

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



Departamento de Ciencias de la  
Computación e Inteligencia Artificial

**Metaheurísticas**

<http://sci2s.ugr.es/graduateCourses/Metaheuristics>  
<https://decsai.ugr.es>

**Guión de Prácticas**

**Práctica 2.a:**  
**Técnicas de Búsqueda basadas en Poblaciones**  
**para el Problema de la Asignación Cuadrática**

Curso 2017-2018

Tercer Curso del Grado en Ingeniería Informática

## Práctica 2.a

# Técnicas de Búsqueda basadas en Poblaciones para el Problema de la Asignación Cuadrática

## 1. Objetivos

El objetivo de esta práctica es estudiar el funcionamiento de las *Técnicas de Búsqueda basadas en Poblaciones* en la resolución del problema de la asignación cuadrática (QAP) descrito en las transparencias del Seminario 2. Para ello, se requerirá que el estudiante adapte las siguientes técnicas metaheurísticas a dicho problema:

- Algoritmos Genéticos: 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 AGGs 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 el método de Búsqueda Local (BL) desarrollado en la Práctica 1.a.

El estudiante deberá comparar los resultados obtenidos con las estimaciones existentes para el valor de los óptimos de una serie de casos del problema, así como con el algoritmo *greedy* básico y la búsqueda local (BL) descritos en el Seminario 2 y desarrollados en la Práctica 1.a.

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 **Viernes 4 de Mayo de 2018** antes de las 23:59 horas. La entrega de la práctica se realizará por internet a través del acceso identificado de la web del departamento de CCIA (<https://decsai.ugr.es>)

## 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 y los estadísticos de calidad (*Desv* y *Tiempo*) serán los mismos que en la Práctica 1.a (véase las Secciones 3 y 4 de dicho guión de prácticas).

## 3. Componentes de los Algoritmos

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

- *Esquema de representación*: Se seguirá la representación en forma de permutación  $\pi$  de tamaño  $n$  que representa una asignación de unidades (los índices del vector) a localizaciones (los contenidos del mismo).
- *Función objetivo*: 
$$\min_{\pi \in \Pi_N} \left( \sum_{i=1}^n \sum_{j=1}^n f_{ij} \cdot d_{\pi(i)\pi(j)} \right)$$
- *Generación de la solución inicial o la población inicial*: La(s) solución/soluciones iniciales se generarán de forma aleatoria en todos los casos.

A continuación veremos las particularidades de cada algoritmo.

### 3.1. 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 de orden, explicados en las transparencias de teoría del tema de Algoritmos Genéticos y del Seminario 3. Uno de ellos será obligatoriamente el de posición mientras que el otro se escogerá entre el OX y el PMX. **Esto resultará en el desarrollo de cuatro AGs distintos, dos generacionales (AGG-posición y AGG-otro) y dos estacionarios (AGE-posición y AGE-otro).**
- *Operador de mutación:* Se considerará el operador de intercambio para representación de orden, explicado en las transparencias del Seminario 2 y de teoría. Para aplicarlo, se escogerá aleatoriamente otro gen con el que intercambiar el patrón existente en el gen a mutar.

### 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 **50000 evaluaciones de la función objetivo**. Se realizará una única ejecución sobre cada caso del problema.

## 3.3. Algoritmos Meméticos

### Algoritmos

El AM consistirá en hibridar el algoritmo genético generacional (AGG) que mejor resultado haya proporcionado con la BL desarrollada. Se estudiarán las tres posibilidades de hibridación siguientes:

1. AM-(10,1.0): Cada 10 generaciones, se aplica la BL sobre todos los cromosomas de la población.
2. AM-(10,0.1): Cada 10 generaciones, se aplica la BL 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 BL 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 AGG será de 10 cromosomas. Las probabilidades de cruce y mutación serán 0,7 y 0,001 (por gen) en ambos casos. Se detendrá la ejecución de la BL aplicada sobre un cromosoma bien **cuando no se encuentre mejora en todo el entorno** o bien **cuando se hayan evaluado 400 vecinos distintos en la ejecución**. El criterio de parada del AM consistirá en realizar **50000 evaluaciones de la función objetivo**, incluidas por supuesto las de la BL. Se realizará una única ejecución sobre cada caso del problema.

## 4. Tablas de Resultados a Obtener

Se diseñará una tabla para cada algoritmo (AGG-posición, AGG-otro, AGE-posición, AGE-otro, AM-(10,1.0), AM-(10,0.1) y AM-(10,0.1mej)) 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.a.

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. Los resultados del Greedy y de la BL corresponden a los de la Práctica 1.a.

Tabla 4.1: Resultados globales en el QAP

Algoritmo	Desv	Tiempo
Greedy	x	x
BL	x	x
AGG-posición	x	x
AGG-otro	x	x
AGE-posición	x	x
AGE-otro	x	x
AM-(10,1.0)	x	x
AM-(10,0.1)	x	x
AM-(10,0.1mej)	x	x

A partir de los datos mostrados en estas tablas, el estudiante 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 producirlas. 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.a, en lo referente al punto d) se incluirá, al menos, la siguiente información:

1. Esquema de representación de soluciones.
2. Descripción en pseudocódigo (no código) de la función objetivo.
3. Descripción en pseudocódigo de los operadores comunes: el operador de generación de vecino/mutación, la generación de soluciones aleatorias, el mecanismo de selección de los AGs y los operadores de cruce y mutación empleado

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

1. Descripción en pseudocódigo de la estructura del método de búsqueda.
2. Además se detallarán, al menos, las siguientes componentes particulares de cada algoritmo:
  - a. Para los AGs, el esquema de evolución y de reemplazamiento considerados.
  - b. 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 guión de la Práctica 1.a. El **método de evaluación** será el descrito en la Sección 7 de dicho guión.