

**UNIVERSIDAD DE JAÉN**  
**ESCUELA POLITÉCNICA SUPERIOR DE JAÉN**



Departamento de Informática. Área de Ciencias  
de la Computación e Inteligencia Artificial

## **Metaheurísticas**

### **Guión de Prácticas**

**Práctica 2: Metaheurísticas basadas en poblaciones;  
Algoritmos Genéticos**

Curso 2016-2017

Tercer Curso del Grado en Ingeniería Informática

# Práctica 2

## Algoritmos Genéticos

### 1. Objetivos

El objetivo de esta práctica es estudiar el funcionamiento de los *Algoritmos Genéticos* (AGs). Para ello, se requerirá que el alumno adapte el algoritmo generacional elitista (AGG) y el algoritmo estacionario (AGE) para resolver el problema del set covering problem (SCP) descrito en las transparencias del Seminario 2 y que compare los resultados obtenidos con el algoritmo *Greedy*.

La práctica se evalúa sobre un total de **4 puntos**, distribuidos de la siguiente forma: AGG (2,25 puntos) y AGE (1,75 puntos).

La fecha de explicación de la práctica será el **lunes 24 de octubre o el martes 25 de octubre**. La fecha límite de entrega será **20 de noviembre de 2016** antes de las 23:59 horas. La defensa de las prácticas será el 21 de noviembre o el 22 de noviembre durante la hora de prácticas. La entrega de la práctica se realizará por internet a través de la correspondiente actividad habilitada para ello en la plataforma ILIAS.

### 2. Trabajo a Realizar

El alumno 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 de código propio 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 y los estadísticos de calidad (*Desv* y *Tiempo*) serán los mismos que en la Práctica 1 (véase la Sección 3 de dicho guión de prácticas).

### 3. Componentes de los Algoritmos

#### Algoritmos

Los AGs de esta práctica presentarán las siguientes componentes:

- *Esquema de representación*: Se seguirá la representación binaria (explicada en la práctica 1).
- *Función objetivo*: Minimizar Z

$$\text{mín}(z) = \sum_{j=1}^n c_j x_j$$

- *Generación de la población inicial*: Todos los cromosomas se generarán de forma aleatoria usando un método simple que genera soluciones factibles y sin columnas redundantes. El método simple consiste en ir seleccionando de forma aleatoria columnas (que cubran objetos no cubiertos previamente), para de esta forma completar el individuo. Posteriormente se eliminan las columnas redundantes.
- *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 sustituyen a los dos peores de la población actual, en caso de ser mejores que ellos.
- *Operadores de cruces*: Se emplearán dos de cruce distintos: Cruce Fusión y Cruce HUX en el algoritmo AGG. Para el algoritmo AGE solo se aplicará el cruce HUX. Ambos cruces son explicados en el seminario 3
- *Operador de mutación*: Selecciona un individuo (considerando la probabilidad de mutación por cromosoma) y cambia el valor de los genes (según una

probabilidad variable de mutación por gen) de 0 a 1 y de 1 a 0. La probabilidad variable permite que al principio se muten más genes y conforme avanza las generaciones se reduce el número genes mutados.

- **Mecanismo de Reparación:** Para reparar el cromosoma se aplica el algoritmo Greedy, explicado en las transparencias del Seminario 2 y 3.
- **Reinicialización:** Se aplicará una reinicialización para evitar un estancamiento de la búsqueda. Será dinámico y se podrán hacer 0, 1 o muchas veces, dependerá de la evolución del algoritmo. En concreto se aplicará una reinicialización cuando se cumpla una de las siguientes condiciones:
  - Después de 20 generaciones no se haya conseguido mejorar la mejor solución encontrada hasta el momento, para el AGG. No se considera en el AGE.
  - O en el caso de que la población converja hacia una misma solución, es decir, que el 80% de la población contengan el mismo individuo. Para ambos algoritmos AGG y AGE.
- La forma de realizar la reinicialización es incluir la mejor solución y el resto de soluciones se generaran aleatoriamente siguiendo el mismo proceso usado para generar la población inicial.

### 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 cromosoma) será de 0.02 en ambos casos y la probabilidad de mutación (por gen) será variable comenzando en 0,2 y cada 1000 evaluaciones se excrementara 0,01. El criterio de parada consistirá en realizar **20000 evaluaciones de la función objetivo**.

## 4. Tablas de Resultados a Obtener

Se diseñará una tabla para cada algoritmo donde se recojan los resultados de la ejecución de dicho algoritmo al conjunto de casos del problema. Tendrá la misma estructura que la Tabla 5.1 del guión de la Práctica 1.

Finalmente, se construirá una tabla de resultados global que recoja los resultados medios de calidad y tiempo para todos los algoritmos considerados, tal como se muestra en la tabla 4.1. 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 4.1: Resultados globales en el SCP

Algoritmo	Desv	Tiempo
<i>Greedy</i>	x	x
AGG + fusión	x	x
AGG + HUX	x	x

AGE + HUX	x	x
-----------	---	---

A partir de los datos mostrados en estas tablas, el alumno realizará un análisis de los resultados obtenidos, que influirá significativamente en la calificación de la práctica. En dicho análisis se deben comparar los distintos algoritmos en términos de calidad de las soluciones y tiempo requerido para obtenerlas. Por otro lado, se puede analizar también el comportamiento de los algoritmos en algunos de los casos individuales que presenten un comportamiento más destacado.

## 5. Documentación y Ficheros a Entregar

Además de la documentación detallada en la Sección 6 del guión de la Práctica 1, en lo referente al punto 4 se incluirá, al menos, la siguiente información:

- a) Esquema de representación de soluciones empleado.
- b) Descripción en pseudocódigo de la función objetivo.
- c) Descripción en pseudocódigo del proceso de generación de soluciones iniciales.
- d) Descripción en pseudocódigo del mecanismo de selección considerado.
- e) Descripción en pseudocódigo de los operadores de cruce y mutación empleados.

En lo que respecta al punto 5, se incluirá la descripción en pseudocódigo de los esquemas de evolución y reemplazamiento considerados. En el punto 6, se incluirá una breve descripción del algoritmo de comparación, el *Greedy*.

Como recomendación, el apartado 4 debería describirse en un máximo de dos páginas y los apartados 5 y 6 en una página cada uno.

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 guión de la Práctica 1. El **método de evaluación** será el descrito en la Sección 7 de dicho guión.