Búsqueda Local: Conectividad

Tecnología Digital V: Diseño de Algoritmos

Universidad Torcuato Di Tella

Recapitulando

1. Dado el estado (solución factible) actual, nos movemos a una solución en el vecindario definido por N_{op} que sea mejor.
2. El algoritmo converge a un óptimo local.
3. No puede garantizar convergencia a un **óptimo global**.

Motivación

Recapitulando

1. Dado el estado (solución factible) actual, nos movemos a una solución en el vecindario definido por N_{op} que sea mejor.
2. El algoritmo converge a un óptimo local.
3. No puede garantizar convergencia a un **óptimo global**.

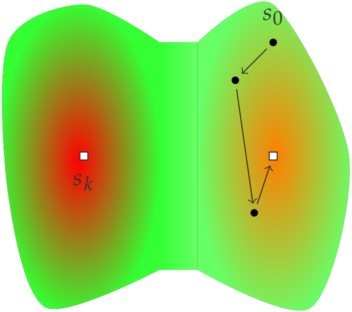
Observación

Quizás moverse siempre a una solución que mejore es demasiado *codicioso*. Quizas podemos relajar esta condición...

Pregunta

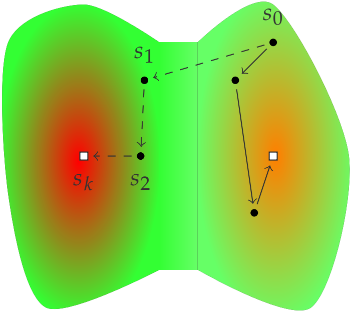
Dada una solución factible $s \in S$ y un vecindario $N(s)$, existe una forma de llegar desde s a una sólución $s^* \in S$ que sea un óptimo global?

Búsqueda local: Conectividad



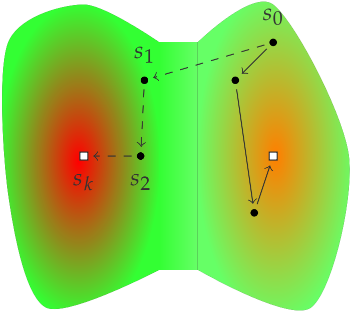
Def: Vecindario Conectado

Un vecindario N_{op} se dice **conectado** si para toda solución factible $s_0 \in S$, existe una secuencia de pasos $s_i \in N_{\text{op}}(s_{i-1}), 1 \leq i \leq k$ tal que s_k sea un óptimo global.



Def: Vecindario Conectado

Un vecindario N_{op} se dice **conectado** si para toda solución factible $s_0 \in S$, existe una secuencia de pasos $s_i \in N_{\text{op}}(s_{i-1}), 1 \leq i \leq k$ tal que s_k sea un óptimo global.

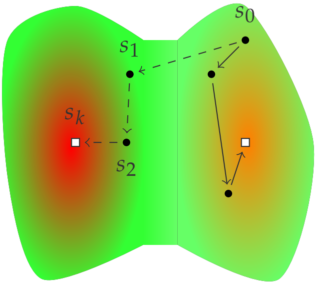


Def: Vecindario Conectado

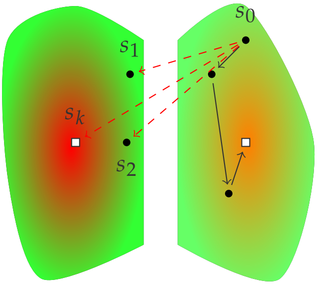
Un vecindario N_{op} se dice **conectado** si para toda solución factible $s_0 \in S$, existe una secuencia de pasos $s_i \in N_{\text{op}}(s_{i-1}), 1 \leq i \leq k$ tal que s_k sea un óptimo global.

Lema

Sea N_{op} un vecindario no conetado, y $s \in S$ una solución no conectada para N_{op}. Entonces para todo $s' \in N_{\text{op}}(s)$ tiene que ser necesariamente una solución no conectada dado N_{op}.

