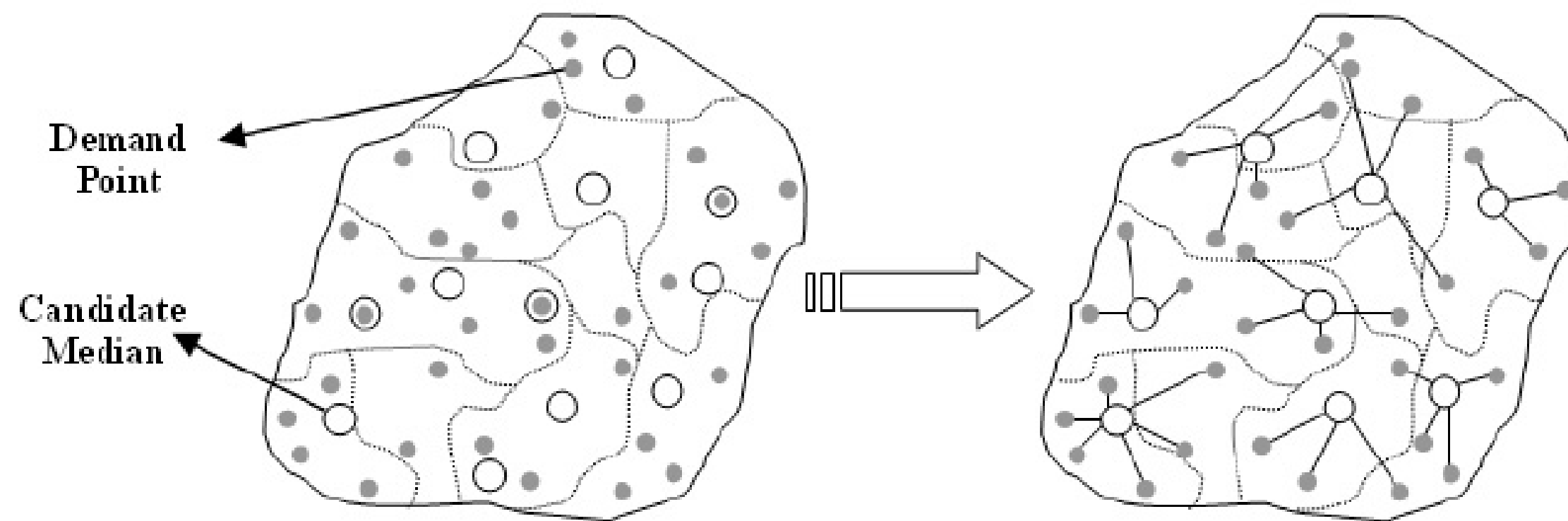


Método evolutivo híbrido

para el problema de

p mediana capacitado



Por: Luisa Toro Villegas

Problema:

p mediana capacitado

2

El problema consiste en encontrar un subconjunto de p nodos en los cuales se localizará una instalación con el fin de abastecer la demanda del conjunto de nodos.

$$\text{Min } Z = \sum_{i \in V} \sum_{j \in V} w_{ij} \cdot x_{ij}$$

$$\sum_{i \in V} y_i = p$$

$$\sum_{j \in V} x_{ij} = 1$$

$$\forall i \in V$$

$$\sum_{i \in V} d_i \cdot x_{ij} \leq c_j \cdot y_j$$

$$\forall j \in V$$

$$x_{ij} \in \{0,1\}$$

$$\forall i \in V, j \in V$$

$$y_i \in \{0,1\}$$

$$\forall i \in J$$

Metodología

Método constructivo

1. Matriz de distancias euclidianas entre todos los nodos.
2. Se escogen los p centros con menor suma total de distancias euclidianas.
3. Se asignan el resto de los nodos al centro que, esté más cerca y sea factible (la demanda sea menor o igual a la capacidad).
4. Se verifica la factibilidad de la solución.
5. Se determina el valor de la función objetivo.

Método GRASP

5

1. Matriz de distancias euclidianas entre todos los nodos.
2. Se escogen p centros aleatorios.
3. Se asignan el resto de los nodos al centro que, esté más cerca y sea factible (la demanda sea menor o igual a la capacidad).
4. Se verifica la factibilidad de la solución.
5. Se repite este proceso n veces, y se escoge la mejor solución (la que tenga la menor función objetivo).

