

Big Bang Big Crunch



Proyecto Final
Problema del Agrupamiento de Restricciones

Metaheurísticas
Alba Casillas Rodríguez
albacaro@correo.ugr.es
Curso 2019/2020

¿Cómo nace la metaheurística?

- En 2006, dos investigadores estambulís, Ibrahim Eksin y Osman Kaan Erol deciden crear un algoritmo de búsqueda inspirado en teorías de la creación, evaluación y destrucción del universo.
- Big Bang Big Crunch (BB-BC) empieza a ser utilizado para problemas de optimización y a ser comparado con distintos algoritmos genéticos debido a su estrecha similitud.

- En 2008 , Ali Koksal Hocaoglu y Hakki Murat Genç describen A BB-BC como un algoritmo principalmente caracterizado por una búsqueda rápida en el espacio de búsqueda y una explotación agresiva en del espacio de soluciones.



Fases principales:

BIG BANG

- *Conjunto de procedimientos de disipación de energía en la naturaleza basada en el desorden y la aleatoriedad.*
- *[Disipación de energía] Se genera una población de soluciones factibles aleatorias o mediante modificaciones.*
- *Representa el proceso de EXPLORACIÓN de soluciones.*

BIG BANG



BIG CRUNCH

BIG CRUNCH

- *Proceso de destrucción de la población manteniendo solamente los mejores.*
- *Al final nos quedamos con una única solución.*
- *Obtenemos los mejores candidatos para la creación de una nueva población mejorada.*
- *Se explora pero también se EXPLOTA las mejores soluciones.*

ELITE POOL

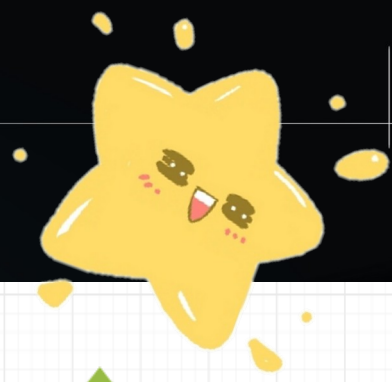
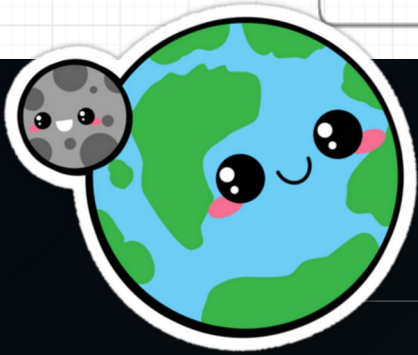
- Uno de los términos más importantes de esta práctica es el uso de un “Conjunto de élite” (elite pool).
- Se almacenará continuamente el 10% de las mejores soluciones de cada población.
- Motivación:
 - Se conoce en todo momento el valor de las mejores soluciones, permitiendo realizar una mayor explotación de estas.
 - La exploración del vecindario toma como “cota” el mejor y peor valor de este grupo, asegurando calidad en la exploración.
 - Las nuevas poblaciones se crearán a partir de estas, no desde el “vacío”, por lo que progresivamente nacerán poblaciones mejores.



Hibridación

- Para la hibridación se ha utilizado *Búsqueda Local*.
- La metaheurística ya es inicialmente una hibridación, por lo que se ha comparado un BB-BC sin *Búsqueda Local* con el BB-BC inicial (con BL).
- La *Búsqueda Local* se aplica a la mejor solución (el centro de la masa) de la población sucesivamente hasta que la población se reduce a un único elemento.
- Los resultados utilizando la hibridación son notablemente mejores que sin ella, igualando al resto de algoritmos.

Estructura del algoritmo



Implementación

Para adaptarlo a nuestro problema (PAR) se han realizado los siguientes cambios:

$$C_i^{new} = C_c + \sigma$$

$$\sigma = \frac{r \alpha (C_{max} - C_{min})}{k}, 0 < \frac{\alpha}{k} < 1$$

- *La generación de vecinos se hace mediante C_c (centro de la masa). Originalmente se trabaja con valores numéricos. Se ha considerado el centro de la masa como la mejor solución y no su valor.*
- *Cada vecino utiliza el centro de la masa al que se le modifican posiciones que tendrán el valor de su padre C_i .*
- *Sigma indicará el número de posiciones del centro de la masa a cambiar.*

*** Estos cambios se explican en profundidad en la documentación*

Resultados

	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T
COPKM	8,13	448,40	645,61	1652,94
BL	7,69	709,20	1015,93	2388,08
COPKM-Arreg	37,24	537,40	51,56	13,93
BL-Arreglado	22,39	55,40	23,86	156,64
AGG-UN	25,23	528,00	39,30	868,21
AGG-SF	27,61	418,00	38,75	875,07
AGE-UN	22,24	45,00	23,44	926,75
AGE-SF	21,41	103,00	24,14	985,57
AMI0-1.0	41,48	1377,00	78,18	898,825
AMI0-0.1	33,78	675,00	51,77	890,42
AMI0-0.1MEJ	29,43	682,00	47,61	891,21
ES	21,48	47,40	22,74	452,35
BMB	21,97	136,80	25,09	958,46
ILS	35,28	77,80	23,67	947,45
ILS-ES	36,82	163,80	26,06	998,13
BB-BC	34,66	1088,20	66,60	877,18
BB-BC + BL	22,07	69,40	23,92	937,91

-Solo obtenemos buenos resultados en BB-BC+BL (BB-BC original).

- Los resultados de BB-BC + BL son tan buenos como los del resto de algoritmos, superando a los genéticos y meméticos.

-Esta tabla muestra los resultados del conjunto más complejo, Ecoli.

