

A dark blue vertical bar is on the left side of the page. A blue arrow-shaped banner points to the right from this bar, containing the date. In the bottom left corner, there are several thin, curved, light grey lines that sweep upwards and to the right.

16-5-2022

PRÁCTICA 2: ALGORITMOS HEURÍSTICOS NO CONSTRUCTIVOS, BÚSQUEDA VNS Y GENÉTICOS

Modelos Bioinspirados y Heurísticas
de Búsqueda

José M^a Prieto Crespo
UNIVERSIDAD DE HUELVA

ÍNDICE

| | |
|---|---|
| 1. VNS | 2 |
| 2. AG | 3 |
| 3. CHC | 5 |
| 4. Multimodal Clearing para AG estacionario | 7 |
| 5. Comparación | 9 |

1. VNS

Algoritmo VNS implementado con la condición de parada donde se sigue intentando un número de búsquedas locales aunque se haya llegado a kmax, por tanto voy a hacer un estudio cambiando Blmax para ver que combinación me da los mejores resultados.

L = 16 individuos

Semillas = [10, 20, 30]

Kms comienzo: [810, 678, 625]

Primer intento

Kmax: 4

Bl max: 5

Evaluaciones: [1224, 1299, 3855]

Mínimos kms: [506, 528, 484]

Capacidades: [212, 211, 213]

Segundo intento

Kmax: 4

Bl max: 10

Evaluaciones: [7502, 10831, 10796]

Mínimos kms: [491, 526, 485]

Capacidades: [212, 211, 213]

Tercer intento

Kmax: 4

Bl max: 15

Evaluaciones: [6333, 9985, 17239]

Mínimos kms: [494, 528, 485]

Capacidades: [212, 211, 213]

Tras probar con distintos valores de BLmax, a partir de las 10 las evaluaciones son demasiadas para unos resultados similares a las 10 BL, por lo que me quedo con las 10 BL.

2. AG

Para la implementación del algoritmo genético he elegido:

- El **tipo** estacionario.
- **Población inicial** de 16 individuos
- **Mutación** con operador de movimiento de 3, tras los resultados obtenidos en la BL para la anterior práctica. Al estar usando el operador de movimiento, modifiko un par de cromosomas cada vez que muto, por lo que el porcentaje de cromosomas que muto va a ser la mitad. De esta forma, para conseguir una mutación del 16% lo que hago es una mutación a partir del operador de movimiento con un 8%, siendo así el doble el 16%.
- **Cruce**: para cada uno de los hijos tomo una serie de posiciones de forma aleatoria para los padres y las intercambio. De esta forma hay mas diversidad y los hijos no se parecen tanto a los padres. Además, en caso de los hijos superar los 205 de capacidad, es decir, no válidos, les asigno un valor de fitness muy alto, de forma que duren muy poco en la población.
- Para la **selección** he usado torneo para un 30% de la población, asegurándome de que sea un mínimo de 3 individuos los que entran en este.
- En cuanto al **reemplazo** el elegido ha sido el de reemplazar el menor, provocando esto muchísima presión selectiva.
- La **condición de parada** la he fijado en "250 no mejoras", es decir, que si el algoritmo no mejora durante 250 iteraciones me salgo del bucle. En caso de que si mejore, busco en 250 iteraciones más (y así sucesivamente).

Voy a llevar a cabo un estudio para los porcentajes de mutación, así como el castigo en el fitness en caso de superar los 205 de capacidad para ver cual es la mejor combinación de valores.

L = 16 individuos

Semillas = [10, 20, 30]

Mínimos kms comienzo: [639, 647, 631]

Primer intento

Mutación del 16%

Castigo = 3 * Km de más

Evaluaciones: [1012, 1170, 1236]

Mínimos kms: [142, 206, 255]

Capacidades mínimas: [264, 250, 237]

A pesar de tener capacidades más o menos altas, es la que mejor resultado da en cuanto a kms de lejos.

