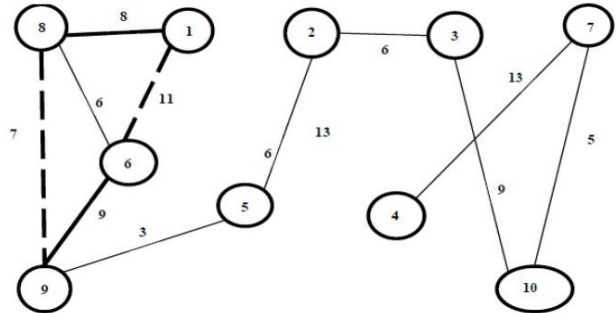


Figura 4.5. Tercera iteración de la búsqueda tabú

Tour length 77

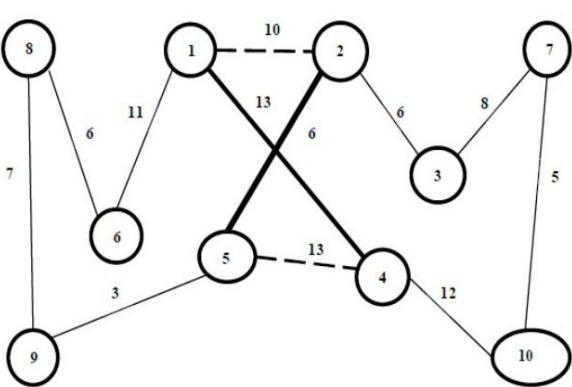


Figura 4.3. Primera iteración de la búsqueda tabú

En la siguiente iteración, se aplican otros dos bordes con el mismo mecanismo si la solución actual y la nueva solución funcionan o no funcionan mejor que los bordes actuales que no se intercambian. Pero algunas restricciones dan una mejor solución, entonces la mejor solución se actualizará con una nueva solución.

Tour Length 79

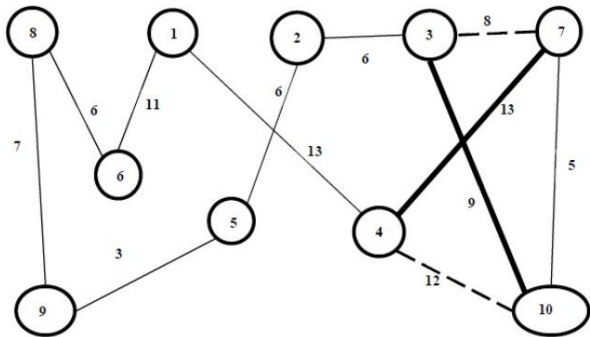


Figura 4.4. Segunda iteración de la búsqueda tabú

La iteración se repite siempre que se cumplan los criterios de parada. Cuando se cumplan los criterios de almacenamiento, corrija la iteración antes de que comience el proceso. En este método no hay un proceso de retroceso. En el ejemplo de la figura 4.4 y la figura 4.5, el TSP considera que la mejor solución es 77.

Muchos investigadores han resuelto las consecuencias del problema del viajante de TSP de varias maneras. Antes de aplicar la búsqueda tabú en TSP, proporciona un resultado de solución casi óptimo o un resultado óptimo, pero después de aplicar la metaheurística de TSA en TSP, siempre proporciona un resultado óptimo. El uso de TSA para resolver TSP puede generar varios o diferentes resultados que han sido realizados por varios artículos de investigación, es decir, [3], [7], [8] y [14], una encuesta comparativa basada en TS, TSP produce mejores resultados. Produjo mejores resultados y una solución eficaz para grandes problemas de prueba de TSP y la calidad depende de la duración de cada búsqueda y de la duración de cada búsqueda. Se han aplicado varios métodos nuevos en TSP mientras se usa TS, algunos de ellos son TS basados en ángulos, búsqueda tabú adaptativa paralela, búsqueda tabú multipunto [11] [12] [13]. Basado en TSA, TSP se ha resuelto VRP, OVRP y algunos investigadores como José Brando et al. y todos en 1997 pueden mejorar la solución mediante el uso de rutas y programación de vehículos de viajes múltiples en metaheurística de búsqueda tabú.

