Investigación de Operaciones en Acción: Heurísticas para la Solución del TSP

Roger Z. Ríos Mercado
Programa de Posgrado en Ingeniería de Sistemas
Universidad Autónoma de Nuevo León

José Luis González Velarde
Centro de Sistemas Integrados de Manufactura
ITESM – Campus Monterrey

Resumen

Uno de los problemas más famosos y difíciles en la teoría de optimización, es del problema del agente viajero (TSP). El interés en el estudio de técnicas para su solución es motivado por la enorme cantidad de aplicaciones prácticas de problemas de toma de decisiones donde éste aparece como subestructura. En este artículo se hace una breve reseña de los métodos de aproximación (heurísticas) más relevantes que se han propuesto para intentar encontrar soluciones factibles de alta calidad.

Palabras clave: Investigación de Operaciones, Problema del Agente Viajero, Heurística, Metaheurística

1. Introducción

En [1] introdujimos al lector con el Problema del Agente Viajero (mejor conocido por TSP, por sus siglas en inglés; Traveling Salesperson Problem), el cual es un problema clásico de optimización combinatoria, una de las subdisciplinas de la investigación de operaciones (IO). Señalamos cómo las aplicaciones de IO se encuentran en prácticamente todos los niveles y en todo tipo de industrias, y cómo una utilización adecuada de las técnicas de IO dándole soporte al complejo proceso de toma de decisiones que enfrentan las empresas, puede tener un impacto económico significativo.

En particular, ilustramos la importancia del TSP con un par de problemas reales: el problema de programación de tareas que se presenta en la manufactura de bienes y el del ruteo de vehículos en el ramo de la logística.

Como una de las características del TSP es el de pertenecer a una clase de problemas muy difíciles de resolver, es decir, hallar la solucion óptima, en la práctica es muy común el utilizar algoritmos de aproximación (heurísticas) para obtener soluciones factibles de alta calidad (relativamente cercanas al óptimo) en tiempos de ejecución relativamente pequeños. En este artículo, a manera de continuación, exponemos algunas de las heurísticas más utilizadas para intentar obtener soluciones al TSP.

2. Qué es el TSP

Retomando la definición efectuada en [1], el TSP se formula de la siguiente manera. Un agente viajero, partiendo de su ciudad de origen, debe visitar exactamente una vez cada ciudad de un conjunto de ellas (previamente especificado) y retornar al punto de partida. Un recorrido con estas características, es llamado dentro de este contexto un *tour*. El problema consiste en encontrar el tour para el cual la distancia total recorrida sea mínima. Se asume que se conoce, para cada par de ciudades, la distancia entre ellas. La Figura 1 ilustra un tour en una instancia de ocho ciudades, representada por un grafo donde cada nodo del grafo corresponde a una ciudad y cada arista que une a un par de nodos representa la parte del tour que pasa por dichos nodos. En la figura se ilustra el tour que visita las ciudades 1, 2, 3, 8, 5, 4, 7, 6 y 1, en ese orden.

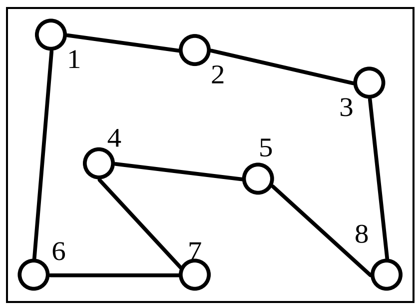

Figura 1: Un tour en un TSP de ocho ciudades

El problema en sí es fácil de formular. Sin embargo, al igual que muchos otros que se presentan en el campo de optimización, es sumamente difícil de resolver (por resolver, nos referimos a encontrar la solución óptima al problema y probar desde luego que ésta es efectivamente la mejor solución posible). En [1] establecimos con más detalle cuándo un problema es “fácil” o “difícil”. La implicación directa de un problema difícil de resolver es que cualquier algoritmo empleado para encontrar la solución óptima emplea un tiempo de cómputo que crece exponencialmente con el tamaño de los datos del problema. Por tal motivo, nace la necesidad de emplear heurísticas, las cuales son procedimientos que aún que no garantizan una solución óptima al problema, obtienen soluciones factibles de alta calidad (relativamente cercanas al óptimo) en un tiempo de ejecución razonable.

3. Algoritmos para la solución del TSP

Heurísticas de Propósito Especial

Empezaremos describiendo algunas heurísticas de propósito especial que han sido propuestas para resolver el TSP. Se llaman de propósito especial, porque explotan la estructura y características particulares de cada problema.

La primera familia de esta clase de heurísticas que describiremos pertencen a las heurísticas de tipo miope (***greedy*** en inglés), son llamadas así porque sólo se preocupan por hacer lo mejor que pueden localmente, sin ver más allá de un cierto entorno muy cercano.

