

Trabajo Final
Manual de usuario
Teoría de autómatas y lenguajes formales
Universidad de Cádiz

June 1, 2021

Alumnos: Alejandro Serrano Fernández
Pedro Antonio Navas Luque

Contents

1	Descripción del lenguaje	3
2	Uso del lenguaje	3
2.1	Configuración	3
2.1.1	CROSSOVER	3
2.1.2	MUTATION	3
2.1.3	REPLACEMENT	4
2.1.4	SELECTION	4
2.1.5	POBLATION	4
2.1.6	FITNESS	4
2.2	Política de intercambios	4
2.2.1	GENERATIONS	5
2.2.2	INTERCHANGE	5
2.2.3	SEND	5
2.2.4	RECEIVE	5
2.3	Ejemplo de Programa completo	5
2.4	Metodo de entrada	6

1 Descripción del lenguaje

El lenguaje desarrollado está en inglés y su cometido es la configuración de algoritmos genéticos.

2 Uso del lenguaje

Para comenzar nuestra configuración deberemos escribir "BEGIN" y para finalizarla "END". Entre estos dos términos escribiremos el identificador del algoritmo genético y entre corchetes su configuración, compuesta por:

- a. CROSSOVER
- b. MUTATION
- c. REPLACEMENT
- d. SELECTION
- e. POBLATION
- f. FITNESS

El orden en el que se declaren es irrelevante, pero no debe faltar ninguno de estos términos pues el programa devolverá un mensaje de error con el término faltante o error y su localización. Es importante que todas acaben en punto y coma.

2.1 Configuración

2.1.1 CROSSOVER

El crossover deberá venir acompañado de un entero solamente, lo que indicará que es crossover de un punto, o por dos enteros separados por "|", lo que indicará que es crossover de dos puntos. En este último caso el primer entero deberá ser menor que el segundo.

Ejemplo: CROSSOVER 12;

2.1.2 MUTATION

Para declarar la mutación deberemos indicar el tipo de mutación, los cuales son "BITFLIP" o "POLYNOMIAL" y un entero que indicará la posición a mutar.

Ejemplo: MUTATION BITFLIP 6;

2.1.3 REPLACEMENT

Para declarar el remplazo simplemente deberemos indicar el enfoque, que será o bien "WORST" o bien "RANDOM". Remplazo por el "peor" individuo o por uno "aleatorio".

Ejemplo: REPLACEMENT WORST;

2.1.4 SELECTION

En la selección deberemos indicar el tipo: "RANKING", "ROULETTE" o "TOURNAMENT". Si se elige este último, además hay que indicar un entero que será el tamaño de los agrupamientos en el torneo.

Ejemplo: SELECTION TOURNAMENT 5;

2.1.5 POBLATION

Para la población simplemente deberemos indicar el tamaño de la misma con un entero.

Ejemplo: POBLATION 150;

2.1.6 FITNESS

Y para el fitness podremos definir una función matemática simple que deberá de ir entre paréntesis.

Ejemplo: FITNESS (2x+15);

2.2 Política de intercambios

Después de introducir las configuraciones de los algoritmos genéticos debemos escribir un apartado para indicar como interactúan entre si los algoritmos genéticos. Cabe destacar que este apartado es común para todos los algoritmos declarados ya que hemos supuesto que todos están conectados con todos.

Ahora bien, esta configuración deberá estar situado después del "END" e iniciado por un "POLICY_BEGIN" y finalizado por "POLICY_END", además de ir entre corchetes.

Esta configuración deberá estar compuesta por:

- a. GENERATIONS
- b. INTERCHANGE
- c. SEND
- d. RECEIVE

Al igual que para la configuración individual, el orden es irrelevante, no debe faltar ninguno y deben acabar en punto y coma.

2.2.1 GENERATIONS

Generations ira acompañado de un entero, el cual indicará cada cuantas generaciones se produce un intercambio.

Ejemplo: GENERATIONS 5;

2.2.2 INTERCHANGE

Interchange también ira acompañado de un entero, indicando así cuantos individuos intercambiará el/los algoritmo/s.

Ejemplo: INTERCHANGE 15;

2.2.3 SEND

Para el send hay que indicar la política que se seguirá para seleccionar los individuos a enviar, las dos opciones son los mejores "BEST" o aleatorios "RANDOM"

Ejemplo: SEND BEST;

2.2.4 RECEIVE

Para el receive hay que indicar la política que se seguirá para incluir los individuos recibidos, sustituyéndolos por los peores "WORST" o aleatorios "RANDOM"

Ejemplo: RECEIVE WORST;

2.3 Ejemplo de Programa completo

Una vez descrito como debe usarse el lenguaje y sus componentes vemos un ejemplo de un programa completo con varias configuraciones de algoritmos genéticos (AG y AG2).

```
BEGIN
AG
[
    CROSSOVER 10 | 90;
    MUTATION BITFLIP 20;
    REPLACEMENT WORST;
    SELECTION TOURNAMENT 5;
    POBLATION 1000;
```

```

        FITNESS (100-7x);
    ]
AG2
[
    REPLACEMENT WORST;
    Crossover 50;
    POBLATION 1000;
    MUTATION BITFLIP 40;
    SELECTION TOURNAMENT 7;
    FITNESS (654894/5x*5+98);
]
END

POLICY_BEGIN
[
    GENERATIONS 10;
    INTERCHANGE 20;
    SEND BEST;
    RECEIVE WORST;
]
POLICY_END

```

2.4 Metodo de entrada

El código del programa deberá ir en un fichero txt llamado "code".