

Traveling Saleman Problem

Avances de proyecto

Edgar Osvaldo López Zúñiga

`edgar.lopez@cimat.mx`

June 1, 2022

Outline

Introducción

Descripción del problema

Variantes

Representación

Representación como
problema de
permutaciones

Soluciones

Complejidad

Búsqueda Local

Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

1 Introducción

- Descripción del problema
- Variantes

2 Representación

- Representación como problema de permutaciones

3 Soluciones

- Complejidad
- Búsqueda Local
- Algoritmo Genético

4 Avances

- Algoritmo Genético

5 Más avances e ideas

Descripción del problema

Introducción

Descripción del problema

Variantes

Representación

Representación como
problema de
permutaciones

Soluciones

Complejidad

Búsqueda Local

Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

Sea $G = (V, E)$ un grafo y \mathbb{F} la familia de todos los ciclos Hamiltonianos en G . Para cada arista $e \in E$ se asigna un costo c_e .

El TSP consiste en encontrar un recorrido de los nodos de G tal que la suma de los costos de las aristas del recorrido sea la menor posible.

Descripción del problema

Introducción

Descripción del problema

Variantes

Representación

Representación como
problema de
permutaciones

Soluciones

Complejidad
Búsqueda Local
Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

Sin perder generalidad se puede asumir que G es un grafo completo. En caso de que no lo sea, se puede reemplazar las aristas faltantes con una de alto costo en comparación con las demás.

Sea el conjunto de nodos $V = \{1, 2, \dots, n\}$. La matriz $C = [c_{ij}]_{n \times m}$ es llamada matriz de costos (distancias o pesos), donde la entrada c_{ij} corresponde al costo de la arista que une a los nodos i y j en G .

Variantes del TSP

Introducción

Descripción del problema

Variantes

Representación

Representación como
problema de
permutaciones

Soluciones

Complejidad
Búsqueda Local
Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

Dependiendo de la naturaleza de la matriz de costos, el TSP se puede clasificar en dos clases. Si C es simétrica, el TSP es llamado *Symmetric Traveling Salesman Problem*(STSP). Si C es no necesariamente simétrica, el problema es llamado *Asymmetric Traveling Saleman Problem* .

El STSP se puede ver como un caso especial del ATSP, aunque también es posible resolver el ATSP como un STSP duplicando el número de nodos.

Otras variantes

Introducción

Descripción del problema

Variantes

Representación

Representación como
problema de
permutaciones

Soluciones

Complejidad
Búsqueda Local
Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

- MAX TSP
- Bottleneck TSP
- TSP con múltiples visitas (TSPM)
- Clustered TSP
- TSP Generalizado (GTSP)
- m-salemen TSP

Problema de permutaciones

Introducción

Descripción del problema

Variantes

Representación

Representación como
problema de
permutaciones

Soluciones

Complejidad

Búsqueda Local

Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

El TSP puede ser visto como un problema de permutaciones. En general se utilizan dos variantes de representación en permutaciones del TSP; representación cuadrática y lineal.

Representación cuadrática

Introducción

Descripción del problema

Variantes

Representación

Representación como
problema de
permutaciones

Soluciones

Complejidad

Búsqueda Local

Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

Representación cuadrática del TSP

Sea \mathbb{P}_n la colección de todas las permutaciones en el conjunto $\{1, 2, \dots, n\}$. El TSP consiste en encontrar $\pi = (\pi(1), \pi(2), \dots, \pi(n))$ tal que el costo

$$c_{\pi(n)\pi(1)} + \sum_i = 1^n - 1 c_{\pi(i)\pi(i+1)}$$

se minimice.

En este caso, π representa el orden en que las ciudades son visitadas.

Representación cuadrática

Introducción

Descripción del problema

Variantes

Representación

Representación como
problema de
permutaciones

Soluciones

Complejidad

Búsqueda Local

Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

Para la representación cuadrática del TSP todos los desplazamientos cíclicos de π nos dan el mismo recorrido, por lo que hay n permutaciones que representan el mismo ciclo Hamiltoniano.

Representación lineal

Introducción

Descripción del problema

Variantes

Representación

Representación como
problema de
permutaciones

Soluciones

Complejidad

Búsqueda Local

Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

En esta representación, solo se consideran factibles permutaciones cíclicas.