TABLA 1

Varias descripciones breves de problemas basadas en TS

S. No	Nombre de Problemas	mecánico sm	Función y memoria	Solución / trabajo futuro Mejor
1.	Problema de rutas de vehículos (1999) entre diferentes Por Gulary Barbaros oglu et.	Intercambio de vértices ruta, euclidiana distancia, dinámico mover,	simpático, Criterio de aspiración, diversificación. n, paralelo informática, corto plazo memoria.	solución admisible, Solicitud de distribución empresarial.
2.	vehículo abierto	Muévete, entrenas	k- árbol, NNH, MST,	Muy buena solución, puede

	problema de enrutamiento (2004) Por: Gestao y Administrar racao publica et.	intercambio, límite inferior, más bajo tenencia dar mejor solución	influencia, simétrica y asimétrica, procesamiento paralelo	o puede que no sea una solución admisible, la tenencia aleatoria ofrece una mejor solución que la tenencia fija	mi mantener nce programación g (2001) Por M. Gopalakrishnan et al.	basado en heurística tabú, diagrama de flujo, superior encuadrado, todos los criterios de búsqueda	agregar combinaciones de artesanías que caen e intercambian, memoria mecanismos, heurística local, PM, TABU	estudio, heurística local y brecha de optimidad promedio se puede reducir
3.	Rutas y programación de vehículos para viajes múltiples problema g (1997) Por Gestao e administrar racao publica et.	Intercambio mover consiste en intercambiar las rutas diferentes,	holgura hacia adelante, holgura negativa, dos aproximaciones: órdenes constructivos secuenciales y sin ruta se inserta una de las rutas existentes	Factor aleatorio, el algoritmo brinda una mejor solución en comparación con la solución manual, mejora la disponibilidad del vehículo	7. Problema del taller de trabajo (2005) Por: Eugenio z y todos	Problema de taller de trabajo, hacer tramo, reenlace de camino, vecino barrio	programación de trabajos en PC, memoria dinámica, orden topológico de programación avanzada basada en tabú, cabeza y cola, parámetros de ajuste,	i-TSAB y relinking proporcionan una herramienta muy poderosa para resolver problemas de taller, muy buena Precisión y tiempo rápido de obtención en
4.	contener r problema de carga (2002) Por A. Bortfeldt y todos	Heurístico, local y óptimos globales, o intercambio,	Algoritmo TSA paralelo, logra utilización del valor medio, soporte mínimo, LIFO, intercambio local y global, espacio de paquete, memoria clasificar de términos, etc.	Tabú paralelo más potente como búsqueda tabú, mejor rendimiento, multi utilizar hilos de búsqueda, se puede resolver WLP, hay trabajo son futuro con paralelización híbrida n enfoque para metaheurística.				
5.	Óptimo Colocadores de PMG t (2005) Por: jiangnan peng y et al.	Mover, inicial UGP colocadores t, conjunto de candidatos mover, aspiración n criterios, criterios de parada,	UGP esquema de colocación óptima, matriz de incidencia, simulación por Autobús IEEE 14, sistema de bus New England 39, sistema de bus IEEE 57,	TS proporciona alta precisión y menos cálculo l esfuerzo, la matriz de incidencia manipula los números enteros y rápidamente observa la red de manera conveniente				
6.	VRP preventivo		Comparativo de tres movimientos					

V. CONCLUSIÓN

El algoritmo de búsqueda tabú (TSA) funciona bien y proporciona resultados óptimos. Se analizan varios problemas diferentes basados en TS. Proporciona la solución básica y basada en problemas para la búsqueda tabú. Cómo se puede aplicar y dónde se puede aplicar. El alumno también puede rastrear y encontrar el concepto de TS y aplicar TS en varios problemas locales para encontrar la mejor solución admisible. Ayuda a los investigadores que trabajan con optimización combinatoria y TSP, TSA, JOB SHOP PROBLEM, VRP, etc.

VI. TRABAJO FUTURO

En el futuro, TS puede aplicarse en varias áreas, como Internet de las cosas y aprendizaje profundo, y puede simularse y crear los mejores enfoques híbridos para resolver la mayoría de los problemas de optimización de admisión.