(a) *El vecino más cercano*: Se trata de un procedimiento constructivo, se parte de elegir un vértice inicial, llamémoslo j_1. Una vez seleccionado, mediremos la distancia que hay de este vértice a los restantes, y elegiremos ahora aquél cuya distancia al vértice inicial sea la mínima (es decir elegimos al vecino más cercano), y lo llamaremos j_2. De la misma forma, construiremos una trayectoria $j_1, j_2, j_3,\ldots j_k, j_{k+1}, \ldots, j_n$, donde el vértice j_{k+1} se elige tomando la mínima distancia que hay desde j_k hasta cada uno de los vértices que sean distintos de los ya elegidos $j_1, j_2, j_3, \quad j_k$. Al terminar, se debe de agregar el arco que va del vértice j_n., hasta el vértice j_1. Con esto habremos completado el tour. Esta heurística tiene una ventaja en las primeras selecciones, sin embargo, el problema que presenta es que en los últimos pasos puede elegir aristas de longitud muy grande, especialmente en la última.

(b) *La inserción más cercana:* Este procedimiento es también constructivo, pero en contraste con el anterior, en el cual se tiene un camino, y sólo al final se completa un tour, aquí tenemos subtours, los cuales van creciendo hasta completar un tour que abarque todos los vértices. Iniciemos con un subtour, al cual llamaremos T, queremos ahora insertar el nodo "más cercano" a este subtour para ampliarlo. Así que examinemos primero todos los nodos j que no estén aún incluidos en T.y vamos a definir para estos nodos, su distancia a T de la siguiente manera: $d(j,T)$ es la distancia mínima que hay desde el nodo j a cualquiera de los nodos que pertenecen a T. Ordenamos las distancias calculadas de menor a mayor, y llamemos j^* al nodo que se encuentra al principio de esta lista, este será el nodo "más cercano" a T. Vamos ahora a seleccionar dentro de T al nodo que se encuentre "más cerca" de j^*, esto es, medimos la distancia desde j^* a cada uno de los nodos de T, y llamaremos k^* aquel nodo dentro de T, cuya distancia a j^* sea la menor de todas. Ampliaremos ahora el subtour insertando a j^* entre k^* y alguno de sus dos vecinos en T, esto es, si (k_1, k^*) y (k^*, k_2) son dos aristas de T y la distancia de j^* a k_1, es menor o igual que la distancia de j^* a k_2, entonces j^* se inserta entre k_1 y k^*. El proceso terminará cuando se haya construido un tour completo. Como en el caso anterior, no se puede garantizar que se produzca una buena solución.

Garantías de Desempeño.

Hemos mencionado que no es posible garantizar que los dos métodos anteriormente descritos produzcan buenas soluciones. ¿Será posible encontrar otros métodos heurísticos con los cuales sí se pueda garantizar un buen desempeño del método? Para contestar esta pregunta, definiremos en primer lugar qué es lo que se entiende por una garantía del desempeño. Dado un ejemplo particular de un problema al cual denotaremos por I (de aquí en adelante llamaremos *instancia* a un caso particular de un problema), A(I) será el valor producido por el algoritmo de aproximación que estemos usando, mientras que OPT(I) es el valor de la solución óptima, como en el TSP, lo que queremos encontrar es el tour de menor longitud, siempre tendremos que OPT(I) debe de ser menor o igual que la longitud de cualquier otro tour, así que OPT(I) ≤.A(I). Diremos que el algoritmo de aproximación tiene una garantía de comportamiento c^* , donde c^* es un número real, si para cualquier instancia del problema I, se puede probar que A(I) ≤. c^* OPT(I). ¿Qué es lo que indica este número c^*? Observemos primero que si su valor es 1, el algoritmo de aproximación siempre producirá la solución óptima, ya que combinando las dos desigualdades, se tiene que OPT(I) = A(I). Por otra parte, su valor no puede ser menor que 1, ya que en ese caso, se tendría que A(I) < OPT(I), es decir el algoritmo de aproximación produciría un valor menor que el óptimo, lo cual es imposible. Así que c^* tiene que ser un valor mayor o igual que 1. Ahora bien, mientras más cerca se encuentre de 1 este valor, tenemos que el algoritmo de aproximación, obtendrá soluciones que no se encuentran muy lejos del valor óptimo, y si este valor es muy grande, esto indica que se pueden producir soluciones muy alejadas del valor óptimo.