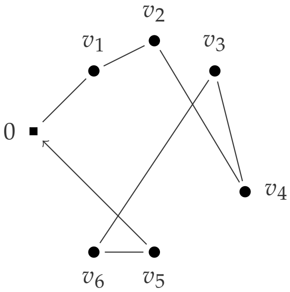


Vecindario conectado



Vecindario no conectado

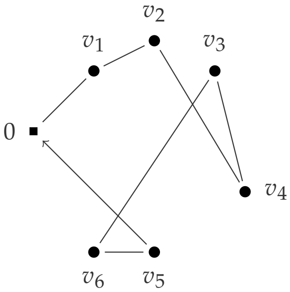
Ejemplo: TSP y Swap



Operador swap

Tomar dos vértices e intercambiar sus posiciones.

$$s = (0, v_1, v_2, v_4, v_3, v_6, v_5, 0)$$

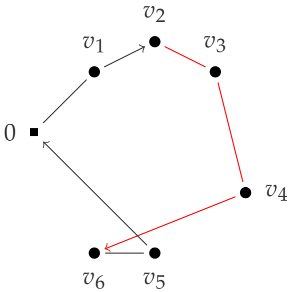


Operador swap

Tomar dos vértices e intercambiar sus posiciones.

$$s = (0, v_1, v_2, v_4, v_3, v_6, v_5, 0)$$

Aplicamos $s' = \textbf{swap}(s, v_3, v_4)$
$s' = (0, v_1, v_2, v_3, v_4, v_6, v_5, 0)$



Ejemplo: TSP y Swap

Pregunta

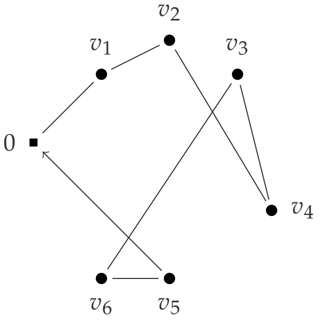
Como demostramos que un vecindario N_{op} es conectado?

Pregunta

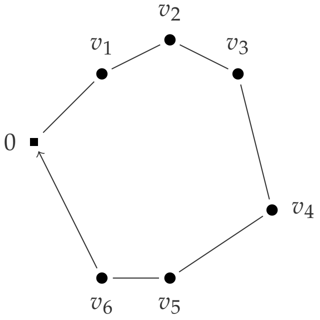
Como demostramos que un vecindario N_{op} es conectado?

Respuesta

Suponemos que conocemos una solución óptima genérica, mostramos explícitamente como armar la secuencia de transformaciones para una solución incial $s_0 \in S$.

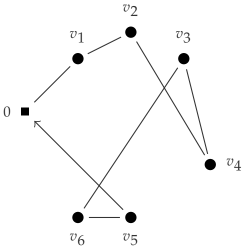


Solución inicial (s_0)



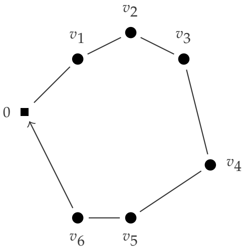
Solución óptima (π)

Búsqueda local: Conectividad



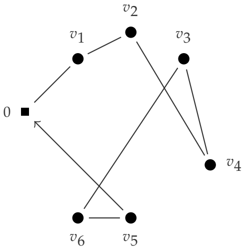
$s_0 = (0, v_1, v_2, v_4, v_3, v_6, v_5, 0)$

Solución inicial



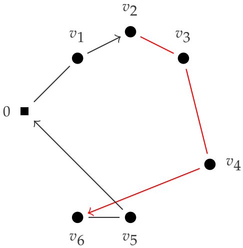
$\pi = (0, v_1, v_2, v_3, v_4, v_5, v_6, 0)$

Solución óptima



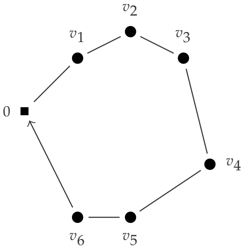
$s_0 = (0, v_1, v_2, v_4, v_3, v_6, v_5, 0)$

Solución inicial



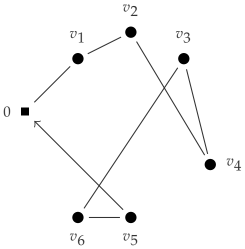
$s_1 = (0, v_1, v_2, v_3, v_4, v_6, v_5, 0)$

$s_1 = \textbf{swap}(s_0, v_3, v_4)$



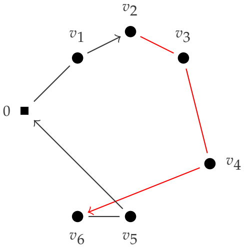
$\pi = (0, v_1, v_2, v_3, v_4, v_5, v_6, 0)$

Solución óptima



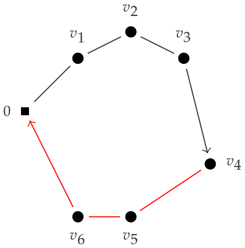
$s_0 = (0, v_1, v_2, v_4, v_3, v_6, v_5, 0)$

Solución inicial



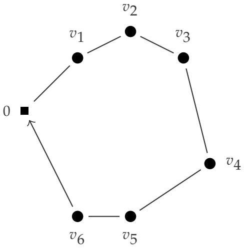
$s_1 = (0, v_1, v_2, v_3, v_4, v_6, v_5, 0)$