REFERENCIAS

- [1] Glover Fred, Búsqueda Tabú - Parte I, Revista ORSA sobre Computación, Volumen 1, Número 3, págs. 190-206, 1989
- [2] Glover Fred, Búsqueda tabú - Parte II, ORSA Journal on Computing, Volumen 2, Número 1, págs. 4-32, 1990.
- [3] Yi He, Yuhui Oiu, Guangyuan Liu, Kaiyou Lei, Un enfoque de búsqueda tabú adaptativo paralelo para el problema del viajante de comercio, Procesamiento del lenguaje natural e ingeniería del conocimiento, 2005, IEEE NLP-KE 2005, Conferencia internacional IEEE, págs. 796 – 801 .
- [4] Meeran, S y Morshed, MS, Un algoritmo de búsqueda tabú genético híbrido para resolver problemas de programación de talleres, Journal of Intelligent Manufacturing, volumen 23, número 4, págs. 1063 – 1078, 2012.
- [5] Gouhui Zhang, Yang Shi, Liang Gao, Un algoritmo genético y búsqueda tabú para resolver horarios flexibles de trabajo, inteligencia y diseño computacional, 2008, ISCID 2008, Simposio internacional sobre (Volumen: 1), págs. 369 – 372 [6] Yajie Tian, Sannomiya N., Yuedong Xu, Una búsqueda tabú con una nueva técnica de búsqueda de vecindario aplicada a problemas de programación de talleres de flujo, Decisión y control, 2000, 39ª Conferencia IEEE (Volumen: 5), págs. 4606 – 4611.

- [7] Laporte G. El problema de las rutas de vehículos: una descripción general de los algoritmos exactos y aproximados. *Revista europea de investigación operativa* 1992;59:45-58.
- [8] Fisher ML. Solución óptima de problemas de rutas de vehículos y árboles k mínimos, Informe 89-12-13, Departamento de Ciencias de la Decisión, The Wharton School, Filadelfia, PA, 1989.
- [9] H. Gehring, A. Bortfeldt, Un algoritmo genético para resolver el problema de carga de contenedores, *Transacciones internacionales en investigación operativa* 4 (1997) 401–418.
- [10] R. Morabito, M. Arenales, Un enfoque de gráfico AND/OR para el problema de carga de contenedores, *International Transactions in Operational Research* 1 (1994) 59–73.
- [11] BKA Ngoi, ML Tay, ES Chua, Aplicación de técnicas de representación espacial al problema del embalaje de contenedores, *Revista Internacional de Investigación de Producción* 32 (1994) 111–123.
- [12] D. Pisinger, Heurística para el problema de carga de contenedores, *European Journal of Operational Research* 141 (2002) 143-153.
- [13] Y. Rochat, E. Taillard, Diversificación e intensificación probabilística en la búsqueda local de rutas de vehículos, *Journal of Heuristics* 1 (1995) 147–167.
- [14] G. Scheithauer, Algoritmos para el problema de carga de contenedores, en: *Operations Research Proceedings*, 1991, Springer, Berlín, 1992, págs.
- [15] M. Gendreau, A. Hertz y G. Laporte, Nuevos procedimientos de inserción y postoptimización para el problema del viajante, CRT-708 Centre de Recherche sur les Transports, Université de Montreal, Montreal, Canadá (1991).
- [16] C.-N. Fiechter* Un algoritmo de búsqueda tabú paralelo para problemas de grandes viajeros (1994) 243-267.
- [17] JoseBrandao, Alan Mercer, Un algoritmo de búsqueda tabú para el problema de programación y rutas de vehículos de viajes múltiples, Elsevier en *European Journal of Operation Research* 100(1997) 180-191.
- [18] Eugeniusz Nowicki y Cxslaw smutnicki, Un algoritmo de búsqueda tabú avanzado para el problema del taller de trabajo, Springer Business media in the Neatherland, 2005, *Journal of Scheduling* 8: 145-159.
- [19] M. Gopalakrisnan, S. Mohan, Z. He , Una heurística de búsqueda tabú para la programación del mantenimiento preventivo, Elsevier, 2001, informática e ingeniería industrial, 149-160.
- [20] Jiangnan Peng, Yuanzhang Sun, HF Wang, Ubicación óptima de la PMU para una observabilidad completa de la red utilizando el algoritmo de búsqueda tabú, Elsevier, 2006, *Electricla power energy system* 223-231.
- [21] A. Bortfeldt, H. Gehring, D. Mack, Un algoritmo de búsqueda tabú paralelo para resolver el problema de carga de contenedores, Elsevier, 2003, computación paralela 641-662.
- [22] Gulay Barbarosoglu, Demet Ozgur, Una búsqueda tabú del vehículo problema de enrutamiento, Elsevier, 1999, computadoras e investigación de operaciones 255-270.
- [23] José Brandao, Un algoritmo de búsqueda tabú para el problema de rutas abiertas de vehículos, Elsevier, 2004, en *European Journal of Operation Research* 552-564.