UNIVERSIDAD DE GRANADA
E.T.S.I. INFORMÁTICA Y TELECOMUNICACIÓN

UNIVERSIDAD DE GRANADA

Departamento de Ciencias de la Computación e Inteligencia Artificial

Metaheurísticas

http://sci2s.ugr.es/graduateCourses/Metaheuristicas
https://decsai.ugr.es

Guión de Prácticas

Práctica 2.b:
Técnicas de Búsqueda basadas en Poblaciones para el Problema del Agrupamiento con Restricciones

Curso 2019-2020

Tercer Curso del Grado en Ingeniería Informática

Práctica 2.b

Técnicas de Búsqueda basadas en Poblaciones para el Problema del Agrupamiento con Restricciones

1. Objetivos

El objetivo de esta práctica es estudiar el funcionamiento de las *Técnicas de Búsqueda basadas en Poblaciones* en la resolución del Problema del Agrupamiento con Restricciones (PAR) descrito en las transparencias del Seminario 2. Para ello, se requerirá que el estudiante adapte las siguientes técnicas metaheurísticas al problema:

- Algoritmos Genéticos (AGs): Dos variantes generacionales elitistas (AGGs) y otras dos estacionarias (AGEs), descritas en el Seminario 3. Aparte del esquema de evolución, la única diferencia entre los dos modelos de AGG y AGE será el operador de cruce empleado.
- Algoritmos Meméticos: Tres variantes de algoritmos meméticos (AMs) basadas en un AGG, descritas en el Seminario 3. La única diferencia entre las tres variantes de AMs serán los parámetros considerados para definir la aplicación de la búsqueda local. Para diseñar los AMs, se utilizará un método de Búsqueda Local Suave (BLS) descrito en el Seminario 3.

El estudiante deberá comparar los resultados obtenidos en una serie de casos del problema con los proporcionados por el obtenido por el método *greedy* COPKM y el resultante de la propia **BL original**; todos ellos implementados en la Práctica 1.b.

La práctica se evalúa sobre un total de **2,5 puntos**, distribuidos de la siguiente forma:

- AGGs (0,75 puntos) y AGEs (1 punto).
- AMs (0,75 puntos).

La fecha límite de entrega será el **lunes 20 de abril de 2020** antes de las 23:55 horas. La entrega de la práctica se realizará por internet a través del espacio de la asignatura en PRADO.

2. Trabajo a Realizar

El estudiante podrá desarrollar los algoritmos de la práctica siguiendo la modalidad que desee: trabajando con cualquiera de los *frameworks* de metaheurísticas estudiados en el Seminario 1, implementándolos a partir del código C proporcionado en la web de la asignatura o considerando cualquier código disponible en Internet.

Los métodos desarrollados serán ejecutados sobre una serie de casos del problema. Se realizará un estudio comparativo de los resultados obtenidos y se analizará el comportamiento de cada algoritmo en base a dichos resultados. **Este análisis influirá decisivamente en la calificación final de la práctica**.

En las secciones siguientes se describen los aspectos relacionados con cada algoritmo a desarrollar y las tablas de resultados a obtener. Los casos del problema serán los mismos que en la Práctica 1.b **pero se ha añadido un conjunto de datos adicional**. De igual manera, el número de ejecuciones a realizar sobre ellos, el procedimiento de validación y los estadísticos de calidad (*Tasa_C*, *Tasa_inf*, *Agregado* y *Tiempo*) serán los mismos que en el guión de la Práctica 1.b. (véase la Sección 3 de dicho guión de prácticas).

3. Conjuntos de Datos Considerados

El estudiante trabajará con **8 instancias del PAR** generadas a partir de los **4 conjuntos de datos** siguientes (obtenidos de la web http://www.ics.uci.edu/~mlearn/MLRepository.html, procesados en algún caso y disponibles en el espacio de PRADO y en la web de la asignatura):

1. **Iris**: Clásico en ciencia de datos. Contiene información sobre las características de tres tipos de flor de Iris. Tiene 3 clases ($k = 3$).
2. **Ecoli**: Contiene medidas sobre ciertas características de diferentes tipos de células que pueden ser empleadas para predecir la localización de ciertas proteínas. Tiene 8 clases ($k = 8$).
3. **Rand:** Conjunto de datos artificial. Formado por tres agrupamientos bien diferenciados generados en base a distribuciones normales ($k = 3$).
4. **Newthyroid:** Contiene medidas cuantitativas tomadas sobre la glándula tiroides de 215 pacientes. Presenta 3 clases distintas ($k = 3$).

A partir de cada conjunto de datos se obtienen 2 instancias distintas generando 2 conjuntos de restricciones, correspondientes al 10% y 20% del total de restricciones posibles. Los llamaremos CS_{10} y CS_{20}, respectivamente.