$s_1 = \textbf{swap}(s_0, v_3, v_4)$



$s_2 = (0, v_1, v_2, v_3, v_4, v_5, v_6, 0)$

$s_2 = \textbf{swap}(s_1, v_5, v_6)$



$\pi = (0, v_1, v_2, v_3, v_4, v_5, v_6, 0)$

Solución óptima

TSP y Swap: Formalización

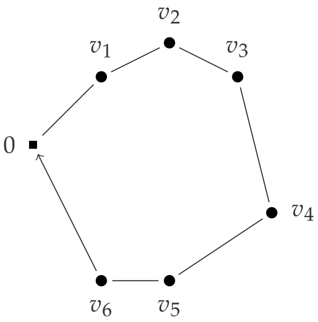
- s: solución inicial
- π: solución óptima

1. **Para** $i = 1, \ldots, n$
2. **Si** $\pi_i \neq s_i$
3. Sea s_j tal que $s_j == \pi_i$
4. Aplicar $s = \mathbf{swap}(s, s_j, \pi_i)$ (Notar que necesariamente $j > i$)
5. **Fin Si**
6. **Fin Para**

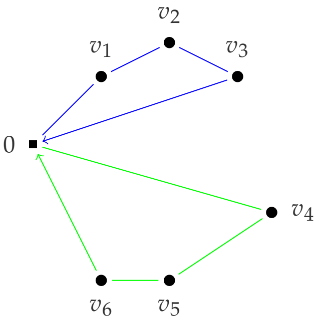
Segundo ejemplo: m-TSP

Definición: m-TSP

Consideramos un grafo completo $G = (V, E)$, con $V = \{0, 1, \ldots, n\}$, o representa el depósito. Sea c_{ij} el costo asociado al arco (i, j) y $m \geq 1$ la cantidad de vehículos. El objetivo del m-TSP es determinar un circuito para cada vehículo de forma tal que cada vértice $i \in V \backslash \{0\}$ sea visitado una única vez, minimizando el costo total, empezando y terminando en el depósito.

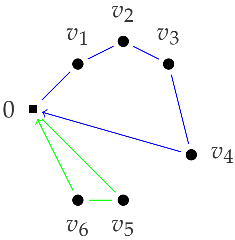


TSP

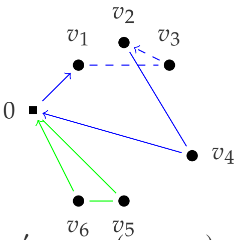


m-TSP $(m = 2)$

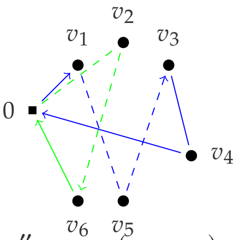
Generalizamos **swap** a múltiples rutas.



s



$s' = \textbf{swap}(s, v_2, v_3)$



$s'' = \textbf{swap}(s, v_2, v_5)$

Pregunta

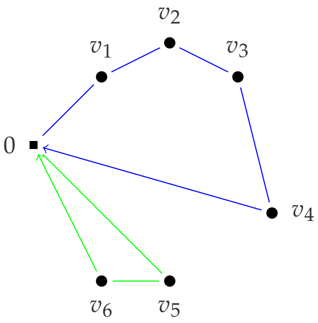
El operador **swap** es conectado para el m-TSP?

Pregunta

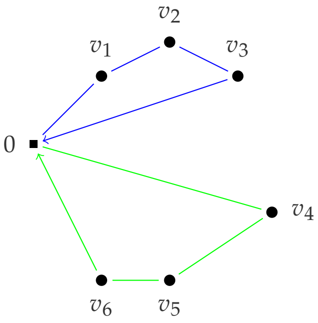
El operador **swap** es conectado para el m-TSP?

Observación

swap no altera la cardinalidad (cantidad de vértices) de una ruta.



Solución factible (s)



Solución óptima (π)

Conectividad y búsqueda local: Para qué?

1. Es una propiedad deseable para un operador, ya que nos garantiza que existe una forma de llegar a un óptimo global del problema.

2. En problemas con un gran número de restricciones los operadores suelen quedar desconectados.

3. Conceptualmete es una propiedad importante para escapar de óptimos locales:
   - **Expandir el vecindario:** generar un árbol **acotado** de secuencias de aplicaciones del operador. Ejemplo: *Chained Lin-Kernighan* para TSP.
   - **Esquemas meta-heurísticos:** bajo determinadas circunstancias, moverse a una solución del vecindario que no sea necesariamente mejor. Ejemplo: Simmulated Annealing, Tabú Search.