Método constructivo con ruido

6

1. Se le añade ruido al conjunto de datos en las coordenadas x y y sumándole una distribución uniforme entre -1 y 1 a cada una.
2. Matriz de distancias euclidianas entre todos los nodos.
3. Se escogen los p centros con menor suma total de distancias euclidianas.
4. Se asignan el resto de los nodos al centro que, esté más cerca y sea factible (la demanda sea menor o igual a la capacidad).
5. Se verifica la factibilidad de la solución.
6. Se determina el valor de la función objetivo.

VND

Búsqueda en vecindarios variables

1. Se construye una solución inicial con el método constructivo, el GRASP o el constructivo con ruido.
2. Se le pasa la solución a un vecindario, el cual se la pasa al siguiente y luego al siguiente.
3. Se itera hasta un tiempo máximo (según el archivo).
4. Se escoge la nueva solución construida a partir de los 3 algoritmos.

1. Toma la solución inicial S .
2. Itera a través de la lista de nodos escogidos como centros y selecciona otro centro. Asigna los nodos de nuevo y revisa si la solución es mejor que la anterior (la función objetivo es menor).
3. Retorna cuando encuentre una mejora en la solución.

Vecindario: cambio de medianas

1. Toma una solución inicial.
2. Para la lista de nodos escogidos como centros, escoge los n nodos más lejanos a cada centro.
3. Esos nodos los desasigna y pasa a un método que los asigna a los centros que tengan más cercano.
4. Retorna la solución con los nodos reasignados.

Vecindario: desasignación y reasignación

1. Toma una solución inicial S .
2. Para cada par de centros, exploro sus nodos asignados y los intercambio de a pares. Si el cambio mejora la solución, lo conservo.
3. Retorno la solución con todos los cambios.

**Vecindario:
intercambio
entre pares de
nodos**

Búsqueda

11

1. Se construye una solución inicial con el método constructivo, el GRASP o el constructivo con ruido.
2. Se le pasa la solución a los diferentes algoritmos, los cuáles buscan una mejor solución (cada uno).
3. Se itera hasta un tiempo máximo (según el archivo).
4. Se escoge la nueva solución construida a partir de los 3 algoritmos.

1. Toma una solución inicial S .
2. Desasigna $m = \text{percentage} * \text{nodos}$ nodos aleatorios.
3. Los reasigna según cuál centro le es más cercano a cada uno.
4. Retorna la solución con los nodos reasignados.

Destrucción y reparación

1. Escoger una solución inicial S .
2. Se escogen los hiperparámetros temperatura inicial T_0 , número de iteraciones L , gradiente g .
3. Se genera otra solución S' según un vecindario.
4. $\Delta = f(S') - f(S)$
5. Si $\Delta \leq 0$, $S = S'$. Si $\Delta > 0$, $S = S'$ con cierta función de probabilidad.
6. Se cambia la temperatura según T .
7. Se detiene cuando se acaban las iteraciones.

$$P_T\{\text{acepta } j\} = \begin{cases} 1, & f(j) \leq f(i) \\ e^{\left(\frac{f(i) - f(j)}{T}\right)}, & f(j) > f(i) \end{cases}$$

Aceptación probabilística: recocido simulado (SA)

1. Genero una solución inicial escogiendo centros aleatorios y asignando los nodos a estos según cuál es el más cercano.
2. Reviso factibilidad.
3. Retorno la solución.

**Multiples
soluciones
iniciales**

Evolutivo

Híbrido

1. Se construyen multiples soluciones iniciales con el método GRASP.
2. Se le pasan las soluciones al método de intercambio de medianas.
3. Se buscan las medianas en común entre las soluciones y se construyen nuevas soluciones.
4. Se verifica si la solución es mejor.
5. Si queda tiempo, se le vuelve a pasar la solución a intercambio de medianas hasta que se acaba el tiempo.

Revisión de similaridades

16

1. Según cuantas soluciones se escojan, se determina que se hace. Si $M = p$, entonces se escogen las M como centros y se realiza la asignación.
2. Si $M < p$, se escogen $p - M$ centros mediante la construcción del constructivo y se realiza la asignación.
3. Si $M > p, n$ se realizan las combinaciones y se verifica cual es mejor.
4. Si la solución generada es mejor, se escoge.

Resultados

Formato

Nodos centro

Número de nodos
asignados

[illegible]

Función objetivo

Tiempo total

Hiperparámetros evolutivo

GRASP

- numofsolutions: número de soluciones iniciales a generar.

