

# In Vitro Fertilization

## Genetic Algorithm



INSTITUTO DE  
INFORMÁTICA  
UFG





“

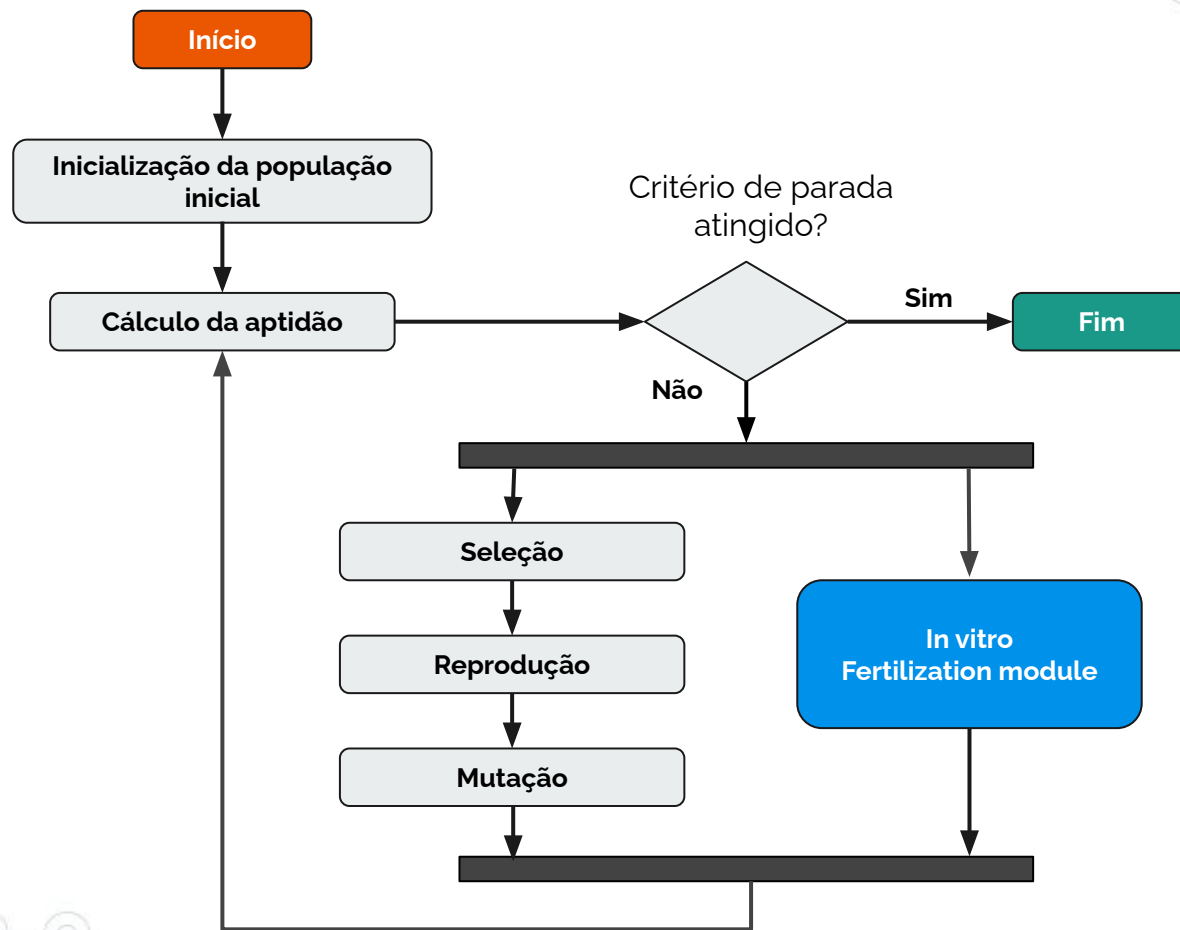
*Módulo de manipulação de informações  
genética paralelo ao fluxo principal do  
Algoritmo Genético*

Inspirado na geração de “bebês de proveta” por Fertilização In Vitro e manipulação genética.