Representación lineal del TSP

Sea \mathbb{C} la colección de todas las permutaciones cíclicas de $\{1, 2, \dots, n\}$. El TSP consiste en encontrar $\sigma = (\sigma(1), \sigma(2), \dots, \sigma(n)) \in \mathbb{C}$ tal que el costo

$$\sum_{i=1}^n c_{i\sigma(i)}$$

sea minimizado.

Bajo esta representación $\sigma(i)$ es el sucesor de i para $i = 1, 2, \dots, n$

Complejidad

Introducción

Descripción del problema

Variantes

Representación

Representación como
problema de
permutaciones

Soluciones

Complejidad

Búsqueda Local

Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

Número de recorridos posibles

Dado que el número de ciudades a visitar sea n , el número total de rutas posibles que cubren todas las ciudades puede verse como el número de soluciones factibles del TSP y es $\frac{(n-1)!}{2}$

Como se puede ver, al comenzar a trabajar con soluciones más grandes, se vuelve impráctica una solución por fuerza bruta probando todas las permutaciones.

Búsqueda local

Introducción

Descripción del problema

Variantes

Representación

Representación como
problema de
permutaciones

Soluciones

Complejidad

Búsqueda Local

Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

En un método de búsqueda local, la estructura del vecindario se introduce para generar movimientos de una solución a otra. Un óptimo local en este contexto es una solución que no puede ser mejorada usando la estructura de vecindario considerada.

k-Intercambio

Introducción

Descripción del problema

Variantes

Representación

Representación como
problema de
permutaciones

Soluciones

Complejidad

Búsqueda Local

Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

Los vecindarios fundamentales para el TSP se basan en diferentes procedimientos de intercambio de aristas e inserción de nodos. Un procedimiento clásico de este tipo es el k -intercambio, este procedimiento también forma el núcleo de otros procedimientos más avanzados.

k-Intercambio

Introducción

Descripción del problema

Variantes

Representación

Representación como
problema de
permutaciones

Soluciones

Complejidad

Búsqueda Local

Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

El nombre se deriva de los métodos propuestos para encontrar recorridos para el TSP llamados recorridos " k -opt". El procedimiento más sencillo en esta categoría es el 2-intercambio (2-opt).

El procedimiento 2-opt es un método de búsqueda local y se requiere una solución inicial válida para comenzar.

2-opt

Introducción

Descripción del problema

Variantes

Representación

Representación como
problema de
permutaciones

Soluciones

Complejidad

Búsqueda Local

Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

El método reemplaza dos aristas no adyacentes (v_i, v_{i+}) y (v_j, v_{j+}) por otras dos (v_i, v_j) y (v_{i+}, v_{j+}) , que son las únicas otras aristas que pueden crear un recorrido válido cuando se eliminan las primeras. Para mantener la orientación del recorrido, uno de los dos sub-recorridos que resultan de eliminar las primeras dos aristas debe ser invertido.

2-opt

Introducción

Descripción del problema

Variantes

Representación

Representación como
problema de
permutaciones

Soluciones

Complejidad

Búsqueda Local

Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

Cambio en el costo

El cambio de costo en la solución generado por un movimiento 2-intercambio se puede expresar como

$$\Delta_{ij} = c(v_i, v_j) + c(v_{i+}, v_{j+}) - c(v_i, v_{i+}) - c(v_j, v_{j+})$$

Una solución 2-óptima se obtiene aplicando movimientos de 2-intercambio hasta que ningún movimiento posible resulte en una reducción del costo.

k-opt

Introducción

Descripción del problema

Variantes

Representación

Representación como
problema de
permutaciones

Soluciones

Complejidad

Búsqueda Local

Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

El procedimiento 2-opt puede ser generalizado para realizar movimientos k -opt que eliminan k aristas y que agregan k nuevas. Existen $\binom{n}{k}$ formas de eliminar k aristas en un recorrido y $(k-1)!2^{k-1}$ formas de reconectar los sub-recorridos. Por esto, movimientos k -opt con $k > 3$ se consideran imprácticos a menos que se restrinja el tamaño del vecindario.

Extensión

Introducción

Descripción del problema

Variantes

Representación

Representación como
problema de
permutaciones

Soluciones

Complejidad

Búsqueda Local

Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

Una extensión que busca reducir la complejidad del procedimiento 3-opt es 2.5-opt, que es extiende el procedimiento 2-opt que considera una inserción de un único nodo cuando el 2-opt falla en encontrar mejoras.

