

# Introduction to genetic algorithms

## Insights for IA

TEAM 2

Facultad de Ingenieria  
Pontificia Universidad Javeriana

# Background

Inspired in the evolution theory of Darwin.

# Why GA?

Numerical methods are based in a set of necessary and sufficient conditions, and assess point to point, genetic algorithm is a multipoint guided random search technique, we don't need additional information about the problem.

# Quadratic function

$$f(x) = ax^2 + bx + c \quad (1)$$

In an analytical way it is enough  $f'(x^*) = 0$ .

$$x^* = \frac{-b}{2a} \quad (2)$$

the exact solution. Another way of finding the solution is using gradient descent, that could climb the curve to reach the point.

# Optimal ilustration

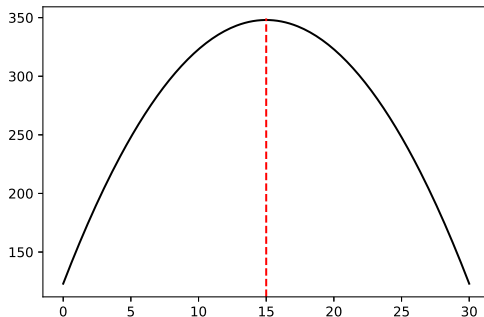


Figure: Quadratic function,  $x^* = 15$

# Two possible candidates

Swap information

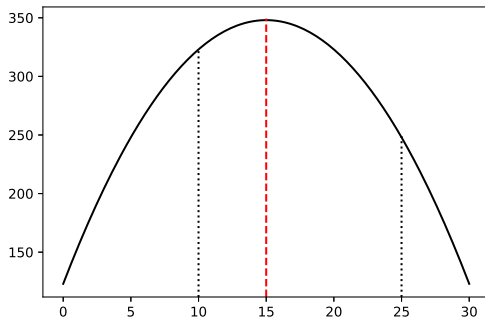


Figure: Quadratic function,  $x^* = 15$

# Swap information

We could represent the candidates in a iterable object and swap the digits.  
for instance:

---

```
x,y = [1,0],[2,5]
x,y = swap(x,y) # Interchange the numbers
x,y = number(x,y) # pass the output arrays to real numbers
```

---

Now we can transform to assess the two points in the function:

$$f'(15) = 0, f'(20) \neq 0 \quad (3)$$

The solution was contained in the candidates!

# Introduction to genetic algorithm

A GA is a program, composed of strings that swap information!. A **Schema** theorem give us a cornerstone to understand how they works!



# Chromosome, gen, allele and locus

Figure: Chromosome representation



The chromosome is genotype and the candidate solution is phenotype, the before example  $x = 15$ .

Individuals with better features have major probability of have offspring.

# Swaping operators

The selection provided the parents that will be interchange information, with the crossover and mutation operator.

# Genetic algorithm

## Implementation

---

### Algorithm 1: Canonical genetic algorithm

---

initialization of population;

**while** *not fill stoping criteria* **do**

    Select Parents ;

    Crossover pairs of parents ;

    Mutate Offspring ;

    Update population ;

**end**

---

# Schema

Constructed over the alphabet  $\{0, 1, *\}$  it is an abstract representation of a chromosome. Where  $*$  is metasyMBOL (taking either 1 or 0).

# Schema

If we have the following schema \* \* \* we could represent  $2^3$  possible instances, for example 111 or 100 and so on. 10101 could be represented by:

- 10101
- \*0101
- 1\*101
- \*\*101
- 1\*\*01
- \*\*\*01
- \*\*\*\*\*

in a total of  $\sum_{i=0}^k \binom{n}{i} = 2^n$  where  $k$  is the number of wildcards.

# Schema properties

- Order:  $o(H)$  the number of fixed position; or the length of the schema minus the number of  $*$ 's.
- Length:  $l(H)$  is the number of bits that composed H.
- Defining length:  $\delta(H)$  The distance among the first and last fixed position.

# Inheritance

Features that remain of parents to offsprings.



# Selection

## Proportional selection

**Selection** determines what parents will be used to create offspring.  
Select possible chromosomes as parents, according its fitness.

$$P(x_i) = \frac{f(x_i)}{\sum_j^n f(x_j)} \quad (4)$$

after of get we can choose a random number among 0 and the sum of total fitness in population, this strategy is denominated wheel strategy.