En el caso del TSP es posible encontrar garantías de desempeño, pero con la condición de que las instancias examinadas posean una propiedad adicional: la *desigualdad del triángulo*. Esta desigualdad puede describirse de la siguiene manera: para viajar de una ciudad a otra es más corto hacerlo directamente que pasando por una ciudad intermedia. Más formalmente, se tiene que cumplir que la distancia de i a j para cualquier par de nodos debe de ser menor o igual que la distancia de i a k más la distancia de k a j, para cualquier otro nodo k. Si esto es cierto, enotnces sí es posible dar una garantía de desempeño. Por ejemplo para la heurística del vecino más cercano. Denotando por NN(I) al valor producido, es posible demostrar que para cualquier instancia I con m ciudades

$$\mathrm{NN}(I) \leq \tfrac{1}{2} (\log_2 m + 1)\ \mathrm{OPT}(I)$$

Pero, por otro lado, para valores arbitrariamente grandes de m, siempre será posible construir instancias con m ciudades, para las cuales

$$\mathrm{NN}(I) > 1/3\ (\log_2 (m + 1) + 4/3)\ \mathrm{OPT}(I)$$

Lo que estos resultados nos indican es entonces que la heurística deja mucho que desear, ya que en la primera desigualdad no tenemos un valor constante que nos dé una garantía para todas las instancias, así que en todo caso podríamos decir que $c^* = \infty$. Por otra parte, la segunda desigualdad nos asegura, que de hecho, encontraremos instancias para las cuales la heurística produce valores muy alejados del óptimo.

¿Sé podrán construir otras heurísticas con una mejor garantía de desempeño? Afortunadamente sí es posible, describirlas aquí, sin embargo, nos llevaría mucho más allá de los alcances de este artículo, la mejor de todas ellas se debe a Christofides [2] y combina la solución de varios problemas de la teoría de grafos para construir un tour, con una garantía de desempeño de $c^* = 3/2$. La implicación de este valor es que, cualquier solución que construyamos con este esquema de aproximación, nos dará un valor que nunca excederá en 50 % al valor de la solución óptima. Algo muy interesante es que si eliminamos esta propiedad de la desigualdad del triángulo, es imposible construir algoritmos de aproximación cuyo comportamiento sea polinomial que tengan una garantía de desempeño, por mala que esta sea. Es decir, si tal construcción fuese posible, entonces podríamos también construir un algoritmo polinomial que resuelve en forma exacta el TSP.

Metaheurísticas

Las metaheurísticas son una clase de métodos de aproximación, que se diseñan para atacar problemas difíciles para los cuales las heurísticas de propósito especial han fracasado en dar resultados efectivos y eficientes. Las metaheurísticas proporcionan marcos generales que permiten crear nuevos híbridos combinando diferentes conceptos derivados de las heurísticas clásicas, la inteligencia artificial, la evolución biológica, los sistemas neuronales, la mecánica estadística y el psicoanálisis freudiano. Estas familias de enfoques incluyen, pero no están limitadas a algoritmos genéticos, GRASP, redes neuronales, búsqueda tabú y recocido simulado. En [3] pueden encontrarse unos excelentes tutoriales de cada una de estas metaheurísticas.

El método metaheurístico que emplearemos aquí, Búsqueda Tabú, fue propuesto por Fred Glover [4] en 1986, y está basado en el psicoanálisis freudiano.

Iniciaremos describiendo qué es un método de búsqueda local. Se trata de un método iterativo el cual da inicio desde una solución arbitraria, el procedimiento consiste en explorar una vecindad previamente definida para cada punto del espacio de soluciones y elige una nueva solución dentro de tal vecindad, la cual mejora el valor que se tiene a mano. La búsqueda termina cuando se alcanza una solución tal que es la mejor dentro de la vecindad predefinida, esto es ya no puede seguirse mejorando. A esta solución se le llama un mínimo local. En muchas ocasiones, este mínimo local será la solución óptima del problema, sin embargo, no podemos esperar que siempre suceda esto. Al contrario es plausible esperar que este mínimo local se encuentre lejos de la solución óptima del problema.

En el caso particular del TSP, un método de búsqueda local sencilla, es el llamado 2-opt. Este consiste en eliminar del tour un par de aristas que no sean adyacentes, y reemplazarlas con el único par de aristas con el cual se puede formar nuevamente un tour. Éste se ilustra en la Figura 2.