Inserción de nodo único

Un nodo seleccionado v_i se inserta entre dos nodos adyacentes v_p y v_q agregando las aristas (v_p, v_i) , (v_i, v_q) , (v_{i-}, v_{i+}) y eliminando (v_p, v_q) , (v_{i-}, v_i) , (v_i, v_{i+}) .

Algoritmo Genético

Introducción

Descripción del problema

Variantes

Representación

Representación como
problema de
permutaciones

Soluciones

Complejidad
Búsqueda Local
Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

La idea principal de los algoritmos evolutivos es que dada una población de individuos en un ambiente con recursos limitados, la competición por esos recursos causa una selección natural. Dada una función de calidad a ser maximizada, se crean aleatoriamente soluciones candidatas y se prueban contra esta función de calidad, y con base en esto, se selecciona a los más aptos para construir la siguiente generación de soluciones.

Algoritmo Genético

Introducción

Descripción del problema

Variantes

Representación

Representación como
problema de
permutaciones

Soluciones

Complejidad

Búsqueda Local

Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

Para construir a la siguiente generación de soluciones se aplican operadores de recombinación o de mutación.

Recombinación

Es un operador que se aplica a dos o más candidatos seleccionados y que produce uno o más soluciones candidatas.

Mutación

Es un operador que se aplica a una solución candidata para modificarla y producir una solución distinta.

Algoritmo Genético

Introducción

Descripción del problema

Variantes

Representación

Representación como
problema de
permutaciones

Soluciones

Complejidad

Búsqueda Local

Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

La aplicación de los operadores de recombinación y mutación sobre las soluciones candidatas, lleva la creación de un conjunto de nuevas soluciones (hijos). Estas nuevas soluciones son evaluadas y compiten para tomar lugar en la nueva generación. Este procedimiento se realiza de forma iterativa hasta que se cumplan las condiciones de paro del algoritmo.

Algoritmo Genético

Introducción

Descripción del problema

Variantes

Representación

Representación como
problema de
permutaciones

Soluciones

Complejidad
Búsqueda Local
Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

Iteración de un algoritmo evolutivo

- 1 Seleccionar padres
- 2 Recombinar soluciones
- 3 Mutar las soluciones resultantes
- 4 Evaluar a los nuevos candidatos
- 5 Seleccionar a los individuos para la nueva generación

Los algoritmos evolutivos siguen este procedimiento general, variando en detalles técnicos. En el caso particular del Algoritmo Genético, se sigue este procedimiento, lo que lo caracteriza es que la representación de las soluciones se hace en cadenas sobre un alfabeto finito.

Introducción

Descripción del problema

Variantes

Representación

Representación como
problema de
permutaciones

Soluciones

Complejidad

Búsqueda Local

Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

- Implementación del procedimiento 2-opt
- Implementación del Algoritmo Genético con cruza ordenada
- Implementación del Algoritmo Genético sin operadores de cruza especiales (Pendiente de revisar)

Algoritmo Genético con cruza ordenada

Introducción

Descripción del problema

Variantes

Representación

Representación como problema de permutaciones

Soluciones

Complejidad

Búsqueda Local

Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

Para la implementación del Algoritmo Genético se utilizó una representación de soluciones como arreglos de enteros, donde el arreglo almacena la permutación correspondiente a un recorrido.

A partir de cada generación de n soluciones candidatas se generan $2n$ soluciones hijas que competirán para formar parte de la siguiente generación.

Algoritmo Genético con cruza ordenada

Introducción

Descripción del problema

Variantes

Representación

Representación como
problema de
permutaciones

Soluciones

Complejidad

Búsqueda Local

Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

La selección de los padres que transmiten su material a la siguiente generación se hace por torneo binario. Una vez que se elige un par de padres, se recombinan para generar 4 soluciones candidatas. Esta recombinación se logra mediante un operador de cruza ordenada.

Algoritmo Genético con cruza ordenada

Introducción

Descripción del problema

Variantes

Representación

Representación como
problema de
permutaciones

Soluciones

Complejidad
Búsqueda Local
Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

Puede notarse que una recombinación clásica del algoritmo genético, generaría constantemente soluciones que no son válidas para el TSP, pues produciría soluciones en las que dos nodos pueden ser visitados dos o más veces, dependiendo del número de soluciones padres que se utilicen para generar descendencia.