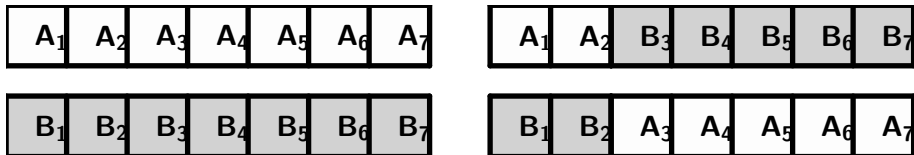
# Mutation and crossover

The crossover and mutation as operators of the evolutionary process, in the search process, one is exploration and the another exploitation of search space.

# Crossover

fix one point

Figure: Crossover in single point (2-index)

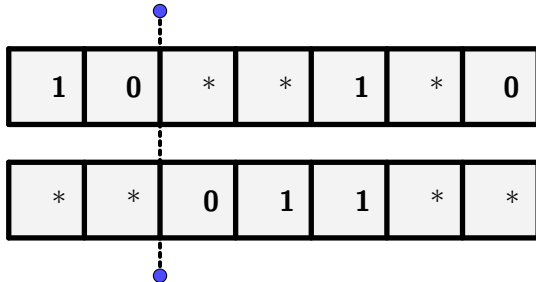


the offspring born choosing a random locus, in two parents, and after concatenate  $\alpha_a$  and  $\alpha_b$  and the locus is equal to  $\pi$  then the offspring will be  $\alpha_a[:\pi]$  concatenate with  $\alpha_b[\pi:]$ .

# Crossover

## Vulnerability-compactibility

Figure: Crossover in single point (2-index)



When crossover is within defining length, the schema have a lower probability of survive, quantifying  $\frac{\delta(H)}{l(H)-1}$ .

# Crossover-inheritance

We can define the probability of that the crossover occur among two parents  $P_c$ , and the probability of a schema will be destroyed as  $p_d$  (related with the vulnerability) We can define that the probability of a schema will be destroyed by  $P_c * P_{dl}$

$$P_s = 1 - P_c \frac{\delta(H)}{l(H) - 1} \quad (5)$$

Therefore the probability that a schema survive will be  $1 - P_d$ .

# Mutation-inheritance

All bits in a string have  $P_{mu}$  of undergoes a mutation, therefore the probability of not mutate is  $1 - P_{mu}$  then for the number of fixed bits we have  $o(H)$  that the probability of complete schema survival to mutation will be

$$P_m = (1 - P_{mu})^{o(H)} \quad (6)$$

# Crossover-Mutation-Inheritance

The probability of a schema survive to crossover and mutation will be:

$$P_{sur} = [1 - p_c \frac{\delta(H)}{L(H) - 1}] (1 - p_{mu})^{o(H)} \quad (7)$$

# Schema theorem

## Inheritance in generations

First we need take in mind the settings:

- Binary codification
- $N_{t-1} = 0$  (There are not parents in the next generation)
- Roulette wheel (proportional selection)
- Crossover in a single point
- Uniform mutation



# Schema theorem

Population(t)	Fitness $f(x)$
$x_1 = 1010$	$f(x_1)$
$x_2 = 0101$	$f(x_2)$
$\vdots$	$\vdots$
$x_n = 1110$	$f(x_n)$
$\sum_i^n f(x_i) = F(t)$	

We can ranking that  $\frac{f(x_i)}{F(t)} > \dots > \frac{f(x_j)}{F(t)}$  and select pairs to crossover.

# Schema theorem

$m(H, t)$  number of strings that match with  $H$  schema in  $t$  generation.

Chromosome	Schema ( $H = 1*01$ )	$f(H, t)$
$x_1 = 1011$	0	-
$x_2 = 1001$	1	$f(x_2)$
$x_3 = 0011$	0	-
$x_4 = 1101$	1	$f(x_4)$
$m(H, t) = 2$		$\frac{f(x_2) + f(x_4)}{2}$

# Schema theorem

## Goal

Determine the **probability** that  $H$  survive in the evolutionary process.  
The average probability of individuals that  $Match(H, x_i) = 1$  be selected in the  $t$ -generation to crossover is :

$$\frac{f(H, t)}{F(t)} \quad (8)$$

# Schema theorem

if for instance we have  $n$  individuals in the population, the probability of my first parent will be a *match* will be  $m(H, t) \frac{f(H, t)}{F(t)}$  and therefore for the  $n$  parents selections we have:

$$m(H, t) \frac{f(H, t)}{F(t)} (n) \quad (9)$$

will be the number of individuals that match that will be selected.

# Schema theorem

In the next generation:

$$m(H, t + 1) = m(H, t) \frac{f(H, t)}{\bar{F}(t)} (n) \quad (10)$$

The average fitness of the population will be  $\bar{F}(t) = \frac{F(t)}{n}$  rewriting we have:

$$m(H, t + 1) = m(H, t) \frac{f(H, t)}{\bar{F}(t)} \quad (11)$$

This means that schemas with higher fitness of average tend to be selected more.