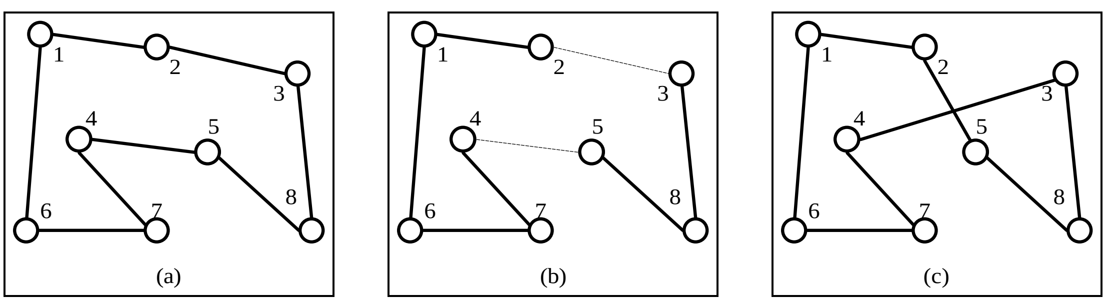

Figura 2: (a) Solución inicial; (b) Eliminación de dos aristas: (2,3) y (5,4);
(c) Nuevo tour sustituyendo con las aristas (2,5) y (3,4)

TS guía un procedimiento de búsqueda local para continuar más allá de óptimos locales, esto es al no poder seguir mejorando la solución, se permite tomar otra solución aún cuando el valor no mejore, sino que se degrade, esto permite salir del óptimo local encontrado, pero al mismo tiempo se corre el peligro de caer en un ciclo, de mejorar-empeorar la solución, para evitar esto, se emplea una estrategia que modifica las vecindades a medida que la búsqueda avanza. TS utiliza estructuras de memoria para determinar esta vecindad modificada, las soluciones permitidas se determinan identificando soluciones encontradas dentro de un horizonte especificado. En nuestro ejemplo, dada una solución particular, una vez suprimido un par de aristas del tour, estas dos aristas no pueden formar parte del tour por un determinado número de iteraciones, este número de iteraciones se conoce como la *permanencia tabú*. Simétricamente cuando un par de aristas se insertan en un tour, no podrán ser suprimidas durante un número de iteraciones. Si la permanencia tabú se elige de manera adecuada, la búsqueda podrá continuar más allá de los óptimos locales sin caer en ciclos, y eventualmente alcanzar, si no el óptimo global del problema, sí soluciones que estén cerca de él.

4. Conclusión

En este artículo hemos mostrado al lector algunos de los algoritmos más notables y populares para encontrar soluciones aproximadas al problema clásico del agente viajero. Esta clase de métodos tienen la ventaja que son relativamente rápidos en sus tiempos de ejecución y por ende, se convierten en una elección importante cuando el tiempo para encontrar una solución es la restricción más importante. Sin embargo, cuando la precisión y calidad de una solución cobra mayor importancia, existe otro tipo de algoritmos basados en técnicas de enumeración implícita, los cuales intentan, a costa de un mayor esfuerzo computacional, encontrar soluciones óptimas globales al problema. Ese será tema para un próximo trabajo.

Referencias

[1] J. L. González Velarde y R. Z. Ríos Mercado. Investigación de operaciones en acción: Aplicación del TSP en problemas de manufactura y logística. *Ingenierías*,2(4):18-23, 1999.

[2] N. Christofides. Worst-case analysis of a new heuristic for the traveling salesman problem. Report 388, Graduate School of Industrial Administration, Carnegie-Mellon University, Pittsburgh, EUA, 1976.

[3] A. Díaz (editor). *Optimización Heurística y Redes Neuronales*. Editorial Paraninfo, Madrid, España, 1996.

[4] F. Glover. Future paths for integer programming and links to artificial intelligence. *Computers and Operations Research*, 1:533-549, 1986.

Fichas Biográficas

El Dr. Ríos Mercado labora actualmente como Profesor de Tiempo Completo y Exclusivo en el Programa de Posgrado en Ingeniería de Sistemas (PISIS) de la FIME de la UANL. Recibió sus títulos de Doctor (Ph.D.) y Maestro en Ciencias (M.S.E.) en Investigación de Operaciones e Ingeniería Industrial de la Universidad de Texas en Austin, y su título de Lic. en Matemáticas de la UANL. Sus intereses de investigación son programación matemática, desarrollo de heurísticas y optimización estocástica, con aplicación a problemas de optimización y toma de decisiones provenientes de la industria del gas y procesos de manufactura. Más sobre su trabajo puede encontrarse en el sitio del PISIS: *http://yalma.fime.uanl.mx/~pisis/*

El Dr. González Velarde labora actualmente como Profesor Titular en el Centro de Sistemas de Manufactura del Campus Monterrey del ITESM. Es Doctor (Ph.D.) en Investigación de Operaciones e Ingeniería Industrial por la Universidad de Texas en Austin, Maestro en Ciencias (M.Sc.) por la Universidad de California en Berkeley, y Licenciado en Matemáticas del Campus Monterrey del ITESM. Sus intereses principales se enfocan a la interfase de la Investigación de Operaciones con las Ciencias Computacionales, el diseño de heurísticas para la Optimización Discreta, con aplicaciones hacia el área de transporte y manufactura, y la Optimización Robusta.