

Diseño y Análisis de Algoritmos
Práctica 4: Algoritmos constructivos y búsquedas por entornos
Max-mean dispersion problem
Víctor Hernández Pérez
alu0100697032

Estructuras de datos:

Se ha optado por representar la solución como un array de booleanos en el que los nodos que se encuentran en la solución se encuentran a true y los que no se encuentran a false.

El problema consta de una matriz de distancias y se guardan también los nodos que están fuera de la solución.

Algoritmos:

Algoritmos voraces: el primero de ellos es el que está explicado en el enunciado de la práctica, parte de la arista de mayor coste y va metiendo los nodos que van maximizando el valor objetivo si se da ese caso. El segundo parte de una solución con todos los nodos dentro de la solución y se van quitando de la misma los nodos que, al ser eliminados, maximicen el valor objetivo.

GRASP: este algoritmo consta de dos fases. En la fase constructiva se crea una lista restringida de candidatos (LRC) formada por las tres aristas de mayor coste, se elige uno de estos candidatos de manera aleatoria el cual definirá los dos nodos que formarán parte de la solución. En la búsqueda local (segunda fase del algoritmo), se van añadiendo nodos a la solución para ver si nuestra solución actual mejora. La búsqueda local para cuando no mejore más.

VNS: se crea una solución inicial formada por un número aleatorio de nodos. A partir de esta solución inicial agitamos según el número de entornos que hayamos atravesado. Agitar la solución consiste en intercambiar aleatoriamente nodos que se encuentran en la solución con nodos que están fuera de la solución. Si $k = 1$ intercambiamos 1 nodo de la solución por 1 nodo fuera de esta, etc.

Si la solución no mejora aumentamos el número de entornos en 1. Si se da el caso de que mejora, el número de entornos vuelve a 1. El algoritmo para cuando el número de entornos llegue al máximo número de entornos permitido definido como $kmax$.

Multiarranque: Ejecutamos un número determinado de veces nuestro algoritmo GRASP, el cual nos devuelve en cada iteración una solución. Si la solución mejora nuestra solución actual, actualizamos.

Conclusiones:

Las ejecuciones se han hecho con unos tamaños de prueba pequeños elaborados por el autor de esta memoria. En el caso de los algoritmos voraces el valor objetivo obtenido en las distintas ejecuciones para un mismo test dan el mismo resultado. En el caso del segundo voraz, el rendimiento suele ser mayor dado que, para los test probados, cuanto mayor es el número de nodos suele ser mayor el valor objetivo y este algoritmo empieza con todos los nodos dentro de la solución.

En el caso del GRASP, al ser probado en instancias tan pequeñas, no se percibe la aleatoriedad del algoritmo.

Tablas:

Voraz

Problema	N	Número de ejecución	Valor objetivo (md)	CPU (segundos)
test4	4	1	11.5	7.14654E-4
test4	4	2	11.5	7.78179E-4
test10	10	1	38.2	0.002040735
test10	10	2	38.2	0.00175974
test12	12	1	47.6	0.002471577
test12	12	2	47.6	0.00240088
test15	15	1	58.2	0.003384234
test15	15	2	58.2	0.003570198

Voraz 2

Problema	N	Número de ejecución	Valor objetivo (md)	CPU
test4	4	1	11.5	6.4294E-5
test4	4	2	11.5	4.5339E-5
test10	10	1	38.2	1.05021E-4
test10	10	2	38.2	5.6097E-5
test12	12	1	47.6	7.3515E-5
test12	12	2	47.6	6.1476E-5
test15	15	1	58.2	7.9918E-5
test15	15	2	58.2	7.5564E-5

GRASP

Problema	N	LRC	Número de ejecución	Valor objetivo (md)	CPU
test4	4	2	1	11.5	4.46979E-4
test4	4	2	2	11.5	4.44418E-4
test4	4	3	1	11.5	4.65422E-4
test4	4	3	2	11.5	5.78639E-4
test10	10	2	1	38.2	6.73158E-4
test10	10	2	2	38.2	5.98363E-4
test10	10	3	1	38.2	7.09531E-4
test10	10	3	2	38.2	0.001041243
test12	12	2	1	47.6	7.62042E-4
test12	12	2	2	47.6	6.82892E-4
test12	12	3	1	47.6	8.4196E-4
test12	12	3	2	47.6	8.38118E-4
test15	15	2	1	58.2	0.001200568
test15	15	2	2	58.2	9.72084E-4
test15	15	3	1	58.2	0.00102485
test15	15	3	2	58.2	0.001116039

Multiarranque

Problema	N	Número de ejecución	Valor objetivo (md)	CPU
test4	4	1	11.5	3.5246E-4
test4	4	2	11.5	5.11017E-4
test10	10	1	38.2	0.001167013
test10	10	2	38.2	0.001102207
test12	12	1	47.6	0.001561225
test12	12	2	47.6	0.001609637
test15	15	1	58.2	0.00274207
test15	15	2	58.2	0.002826086

VNS

Problema	N	K	Número de ejecución	Valor objetivo (md)	CPU
test4	4	2	1	4.5	7.0697E-5
test4	4	2	2	5.5	1.09119E-4
test4	4	3	1	8.6	9.7336E-5
test4	4	3	2	8.6	1.20646E-4
test10	10	2	1	7	8.914E-5
test10	10	2	2	10	1.65728E-4
test10	10	3	1	12.6	9.3238E-5
test10	10	3	2	14.0	9.9385E-5
test12	12	2	1	12.5	8.0687E-5
test12	12	2	2	12.5	6.9416E-5
test12	12	3	1	10.3	7.1466E-5
test12	12	3	2	14.3	1.01435E-4
test15	15	2	1	7.5	9.375E-5
test15	15	2	2	7.5	1.09376E-4
test15	15	3	1	13.6	1.2833E-4
test15	15	3	2	14.3	9.6056E-5