Median search

- changes: número de búsquedas de cambios a realizar

Genetic

- numinitialsolutions: número de soluciones GRASP a generar.
- M: número mejores soluciones a escoger entre las generadas y mejoradas.

Cota inferior

20

Archivo	Cota inferior		
		pmedcap8.txt	21 335
pmedcap1.txt	270	pmedcap9.txt	28 393
pmedcap2.txt	293	pmedcap10.txt	27 240
pmedcap3.txt	384	pmedcap11.txt	69 562
pmedcap4.txt	380	pmedcap12.txt	73 462
pmedcap5.txt	8 014	pmedcap13.txt	77 362
pmedcap6.txt	15 395	pmedcap14.txt	81 271
pmedcap7.txt	21 928	pmedcap15.txt	85 271

Tiempos

21

Archivo	Tiempo (min)		
		pmedcap8.txt	5
pmedcap1.txt	1	pmedcap9.txt	5
pmedcap2.txt	1	pmedcap10.txt	5
pmedcap3.txt	1	pmedcap11.txt	10
pmedcap4.txt	1	pmedcap12.txt	10
pmedcap5.txt	1	pmedcap13.txt	10
pmedcap6.txt	5	pmedcap14.txt	10
pmedcap7.txt	5	pmedcap15.txt	10

Función objetivo

22

	pmedcap1.txt	pmedcap4.txt	pmedcap10.txt	pmedcap15.txt
GRASP	838 - 210%	1 325 - 249%	68 065 - 150%	227 230 - 166%
Constructivo	1151 - 326%	2 932 - 672%	257 168 - 844%	2 121 826 - 2388%
Evolutivo	722 - 167%	1 087 - 186%	64 204 - 136%	216 595 - 154%
Búsqueda local	785 - 191%	1 084 - 185%	66 150 - 143%	224 865 - 164%

Archivo	Diferencia		
		pmedcap8.txt	124%
pmedcap1.txt	167%	pmedcap9.txt	172%
pmedcap2.txt	153%	pmedcap10.txt	136%
pmedcap3.txt	206%	pmedcap11.txt	134%
pmedcap4.txt	186%	pmedcap12.txt	138%
pmedcap5.txt	126%	pmedcap13.txt	151%
pmedcap6.txt	149%	pmedcap14.txt	150%
pmedcap7.txt	147%	pmedcap15.txt	154%

Experimentación

Evolutivo

25

MedianSearch : changes = 100
GRASP: numInitialSolutions = 50

		pmedcap1.txt	pmedcap5.txt	pmedcap10.txt
Evolutivo		722 - 167%	18 115 - 126%	64 204
20	M = 10	749 - 177%	18 890 - 136%	64 805
10	M = 10	760 - 181%	19 186 - 139%	67 170
10	M = 5	767 - 184%	20 252 - 153%	68 373

NumInitialSolutions

Evolutivo

26

Genetic:
numInitialSolutions = 10
M = 5

		pmedcap1.txt	pmedcap5.txt	pmedcap10.txt
Evolutivo		781 - 189%	20 460 - 155%	63 051 - 131%
200	100	738 - 173%	20 253 - 153%	65 700 - 141%
200	200	761 - 182%	20 579 - 157%	65 038 - 139%
500	200	722 - 167%	19 988 - 149%	63 000 - 131%

NumInitialSolutions changes

Conclusiones

Conclusiones

- Mientras más *soluciones iniciales*, tanto para **GRASP** como para el **genético**, mejores resultados se tienen.
 - El hiperparámetro de *changes* en el **genético** no es tan importante porque de igual manera se itera en el método.
- No se considera el tiempo ya que los algoritmos corren por un tiempo establecido (no se incluye el tiempo de encontrar la solución inicial).

Referencias

Referencias

30

Alkhedhairi, Abdulrahman. (2008). Simulated annealing metaheuristic for solving P-median problem. Int. J. Contemp. Math. Sciences. 3. 1357-1365.

Ghoseiri, K., & Farid Ghannadpour, S. (2013). (PDF) an efficient heuristic method for capacitated P-median problem. ResearchGate. Retrieved March 28, 2022, from https://www.researchgate.net/publication/255592172_An_efficient_heuristic_method_for_capacitated_P-Median_problem

Levanova, T. V., & Gnusarev, A. Y. (2018, July). Simulated annealing for competitive p-median facility location problem. In Journal of Physics: Conference Series (Vol. 1050, No. 1, p. 012044). IOP Publishing.