Ejemplo

Si se tienen dos soluciones candidatas a recombinar $s_1 = (1, 2, 3, 4, 5)$ y $s_2 = (3, 5, 1, 4, 2)$ y se utiliza el operador de cruza a un punto dividiendo entre los nodos 2 y 3, se generarían dos soluciones hijas $s_3 = (1, 2, 1, 4, 2)$ y $s_4 = (3, 5, 3, 4, 5)$ que no son soluciones válidas para el TSP.

Algoritmo Genético con cruza ordenada

Introducción

Descripción del problema

Variantes

Representación

Representación como
problema de
permutaciones

Soluciones

Complejidad

Búsqueda Local

Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

El operador de cruza ordenada selecciona aleatoriamente nodos de las soluciones padre y los recombina de forma que se generen soluciones válidas para el TSP.

Cruza Ordenada

Dados dos padres, se selecciona un subconjunto de nodos de uno de ellos (p_1) y se copia en uno de los hijos, el resto de los nodos en el recorrido son seleccionados de la otra solución padre (p_2) de forma que mantengan el orden que se tiene en esta solución y que el hijo generado sea una solución válida. Para el otro hijo se hace algo parecido pero primero se copian los nodos de p_2 y se impone el orden de p_1 en los restantes.

Algoritmo Genético con Cruza ordenada

Introducción

Descripción del problema

Variantes

Representación

Representación como
problema de
permutaciones

Soluciones

Complejidad

Búsqueda Local

Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

La mutación utilizada por ahora consiste en intercambiar el orden de visita de dos ciudades en el recorrido.

Algoritmo Genético sin operadores de cruce especiales (por revisar)

Introducción

Descripción del problema

Variantes

Representación

Representación como problema de permutaciones

Soluciones

Complejidad

Búsqueda Local

Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

La idea consiste en hacer uso de una representación diferente para poder utilizar los operadores de cruce clásicos del Algoritmo Genético.

Secuencia de inversión de una permutación

Dada una permutación $\{\pi(1), \dots, \pi(n)\}$ denotaremos con a_j al número de enteros en la permutación que preceden a j pero que son mayores que j . Para la permutación $(6, 2, 3, 4, 1, 7, 5)$, la secuencia de inversión es $4, 1, 1, 1, 2, 0, 0$.

Algoritmo Genético sin operadores de cruce especiales

Introducción

Descripción del problema

Variantes

Representación

Representación como
problema de
permutaciones

Soluciones

Complejidad

Búsqueda Local

Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

Si se utiliza esta secuencia de inversión en lugar de la representación por permutaciones, es posible utilizar los operadores de cruce clásicos del algoritmo genético, ya que no hay una restricción que nos diga que los elementos de esta representación deben ser diferentes entre sí.

Algoritmo Genético con Cruza ordenada

Introducción

Descripción del problema

Variantes

Representación

Representación como
problema de
permutaciones

Soluciones

Complejidad

Búsqueda Local

Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

La cruza que se ha implementado con esta representación es una cruza a dos puntos. La mutación utilizada por ahora consiste en intercambiar el orden de visita de dos ciudades en el recorrido.

Más avances e ideas

Introducción

Descripción del problema

Variantes

Representación

Representación como
problema de
permutaciones

Soluciones

Complejidad

Búsqueda Local

Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

- Se está haciendo uso de las instancias de TSPLIB para probar las soluciones generadas.
- Se implementó una forma de cargar instancias euclidianas y explícitas.
- Por ahora solo se han utilizado dos instancias para probar que las soluciones generadas sean válidas. Se piensa utilizar más para comparar resultados.
- Pendiente de revisar que las soluciones generadas por el acercamiento de secuencias de inversión sean válidas.

Más avances e ideas

Introducción

Descripción del problema

Variantes

Representación

Representación como
problema de
permutaciones

Soluciones

Complejidad

Búsqueda Local

Algoritmo Genético

Avances

Algoritmo Genético

Más avances e ideas

- Se tiene pensado implementar el procedimiento 2.5-opt.
- Se piensa implementar una mutación basada en movimientos 2-opt para el Algoritmo Genético.
- Se piensa implementar un algoritmo híbrido agregando búsquedas locales.
- Análisis estadístico de los resultados.