Los ficheros *.dat* corresponden a los conjuntos de datos. Los ficheros *.const* corresponden a los conjuntos de restricciones. Por ejemplo: *iris_set.dat* es el fichero de datos de Iris, y los conjuntos de restricciones son *iris_const_set_10.const* e *iris_set_const_20.const*. Los ficheros de datos contienen en cada fila una instancia del conjunto en cuestión con sus características separadas por comas (sin espacios). Los ficheros de restricciones almacenan una matriz de restricciones con el formato descrito en las transparencias del Seminario 2, con los valores separados por comas y sin espacios.

4. Componentes de los Algoritmos

Los algoritmos de esta práctica tienen en común las siguientes componentes:

- *Esquema de representación*: Se seguirá la representación entera basada en un vector *S* de tamaño *n* (número de instancias del conjunto de datos) con valores en *{1, ..., k}* (siendo *k* el número de agrupamientos, que coincidirá con el del clases del conjunto de datos real) que indican el cluster asociado a cada instancia, explicado en las transparencias del seminario.

- *Función objetivo*: Será la combinación mediante el peso λ de las medidas de $\overline{C}$ (desviación general de la partición) e *infeasability* resultantes del agrupamiento obtenido por la solución representada en el vector *S*. El valor de λ está explicado en las transparencias del Seminario 2. El objetivo será minimizar esta función.

- *Generación de la solución inicial*: La solución inicial se generará de forma aleatoria escogiendo un valor en *{1, ..., k}* en cada posición del vector *S*, **comprobando que cada cluster tiene al menos una instancia asignada**.

- *Esquema de generación de vecinos*: Se empleará el movimiento de cambio de cluster que altera el vector *S* sustituyendo el valor s_i en una posición *i*-ésima por otro valor $l \in \{1, ..., k\}\ y\ l \neq s_i$, **comprobando que no deja ningún cluster vacío**.

- *Criterio de aceptación*: Se considera una mejora cuando disminuye el valor global de la función objetivo.

  A continuación veremos las particularidades de cada algoritmo.

4.1. Algoritmos Genéticos

Algoritmos

Los AGs de esta práctica presentarán las siguientes componentes:

- *Esquema de evolución*: Como se ha comentado, se considerarán dos versiones, una basada en el esquema generacional con elitismo (AGG) y otra basada en el esquema estacionario (AGE). En el primero se seleccionará una población de padres del mismo tamaño que la población genética mientras que en el segundo se seleccionarán únicamente dos padres.

- *Operador de selección*: Se usará el torneo binario, consistente en elegir aleatoriamente dos individuos de la población y seleccionar el mejor de ellos. En el esquema generacional, se aplicarán tantos torneos como individuos existan en la población genética, incluyendo los individuos ganadores en la población de padres. En el esquema estacionario, se aplicará dos veces el torneo para elegir los dos padres que serán posteriormente recombinados (cruzados).

- *Esquema de reemplazamiento*: En el esquema generacional, la población de hijos sustituye automáticamente a la actual. Para conservar el elitismo, si la mejor solución de la generación anterior no sobrevive, sustituye directamente la peor solución de la nueva población. En el estacionario, los dos descendientes generados tras el cruce y la mutación (esta última aplicada según una determinada probabilidad) sustituyen a los dos peores de la población actual, en caso de ser mejores que ellos.

- *Operador de cruce*: Se emplearán dos operadores de cruce para representación real, explicados en las transparencias de teoría del tema de AGs y del Seminario 3. Uno de ellos será el operador uniforme mientras que el otro será el cruce por segmento fijo. **Esto resultará en el desarrollo de cuatro AGs distintos, dos generacionales (AGG-UN y AGG-SF) y dos estacionarios (AGE-UN y AGE-SF)**

- *Operador de mutación*: Emplearemos el operador de mutación uniforme. Supongamos que hemos determinado que debe mutar un gen en un cromosoma. Basta con generar dos números aleatorios, uno para determinar el gen que muta (en el intervalo $\{0, \ldots, n-1\}$) y otro para determinar el nuevo valor (que debe ser distinto del actual y mantener las restricciones). Coincidirá por tanto con el operador de generación de vecinos de la BL.

Valores de los parámetros y ejecuciones

El tamaño de la población será de 50 cromosomas. La probabilidad de cruce será 0,7 en el AGG y 1 en el AGE (siempre se cruzan los dos padres). La probabilidad de mutación (**por gen**) será de 0,001 en ambos casos. El criterio de parada en las dos versiones del AG consistirá en realizar **100000 evaluaciones de la función objetivo**.

4.2. Algoritmos Meméticos

Algoritmos

El AM consistirá en hibridar el AGG que mejor resultado haya proporcionado con una nueva BLS descrita en el Seminario 3. Se estudiarán las tres posibilidades de hibridación siguientes:

1. AM-(10,1.0): Cada 10 generaciones, se aplica la BLS sobre todos los cromosomas de la población.
2. AM-(10,0.1): Cada 10 generaciones, se aplica la BLS sobre un subconjunto de cromosomas de la población seleccionado aleatoriamente con probabilidad p_{LS} igual a 0.1 para cada cromosoma.
3. AM-(10,0.1mej): Cada 10 generaciones, aplicar la BLS sobre los $0.1 \cdot N$ mejores cromosomas de la población actual (N es el tamaño de ésta).