Segundo intento

Mutación del 16%

Castigo = 8 * Km de más

Evaluaciones: [860, 1174, 1128]

Mínimos kms: [362, 304, 320]

Capacidades mínimas: [222, 228, 226]

Bajan las capacidades mucho pero aumentan casi al doble los kms totales a ejecutar

Tercer intento

Mutación del 24%

Castigo = 8 * Km de más

Evaluaciones: [1114, 1142, 1046]

Mínimos kms: [349, 321, 342]

Capacidades mínimas: [223, 226, 223]

No renta cambiar el porcentaje de mutación ya que no hay mucha diferencia.

Cuarto intento

Mutación del 24%

Castigo = 5 * Km de más

Evaluaciones: [1284, 1660, 1298]

Mínimos kms: [274, 301, 268]

Capacidades mínimas: [234, 228, 235]

No está mal, es un punto intermedio entre 8 kms de castigo y 3 kms. Mejor resultado de todos.

Quinto intento

Mutación del 16 %

Castigo = 5 * Km de más

Evaluaciones: [1224, 1750, 1182]

Mínimos kms: [236, 286, 281]

Capacidades mínimas: [242, 231, 232]

Los mejores valores han sido obtenidos para una mutación del 24% y un castigo de 5 por Km de más, a pesar de ser la segunda de las combinaciones que más evaluaciones tiene.

3. CHC

Para el cruce en la función “recombina” intercambio la mitad de los valores que difieren de los padres, además de aplicar la mutación “Parent Centered” para cada uno de los valores intercambiados, con una distribución centrada en el valor y una amplitud establecida en 2.

Como para el resto de algoritmos probados, he llevado a cabo un estudio modificando la condición de parada (número de reinicios de la población máximos) además de la élite que se guarda al reinicializar la población.

L = 16 individuos

Semillas = [10, 20, 30]

Mínimos kms comienzo: [639, 647, 631]

Primer intento

Numero Reinicios = 4

Elite = 2

Evaluaciones: [762, 750, 760]

Mínimos kms: [334, 345, 353]

Capacidades mínimas: [235, 237, 241]

Segundo intento

Numero reinicios: 4

Elite: 1

Evaluaciones: [742, 732, 746]

Mínimos kms: [365, 383, 303]

Capacidades mínimas: [232, 230, 237]

En general mejor solución que la anterior, ya que obtiene Kms parecidos y mejora la media de la capacidad. Esto se debe a que explora nuevas posibles soluciones, así como nuevos valores.

Tercero intento

Número reinicios: 2

Élite: 1

Evaluaciones: [462, 430, 456]

Mínimos kms: [343, 383, 330]

Capacidades mínimas: [244, 230, 235]

Cuarto intento

Número reinicios: 10

Élite: 1

Evaluaciones: [1644, 1612, 1414]

Mínimos kms: [334, 360, 314]

Capacidades mínimas: [233, 225, 233]

Quinto intento

Número reinicios: 10

Élite: 2

Evaluaciones: [1624, 1650, 1634]

Mínimos kms: [326, 268, 334]

Capacidades mínimas: [234, 245, 231]

Sexto intento

Número reinicios: 10

Élite: 5

Evaluaciones: [1552, 1564, 1546]

Mínimos kms: [275, 347, 272]

Capacidades mínimas: [247, 229, 240]

Los mejores obtenidos han sido el quinto y el sexto intento, sin embargo el mejor resultado lo he logrado con el sexto intento, que además baja el número de evaluaciones totales.

4. Multimodal Clearing para AG estacionario

En mi caso, el multimodal lo he implementado usando Clearing para un algoritmo genético estacionario con los mismos parámetros que los ya mencionados anteriormente.

Para este algoritmo, el valor del radio para la distancia Hamming es crítico, por lo que voy a hacer un estudio modificando los distintos valores de la distancia Hamming para poder comparar y obtener la mejor de las combinaciones.