## Caracterização

- ⊙ Proposto por Camilo-Junior e Yamanaka em 2011;
- ⊙ Inspirado na geração de “bebês de proveta” por Fertilização In Vitro e manipulação genética;
- ⊙ Atenuar a perda de material genético promissor ao longo das gerações;
- ⊙ Fluxo paralelo aos operadores genéticos do AG;
- ⊙ Recebe uma porção da população do AG e (eventualmente) devolve um indivíduo melhor do que o “melhor corrente”;

# Fluxograma



## Procedimento do método *In Vitro*

1. Definir o melhor indivíduo como “pai”;
2. Define a quantidade de indivíduos vindos do “host”;
3. Na estratégia EAR dos N indivíduos metade tem os cromossomos alterados gerando N'. Na estratégia AR  $N' = N$ ;
4. O pai é re combinado com os N' indivíduos;
5. Se houver filhos melhores do que o pai, insere-se os na população do AG.

## Duas diferentes estratégias

### ◎ **AR** - Assisted Recombination

- Realiza apenas a **recombinação** do material genético contido na porção da população que foi selecionada para o módulo;
- Capacidade de manter um material genético de boa qualidade ao longo da evolução;

### ◎ **EAR** - Exploratory Assisted Recombination

- Possui duas fases;
- Na primeira ocorre a **exploração** do espaço de busca através da perturbação nos genes de parte dos indivíduos vindos da população;
- Na segunda fase ocorre a **Recombinação**.

# Estratégias EAR

## ◎ EAR-P

- Realiza mutação em parte dos genes de metade das mães;

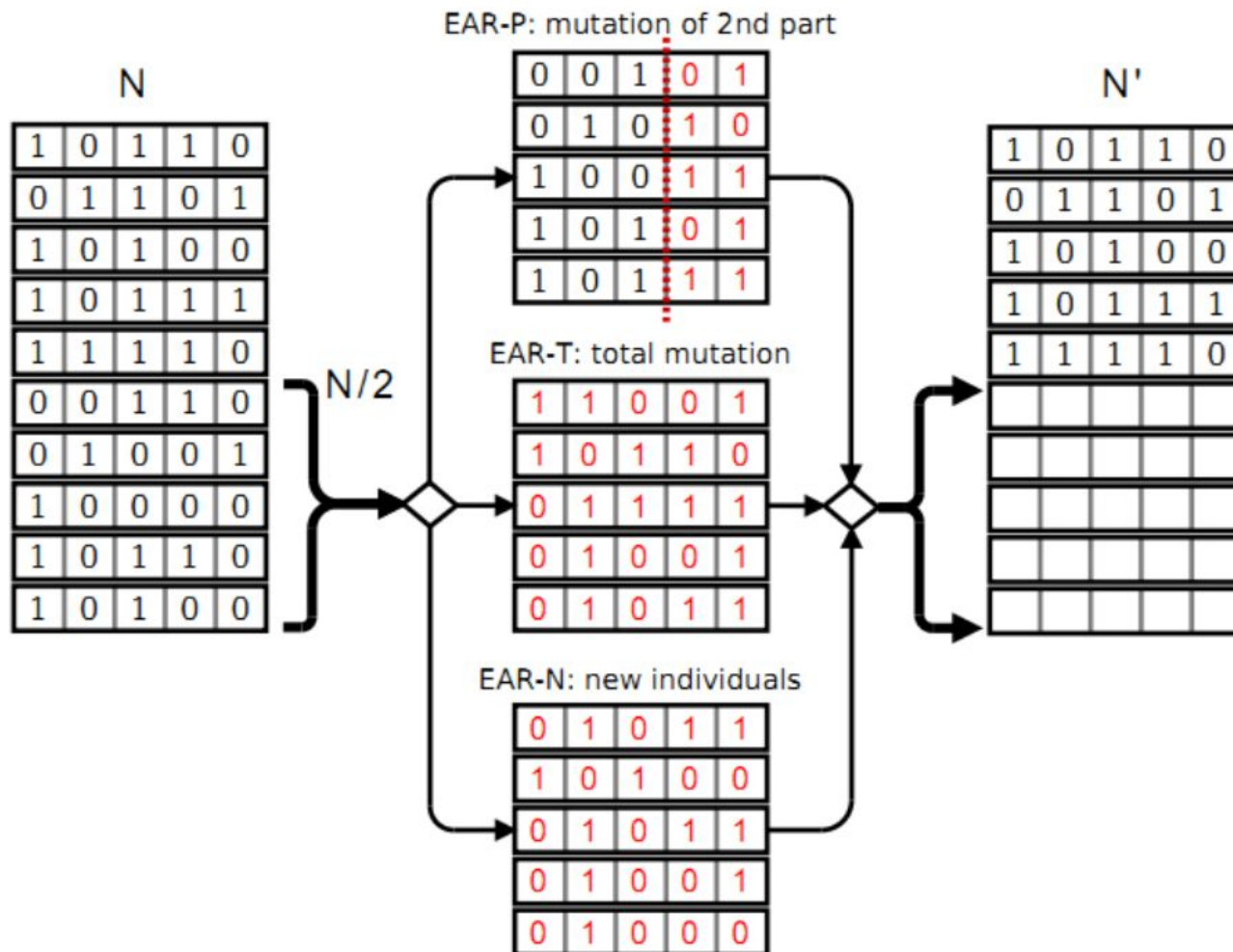
## ◎ EAR-T

- Realiza mutação em todos os genes de metade das mães;