Valores de los parámetros y ejecuciones

El tamaño de la población del AM será de 10 cromosomas. Las probabilidades de cruce y mutación serán 0,7 (por cromosoma) y 0,001 (por gen) en ambos casos. Cuando

la BLS no produce cambios en el cromosoma decimos que falla. El contador de fallos está diseñado para evitar que se desperdicien evaluaciones de la función objetivo en cromosomas de mucha calidad. Permitiremos como mucho ξ fallos en la BLS, siendo $\xi = 0{,}1 \cdot n$. El criterio de parada del AM consistirá en realizar **100000 evaluaciones de la función objetivo**, incluidas por supuesto las de la BLS.

5. Tablas de Resultados a Obtener

Se diseñará una tabla para cada algoritmo (COPKM, BL, AGG-UN, AGG-SF, AGE-UN, AGE-SF, AM-(10,1.0), AM-(10,0.1) y AM-(10,0.1mej)) y cada porcentaje de restricciones donde se recojan los resultados de la ejecución de dicho algoritmo en los conjuntos de datos considerados. Tendrá la misma estructura que la Tabla 6.1 de la Práctica 1.b. Los resultados para COPKM y BL también serán los obtenidos en la Práctica 1.b en los conjuntos de datos previos utilizados en la Práctica 1.

Finalmente, se construirán dos tablas de resultados globales (una por porcentaje de restricciones) que recojan los resultados medios de calidad y tiempo para todos los algoritmos considerados, tal como se muestra en la Tabla 5.1. Para rellenar estas tablas se hará uso de los resultados medios mostrados en las tablas parciales. Aunque en la tabla que sirve de ejemplo se han incluido todos los algoritmos considerados en esta práctica, naturalmente sólo se incluirán los que se hayan desarrollado.

Tabla 5.1: Resultados globales en el PAR con XX% de restricciones

	Iris				**Ecoli**				**Rand**				**Newthyroid**			
	Tasa_C	***Tasa_inf***	***Agr.***	**T**	***Tasa_C***	***Tasa_inf***	***Agr.***	**T**	***Tasa_C***	***Tasa_inf***	***Agr.***	**T**	***Tasa_C***	***Tasa_inf***	***Agr.***	**T**
COPKM	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
BL	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
AGG-UN	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
AGG-SF	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
AGE-UN	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
AGE-SF	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
AM-(10,1.0)	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
AM-(10,0.1)	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
AM-(10,0.1mej)	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

A partir de los datos mostrados en estas tablas, el estudiante realizará un análisis de los resultados, ***que influirá significativamente en la calificación de la práctica***. En dicho análisis se deben comparar los distintos algoritmos en términos de las tasas de desviación general de la partición obtenidas (capacidad del algoritmo para obtener soluciones de calidad), número de restricciones no satisfechas y tiempos requeridos para obtener las soluciones (rapidez del algoritmo). En esta práctica se comparará el rendimiento de las metaheurísticas consideradas con respecto al algoritmo de referencia, el COPKM, y la BL de la práctica anterior. En las siguientes prácticas se comparará también el rendimiento de las distintas metaheurísticas consideradas entre sí.

6. Documentación y Ficheros a Entregar

Además de la documentación detallada en la Sección 7 del guión de la Práctica 1.b, en lo referente al punto d) se incluirá, al menos, la siguiente información:

1. Esquema de representación de soluciones empleado.
2. Descripción en pseudocódigo de la función objetivo.
3. Pseudocódigo del proceso de generación de soluciones aleatorias.
4. Pseudocódigo de la selección de los AGs y los operadores de cruce y mutación.

En lo que respecta al punto e), se incluirá la siguiente información:

1. Para los AGs, el esquema de evolución y de reemplazamiento considerados.
2. Para los AMs, el esquema de búsqueda seguido por cada algoritmo en lo que respecta a la integración de la BL dentro del AG.

Como recomendación, el apartado d) debería describirse en un máximo de cuatro páginas. En el apartado e), el número total de páginas para describir cada algoritmo (incluyendo el pseudocódigo del esquema de búsqueda y de las componentes particulares) sería de dos páginas.

Aunque lo esencial es el contenido, también debe cuidarse la presentación y la redacción. Se recuerda que **la documentación nunca deberá incluir listado total o parcial del código fuente en caso de haberlo implementado**. En lo referente al **desarrollo de la práctica**, se seguirán los mismos criterios descritos en la Sección 6 del guion de la Práctica 1.b. El **método de evaluación** será el de la Sección 7 de ese guion.