El número de generaciones entre operaciones de aclarado lo he establecido en 10, por lo que se realizaría cada 80 generaciones ($10 \cdot 16 / 2$).

L= 16 individuos

Primer intento

Distancia Hamming: 5

Evaluaciones: [1914, 1580, 1854]

Mínimos kms: [282, 269, 265]

Capacidades mínimas: [232, 234, 236]

Segundo intento

Distancia Hamming: 7

Evaluaciones: [1736, 3018, 1406]

Mínimos kms: [281, 269, 268]

Capacidades mínimas: [232, 234, 235]

Al subir el tamaño de distancia mínima, se van a eliminar más elementos (y 5 ya era alto, por lo que no hay un cambio apreciable)

Tercer intento

Distancia Hamming: 3

Evaluaciones: [2360, 1760, 1920]

Mínimos kms: [274, 270, 276]

Capacidades mínimas: [234, 235, 234]

Cuarto intento

Distancia Hamming: 2

Evaluaciones: [3186, 2494, 1132]

Mínimos kms: [269, 261, 268]

Capacidades de esos kms: [235, 237, 235]

Mejores resultados para el cuarto intento con distancia hamming: 2

En caso de cambiar los castigos por kms sobrepasados cambiaría los resultados bastante, al igual que lo hace el algoritmo genético básico.

Si modifico el valor de p para el mejor resultado (D.H = 2):

Primer intento

Distancia Hamming: 2

P: 5

Evaluaciones: [1740, 1754, 2378]

Mínimos kms: [274, 269, 258]

Capacidades de esos kms: [234, 235, 237]

Segundo intento

Distancia Hamming: 2

P: 15

Evaluaciones: [1448, 1584, 2552]

Mínimos kms: [281, 280, 265]

Capacidades de esos kms: [233, 232, 236]

Tras los cambios se puede observar como no mejora, por lo que me quedo con los valores del cuarto intento con la p inicial.

5. Comparación

| Algoritmo | Ev. Medias | Ev. Mejor | Desv. Ev | Mejor Kms | Media Kms | Desv. Kms |
|-----------------|------------|-----------|----------|-----------|-----------|-----------|
| BL | 813.8 | 497 | 197.63 | 483.29 | 506.52 | 18.83 |
| VNS | 9709.66 | 7502 | 1561.121 | 485 | 500.67 | 18.08 |
| A.G. Básico | 1385.33 | 1182 | 258.43 | 236 | 267.67 | 22.48 |
| A.G. Multimodal | 2270.33 | 1132 | 852.925 | 261 | 266 | 3.56 |
| CHC | 1554 | 1546 | 7.483 | 272 | 298 | 34.67 |

El **algoritmo VNS** queda como perdedor de todos los probados, con una media de kms muchísimo más alta que el resto, ya que se diferencia en casi el doble. Esto seguramente se deba a un fallo de implementación que no he conseguido ver.

Como se puede observar, los mejores resultados han sido los obtenidos por el **algoritmo Genético Multimodal**, el cual además de dar el menor número de Kms totales, lo hace sin tener un gran número de evaluaciones medias. Lo bueno de este algoritmo es que no solo da los mejores resultados, sino que llega a dar varias buenas soluciones, permitiendo esto la convergencia a varios óptimos, a diferencia de otros algoritmos que convergen a cualquier óptimo.

El **algoritmo genético básico**, en su defecto, ha obtenido una peor media de Kms que el multimodal, pero esto ha hecho que las capacidades medias sean bastante menores también, por lo que en mi caso ha sido bastante más estable. En caso de tener que quedarme con uno, me quedaría con este.

Por último, el **algoritmo CHC**. Este, aparte de darme una mayor media de Kms que el A.G. básico y el multimodal, también obtiene unas capacidades bastante mayores que las de estos dos últimos mencionados, por lo que su uso en este problema no sería lo más lógico.