## ◎ EAR-N

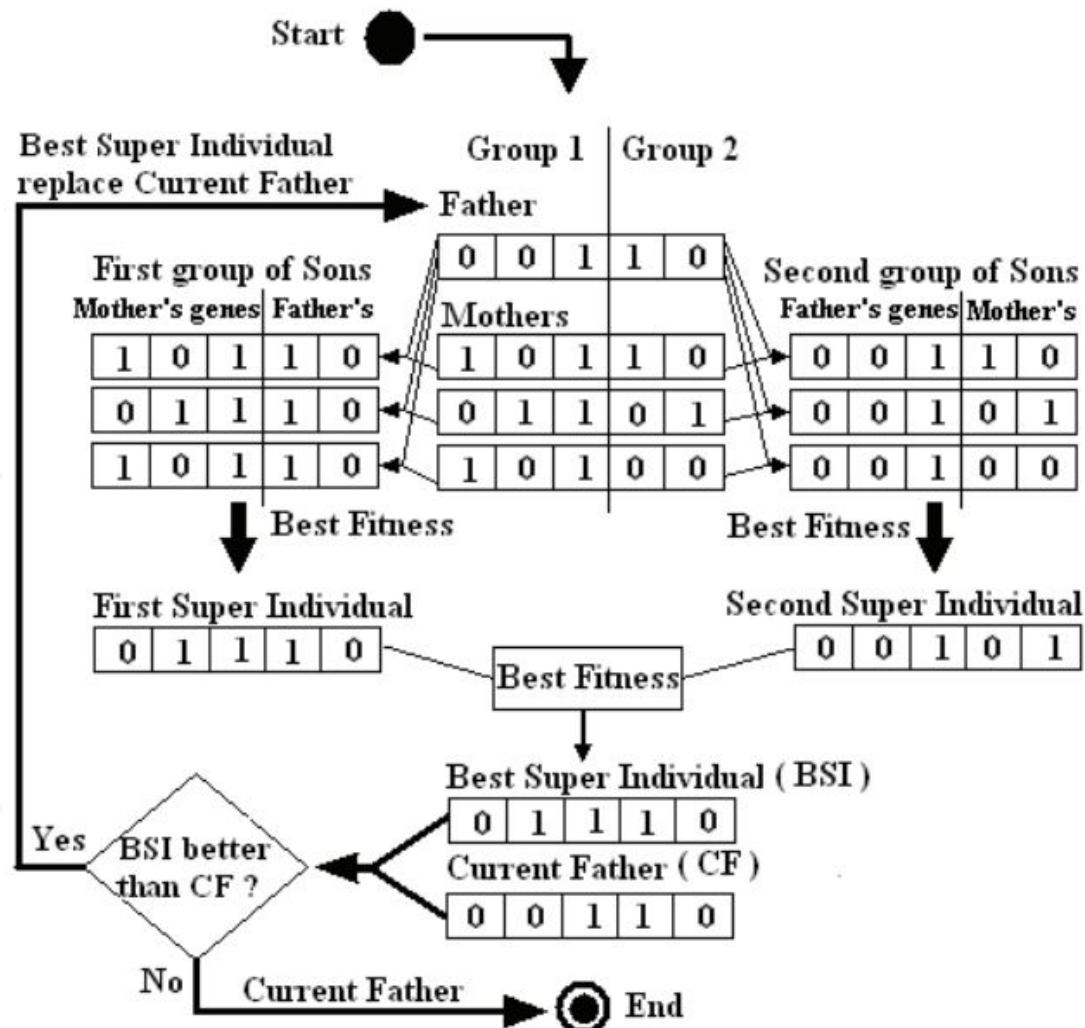
- Remove metade das mães e as substitui por novas mães aleatoriamente geradas