# Schema theorem

## Lower-bound

Why is a low bound?

$$E[m(H, t + 1)] \geq m(H, t) \frac{f(H, t)}{\bar{F}(t)} \left[1 - p_c \frac{\delta(H)}{L(H) - 1}\right] (1 - p_{mu})^{o(H)} \quad (12)$$

# Building Block Hypothesis

BBH

Now define the rate of growth as  $\gamma = \frac{f(H,t)}{\bar{F}(t)} [1 - p_c \frac{\delta(H)}{L(H)-1}] (1 - p_{mu})^{o(H)}$ .

$$E(m(H, t)) \geq m(H, t) \gamma \quad (13)$$

if  $\gamma \geq 1$  then  $H$  is a **Building block**.

# Exponential growth of schemata

$$\begin{aligned}\frac{dm(H, t)}{dt} &= \gamma m(H, t) \\ \frac{1}{m(H, t)} dm(H, t) &= \gamma dt \\ \int_0^t \frac{1}{m(H, t)} dm(H, t) &= \int_0^t \gamma dt\end{aligned}\tag{14}$$

Remember that  $\frac{d \ln(u)}{du} = \frac{1}{u}$ , also that that  $\gamma t - \gamma 0 = \gamma t$ .

$$\ln m(H, t) - \ln m(H, 0) = \gamma t\tag{15}$$



# Exponential growth of schemata

According to the last expression we have;

$$\ln\left(\frac{m(H, t)}{m(H, 0)}\right) = \gamma t \quad (16)$$

Now we can see that:

$$m(H, t) = e^{\gamma t} m(H, 0) \quad (17)$$

The number of **instances** of  $H$  increase exponentially. According to **BBH** GA's search the optimal solution with building blocks and its crossover and mutation.

# Deception

**BBH** could be loose the optimal solution!, rejecting solutions that seems not good, the increase of homogeneity of solution, could lead to lose important genetic material.

# Selection

## Roulette wheel bias

The main problem with proportional selection is the sensitivity to outliers values. Could be exist a bias due a higher fitness value in the population, that lead to premature convergence.

# Tournament selection

we have a population of  $n$  individuals;  $x_1, x_2, x_3, \dots, x_n$ . We sample pairs of individuals.

Sampled	Fitness	Winner
$(x_3, x_2)$	$f(x_3) > f(x_2)$	$x_3$
$(x_1, x_1)$	$f(x_1) = f(x_1)$	$x_1$
$(x_5, x_6)$	$f(x_5) < f(x_6)$	$x_6$
$(x_n, x_8)$	$f(x_n) < f(x_8)$	$x_8$
$(x_3, x_1, x_6, x_8)$		

Take in mind, the individuals that *Not being sampled* ( $x_j : j \neq 3, 2, 1, 5, 6, 8, n$ ), and *Not being selected* ( $x_2, x_5, x_n$ ).

# Loss of diversity

Motoki (2002)

Selection schemes must tackle loss of diversity, **not being sampled** and **not being selected** introduce the bias in tournament selection.

$$Loss(t, n) = \frac{1}{n} \sum_{k=1}^N \left(1 - \frac{k^t - (k-1)^t}{n^t}\right)^n \quad (18)$$

where  $n$  is population size, and  $k$  is the tournament size, Where  $k$  increase the loss of diversity also increase.

# Selection

## Tournament selection

Given that is a random process, the selection of the better individuals not is guaranteed.

The size of the tournament( $k$ ) in each tournament will be select the individual with major fitness, to crossover.

# Tournament selection

## Insights

When  $k$  tournament size increase the probability of the individuals with less fitness is low.

# Implicit parallelism

Process  $3^n$  schemate, with only  $n$  as input.



# Mutation

# Mutation

## Shuffle indices

# genetic TSP

# Elitism process

$\mu + \lambda$  algorithm

# Genetic algorithms in ML

Here there are important remarks are used to avoid greedy algorithms for instance:

- Decision trees
- K-means
- Tuning hyperparameters and topology design in Neural Networks.

# Memetic algorithms

# References

# Following techniques to research

Simulated annealing, tabú search... ant PSO and ant-colony.



# References

- Blickle, T., Thiele, L. (1995, July). A Mathematical Analysis of Tournament Selection. In ICGA (Vol. 95, pp. 9-15).
- Motoki, T. (2002). Calculating the expected loss of diversity of selection schemes. Evolutionary Computation, 10(4), 397-422.

