Computação Evolucionária

2013/2014

Notes for the PL 6

Ernesto J. F. Costa

15 de Março de 2014

Class PL 6

Evolutionary Algorithms: design issues

6.1 Introduction

We discussed in class the structure of an evolutionary algorithm, whose main components can be viewed in figure 6.1.

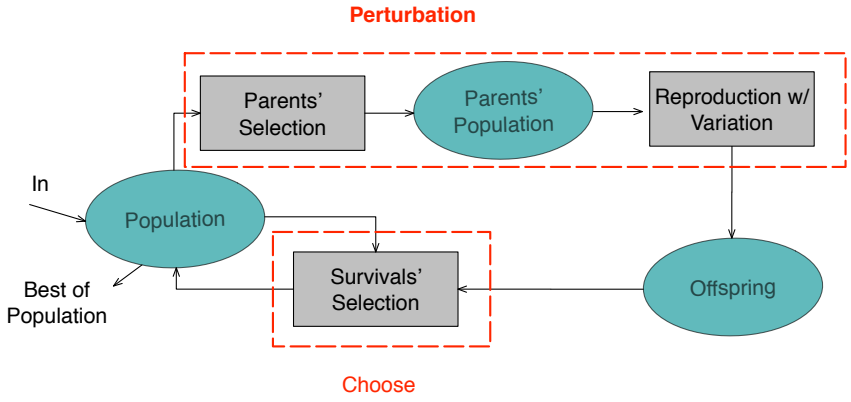

Figura 6.1: Evolutionary Algorithm: general view

The design issues we face when applying this model to a concrete problem are **how to**:

- Choose a **representation** for the candidate solutions
- Define an appropriated **fitness** function
- Choose the **variation** operators (mutation and crossover)

- Choose the **selection** mechanisms (parents and survivors)
- Choose the **stop criterion**
- Define the **initial population**

During the lectures and lab classes we have been discussing these aspects, but the alternatives that we explored to each of these aspects were, in general, limited. It is now time to extend our knowledge regarding these design problems, and see how the options available may, or may not, influence the quality of the final result. We will be using two problems that we already presented: finding the optimum of the Rastrigin Function and the Traveling Salesman Problerm (TSP). Even if we could address all the design issues for each of the two problems, we will split them and use one of the two problems for each subset. In the case of the Rastrigin Function we will be focused on the problems of the selection mechanisms (parents and survivors' selection).In the case of the TSP we will be concentrated on the representation and the variation operators.

6.2 Warmup

We will start by implementing what we need for achieving the goal of the work. You can used the code presented in the class or implement your own versions. It is yup to you, but remember that the best way to learn is to do it.

Problema 6.1 F

Implement the following parents' selection mechanisms:

- Tournament (Deterministic)
- Roulette Wheel
- Stochastic Universal Sampling

Problema 6.2 F

Implement the following survivors' selection mechanisms.

- Generational
- Generational with Elitism
- Steady State

Problema 6.3 F

Implement a function that decode a vector of floats into a permutation of integers, as explained in class, that we need when using a random keys representation of a permutation.

Problema 6.4 M

Implement the following crossover variation operators.

- Partially Mapped Crossover (PMX)
- Order Crossover
- Cycle Crossover

6.3 Benchmark Problems

We are going to review two problems, that we already discussed, i.e., the Rastrigin Function and the Traveling Salesman Problem. A brief description of each problem follows.

6.3.1 Rastrigin Function

The Rastrigin function is an Highly multimodal with many local minima (see figure 6.2 for a 2D version). It is defined by the equation 6.1.

$$f(\langle x_1, \ldots, x_n \rangle) = A \times n + (\sum_{i=1}^{n} x_i^2 - A \times cos(2\pi x_i) \qquad x_i \in [-5.12, 5.12] \quad (6.1)$$

with A typically equal to 10. The global minimum is zero, at point zero:

$$Min(Rastrigin) = Rastrigin(0, \ldots, 0) = 0$$

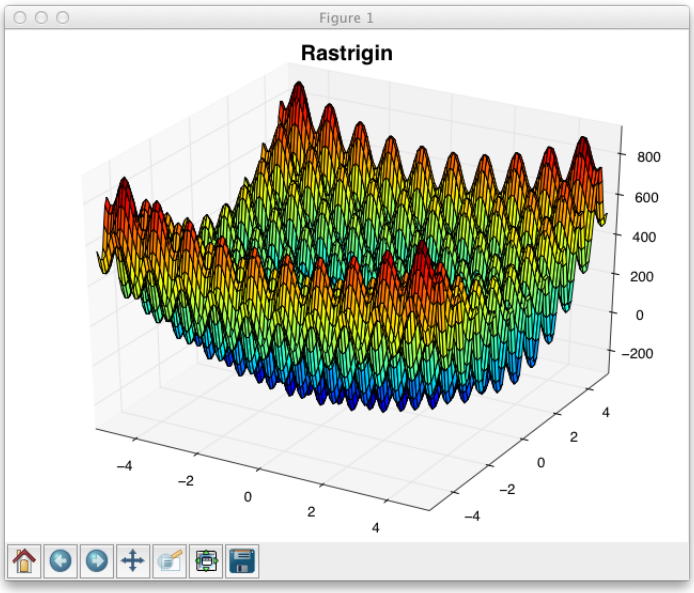

Figura 6.2: **Rastrigin**

6.3.2 Traveling Salesman Problem

We have talked already about the Traveling Salesman Problem (TSP): a person that has to visit several cities starting from his home town, one and just once, and then return to his/her starting city,. The tour must have a minimum (distance) cost. You are going to need a file with the description of a concrete example. There are some examples in **Inforestudante**. Notice that these files have a specific **format** as it is illustrated in the figure 6.3.

For the purpose of the experiments described below, the only relevant part are the 2D coordinates of each city.

6.4 Experiments

For each of the experiments below you should run the algorithms several times. Due to time constraints it is not possible, in principle, to repeat the experiments a convenient number of times, i.e., 30 or more. Try to adapt this number to the available time. At home you can repeat the experiments using a more appropriate number of runs. During each run you are going to collect the value of the **best** result **at the end** of the run, and the **average fitness of the population** (again at the end of the run). You should plot these results and draw your conclusions.

```
NAME: berlin52
TYPE: TSP
COMMENT: 52 locations in Berlin (Groetschel)
DIMENSION: 52
EDGE_WEIGHT_TYPE: EUC_2D
NODE_COORD_SECTION
1 565.0 575.0
2 25.0 185.0
3 345.0 750.0
4 945.0 685.0
...
EOF
```

Figura 6.3: Example of the format of a tsp file. You may have more than one COMMENT line.

6.4.1 Experiment 1

In this experiment we will be using the Rastrigin Function with 10 dimensions.The goal is to study the influence of the **parents**' selection mechanisms on the quality of the final result. Develop three versions of a simple Evolutionary Algorithm, one using tournament selection, another using roulette wheel selection and a third one using the stochastic universal sampling selection. Compare the three versions using the best at the end and the average fitness at the end of each run. Collect also the **number of fitness evaluations** needed to attain the desired result. Draw your conclusions.

6.4.2 Experiment 2

In this experiment we will be using the Rastrigin Function with 10 dimensions.The goal is to study the influence of the **survivors**' selection mechanisms on the quality of the final result. Develop three versions of a simple Evolutionary Algorithm, one using a generational selection, another using generational selection with elitism and a third one using a steady state selection. Compare the three versions using the best at the end and the average fitness at the end of each run. Collect also the **number of fitness evaluations** needed to attain the desired result. Draw your conclusions.

6.4.3 Experiment 3

In this experiment we will be using the TSP as benchmark.The concrete instances will be *berlin52.tsp* and/or *wi29.tsp*. The goal is to study the influence of the **crossover** variation operators on the quality of the final result. Develop there versions of a simple Evolutionary Algorithm, one using the PMX operator, another using Order Crossover and a third one using the Cycle Crossover. Compare the three versions.

6.4.4 Experiment 4

In this experiment we will be using the TSP as benchmark.The concrete instances will be *berlin52.tsp* and/or *wi29.tsp*. The goal is to study the influence of the **representation** quality of the final result. Implement a solution to the problem based on the **random keys**, run several times for that representation, and compare the results with the best ones obtained in experiment 3.