# Estratégias EAR\*

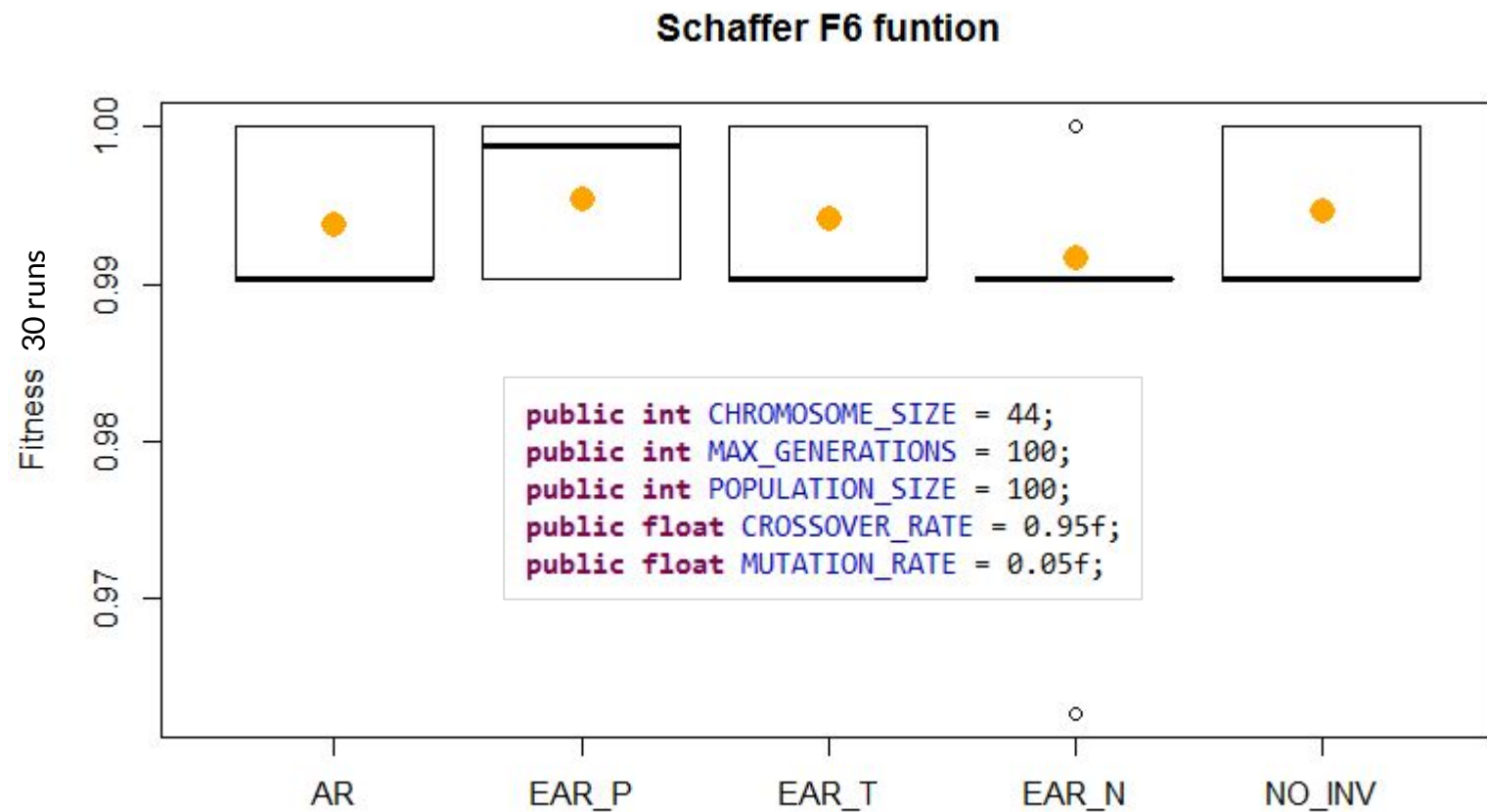




# Exemplo do processo de recombinação

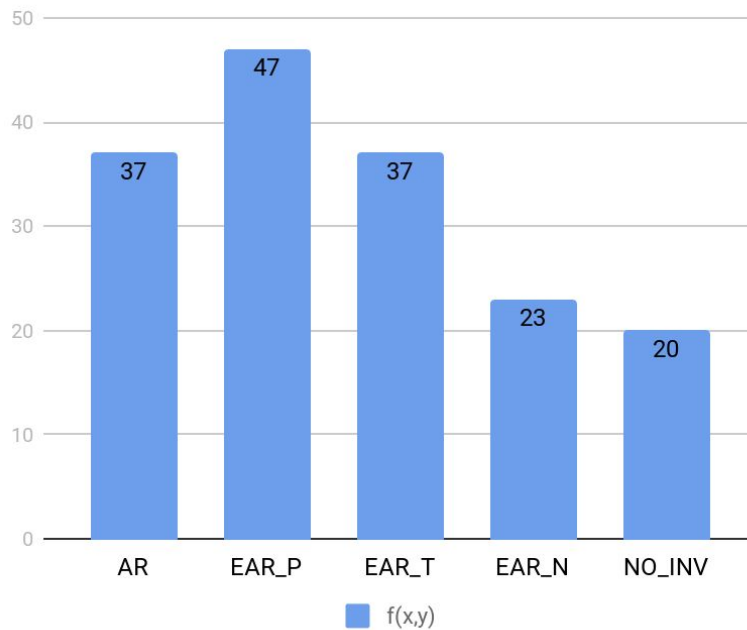


# Testando os operadores

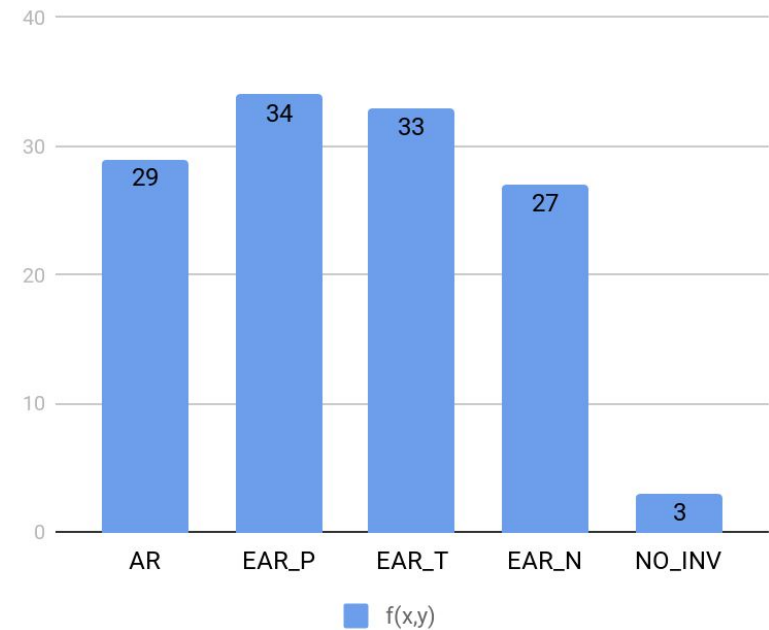


# Testando os operadores

5% mut, 90% cros, 100 ind, 100 gen, 100 run



8% mut, 65% cros, 100 ind, 100 gen, 100 run



## In Vitro Fertilization Genetic Algorithm

Celso G. Camilo-Junior<sup>1</sup> and Keiji Yamanaka<sup>2</sup>

<sup>1</sup>*Informatics Institute, Federal University of Goiás*

<sup>2</sup>*School of Electrical Engineering, Federal University of Uberlandia  
Brazil*

### 1. Introduction

Some techniques are applied to the optimization problems. However, a few achieve satisfactory performance when the problem is complex, for example, multimodal or multiobjective. The Mathematical Programming algorithms which use the gradient as a search guide have difficulty and do not often reach the optimum in multimodal problems. The metaheuristics, on the other hand, don't ensure the global optimum but they have good results and, therefore, are quite used in these scenarios.

Metaheuristics could be classified by the number of potential solutions that are used: solution-based metaheuristics, such as Hill-Climbing, Simulated Annealing and Tabu search; population-based metaheuristics, such as the Evolutionary Algorithms; or hybrids metaheuristics. The population-based algorithms start the search with several points in the search space and, through the interactions of among points, try to find a new point with the highest value of the objective function. This strategy, therefore, explores the search space in several places simultaneously (Oriented Exploration). The solution-based algorithms are based on a single point to exploit the search space (Oriented Exploitation). They often use techniques to escape from the local optimum. The hybrid strategy tries to join the intensification and diversification to improve both quality of computed solutions and the robustness of solvers. All strategies show good results depending on the approached problem. As an example of metaheuristic, we can cite: Simulated Annealing, Tabu Search, Grasp,

# Obrigado!

## Perguntas?

**altinoneto@inf.ufg.br**

**I4Soft**

Intelligence for Software