

Grado en Ingeniería Informática

Metaheurísticas

Curso 2019/2020



Universidad de Jaén

Práctica 2

1. Objetivos

El objetivo de esta práctica es estudiar el funcionamiento de las Metaheurísticas basadas en poblaciones, y en concreto de los Algoritmos Genéticos. Para ello, se requerirá que el estudiante implemente un:

- Algoritmo Genético Generacional.

La fecha límite de entrega será el **viernes 22 de noviembre de 2019 antes de las 23:59 horas**. La entrega se realizará a través de la plataforma virtual.

2. Trabajo a realizar

Este guion será una continuación del primero, por tanto, el trabajo a realizar y los casos de estudio permanecen sin cambios.

3. Algoritmo genético

Se aplicará un único algoritmo con un modelo evolutivo generacional que se detalla a continuación.

Antes es importante destacar que para la **representación de soluciones y la función objetivo se mantienen los mismos conceptos del guion anterior**, y que la **población inicial deberá generar todas las soluciones factibles de forma completamente aleatoria**.

Generacional

Los distintos componentes del algoritmo son:

- Esquema de evolución que hará evolucionar la población completa con una **élite de 2 individuos** y otro esquema con una **élite de 3 individuos**.
- Operador de selección basada en un torneo binario, es decir con $k=2$, aplicándose tantas veces como individuos tengamos en la población. En

este proceso todos los individuos de la población inicial (incluyendo los individuos ganadores en la población anterior) entran en el proceso del Torneo.

- Reemplazamiento que será realizado con el reemplazo de la población completa por la población descendiente. Para conservar el elitismo, si las mejores soluciones de la generación anterior no sobreviven, sustituyen directamente a las peores soluciones de la nueva población.
- Operador de cruce se deberán implementar dos operadores el **cruce OX2** y el **cruce MOC** (se explicarán en clase).
- Operador de mutación será un intercambio de tres posiciones de la solución.

Parámetros de los algoritmos

Se detallan a continuación todos los parámetros de los algoritmos:

- Tamaño de la población de 50 individuos.
- Máximo número de evaluaciones 50.000 (condición de parada).
- Probabilidad de cruce de 0.7.
- Probabilidad de mutación para cada gen será de 0.05.

Resumen de los algoritmos

- Generacional con élite 2 y cruce OX2.
- Generacional con élite 3 y cruce OX2.
- Generacional con élite 2 y cruce MOC.
- Generacional con élite 3 y cruce MOC.

4. Tablas de resultados y análisis

Seguiremos trabajando el mismo formato que en el guión anterior.

Para ello, se cogerán los resultados obtenidos en la Práctica 1 y se incluirán los mismos obtenidos por los algoritmos diseñados en este guión.

El análisis que se debe realizar de hacerse de la siguiente forma y justificarlo:

1. Elite 2 cruce OX2 Vs. Elite 2 cruce MOC
2. Elite 3 cruce OX2 Vs. Elite 3 cruce MOC
3. El mejor de 1) Vs. el mejor de 2)

5. Documentación y normativa

En este caso también seguiremos con la misma normativa que ya hemos trabajado en la Práctica 1:

- Solo se admitirá el formato **ZIP**. No se corregirán guiones en cualquier otro formato al indicado.
- La memoria que acompañará en la entrega será exclusivamente en formato PDF y se incluirá una portada con:
 - Identificación de los dos alumnos (Nombre, apellidos y DNI).
 - Identificación del guión.
 - Identificación del grupo de los alumnos.
 - Algoritmos implementados en la práctica.
- Los nombres de los ficheros siguen el mismo formato anterior junto con el código que se detalla más adelante donde:
- Una breve descripción.
- Descripción en **pseudocódigo** de las **estructuras de los genéticos** y de todas aquellas **operaciones relevantes** de cada algoritmo. El pseudocódigo **deberá forzosamente reflejar la implementación y el desarrollo realizados incluyendo los diseños y representación de los modelos evolutivos**, y no ser una descripción genérica extraída de las transparencias de clase o de cualquier otra fuente. La descripción de cada algoritmo no deberá ocupar más de **2 páginas**.

- Breve explicación del **procedimiento considerado para desarrollar la práctica**: implementación a partir del código proporcionado en prácticas o a partir de cualquier otro, o uso de un framework de metaheurísticas concreto. Inclusión de un pequeño **manual de usuario describiendo el proceso para que el profesor de prácticas pueda replicarlo**.
- Experimentos y análisis de resultados:
 - Descripción de los valores de los parámetros considerados en las ejecuciones de cada algoritmo (**incluyendo las semillas utilizadas**).
 - Resultados obtenidos según el formato especificado.
 - Análisis de resultados. El análisis deberá estar orientado a **justificar** (según el comportamiento de cada algoritmo) **los resultados** obtenidos en lugar de realizar una mera “lectura” de las tablas. Se valorará la inclusión de otros elementos de comparación tales como gráficas de convergencia, boxplots, análisis comparativo de las soluciones obtenidas, representación gráfica de las soluciones, etc.
 - Análisis de tiempos y desviaciones, e influencia del tamaño de la instancia en los resultados obtenidos, etc.
 - Por último, es clave obtener un fichero de registro que vaya almacenando la información de la ejecución del algoritmo indicando las soluciones, en qué generación fueron creadas y costes que va obteniendo en las distintas iteraciones los mejores de cada generación.
- Referencias bibliográficas u otro tipo de material distinto del proporcionado en la asignatura que se haya consultado para realizar la práctica (en caso de haberlo hecho).

Aunque lo esencial es el contenido, también debe cuidarse la presentación y la redacción. **La documentación nunca deberá incluir listado total o parcial del código fuente.**

En lo referente al **desarrollo de la práctica**, se entregará una carpeta llamada **software** que contenga una versión ejecutable de los programas desarrollados, así como los ficheros de datos de los casos del problema y el código fuente implementado o los ficheros de configuración del framework empleado. El código fuente o los ficheros de configuración se organizarán en la estructura de directorios que sea necesaria y deberán colgar del directorio FUENTES en el raíz. Junto con el código fuente, hay que incluir los ficheros necesarios para construir los ejecutables según el entorno de desarrollo empleado (tales como *.prj, makefile, *.ide, etc.). La versión ejecutable de los programas y los ficheros de datos se incluirán en un subdirectorio del raíz de nombre BIN. En este mismo directorio se adjuntará un pequeño fichero de texto de nombre LEEME que contendrá breves reseñas sobre cada fichero incluido en el directorio. Es

importante que los programas realizados puedan leer los valores de los parámetros de los algoritmos desde fichero, es decir, que no tengan que ser recompilados para cambiar éstos ante una nueva ejecución. Por ejemplo, la semilla que inicializa la secuencia pseudoaleatoria debería poder especificarse como un parámetro más.

La entrega se llevará a cabo a través de la actividad correspondiente en ILIAS. El plazo de entrega termina el 22 de noviembre de 2019 a las 23:59.

En caso de que el comportamiento del algoritmo en la versión implementada/desarrollada no coincida con la descripción en pseudocódigo o no incorpore las componentes requeridas, se podría reducir hasta en un 50% la calificación del algoritmo correspondiente.