華東師範大學
EAST CHINA NORMAL UNIVERSITY

Memetic Algorithm

文化基因算法

吴婷钰

Outline

- Introduction
- Memetic Algorithm
- Applications

Introduction

Introduction

- Memetic algorithms are optimization techniques based on the synergistic combination of ideas taken from different algorithmic solvers, such as population-based search and local search.

- Gene vs. Meme
  - **Common:** In the process of evolution and development through crossover and mutation operations.
  - **Difference:**
    1. Gene: In biological evolution, **variation is random**, only a few good variation can be retained in natural selection.
    2. Meme: Cultural transmission process often with **full knowledge-based professional fields**, evolution is faster.

Introduction

- Inspired by both Darwinian principles of natural evolution and Dawkins' notion of a meme, the term "Memetic Algorithm"(MA) was introduced by Moscato in 1989 where he viewed MA as being close to a form of population-based hybrid genetic algorithm(GA) coupled with an individual learning procedure capable of performing local refinements.

- In essence, most MAs can be interpreted as a search strategy in which a population of optimizing agents cooperate and compete. The success of MAs can probably be explained as being a direct consequence of the synergy of the different search approaches they incorporate.

Memetic Algorithm

Memetic Algorithm

- **A General Template:**
  - Similar to that of a local search procedure acting on a set of $|pop| \geq 2$ configurations.

Algorithm 2 A population-based search algorithm

```
1  Procedure Population-Based-Search-Engine;
2  begin
3      Initialize pop using GenerateInitialPopulation();
4      repeat
5          newpop ← GenerateNewPopulation(pop);
6          pop ← UpdatePopulation (pop, newpop);
7          if pop has converged then
8              pop ← RestartPopulation(pop);
9          endif
10     until TerminationCriterion() ;
11 end
```

Memetic Algorithm

- **GenerateInitialPopulation procedure:**
  - Generating $|pop|$ random configurations
  - Sophisticated seeding mechanism(for instance, some constructive heuristic)
  - Local-Search-Engine

Algorithm 3 Injecting high-quality solutions in the initial population.

```
1  Procedure GenerateInitialPopulation;
2  begin
3      Initialize pop using EmptyPopulation();
4      for j ← 1 to popsize do
5          i ← GenerateRandomConfiguration();
6          i ← Local-Search-Engine (i);
7          InsertInPopulation individual i to pop;
8      endfor
9      return pop;
10 end
```

Memetic Algorithm

- **TerminationCriterion function :**

  - Setting a limit on the total number of iterations
  - Reaching a maximum number of iterations without improvement
  - Having performed a certain number of population restarts.

Memetic Algorithm

- **GenerateNewPopulation procedure :**

Algorithm 4 The pipelined GenerateNewPopulation procedure.

```
1  Procedure GenerateNewPopulation (pop);
2  begin
3      buffer^0 ← pop;
4      for j ← 1 to n_op do
5          Initialize buffer^j using EmptyPopulation();
6      endfor
7      for j ← 1 to n_op do
8          S^j_par ← ExtractFromBuffer (buffer^(j−1), arity^j_in);
9          S^j_desc ← ApplyOperator (op^j, S^j_par);
10         for z ← 1 to arity^j_out do
11             InsertInPopulation individual S^j_desc[z] to buffer^j;
12         endfor
13     endfor
14     return buffer^(n_op);
15 end
```

Memetic Algorithm

- **UpdatePopulation procedure:**
  - The UpdatePopulation procedure is used to reconstruct the current population using the old population *pop* and the newly generated population *newpop*.
    - the *plus* strategy: the current population is constructed by taking the best *popsize* configurations from *pop* ∪ *newpop*.
    - the *comma* strategy: the best popsize configurations are taken just from *newpop*(|*newpop*| > *popsize*). As a general guideline, it is usually regarded as less prone to stagnation, with the ratio |*newpop*| /*popsize* ≃ 6 being a common choice

Memetic Algorithm

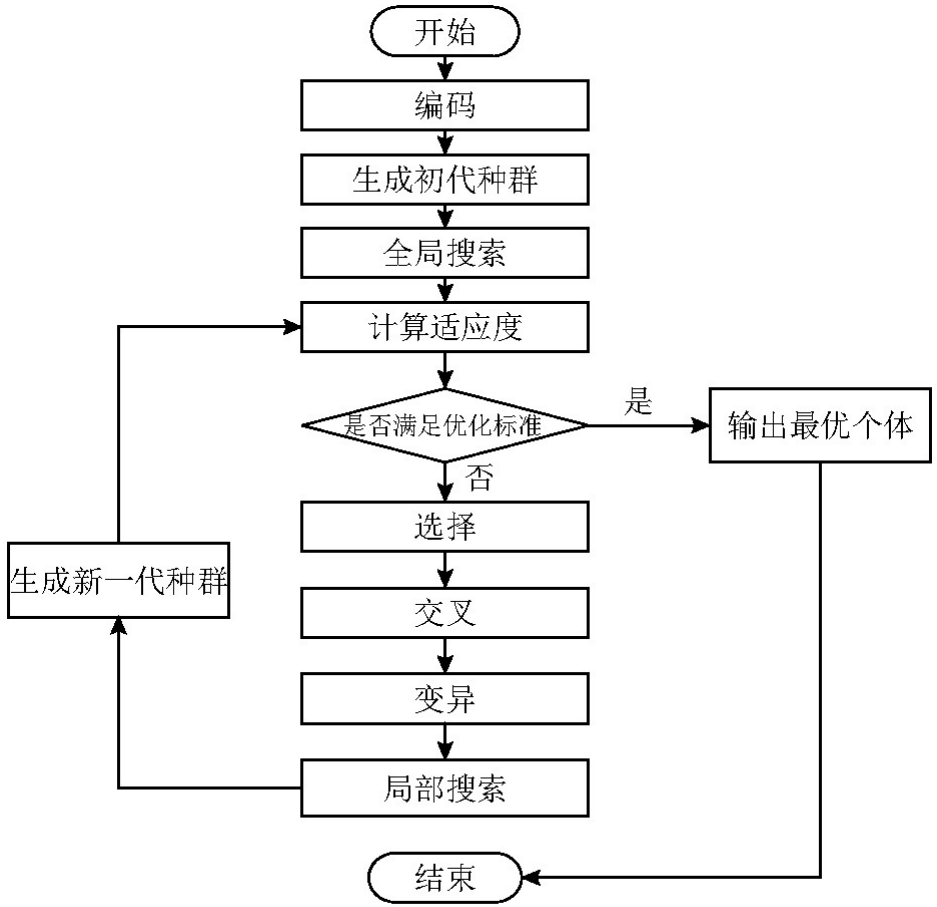
- **RestartPopulation procedure:**
  - *random-immigrant* strategy

Algorithm 5 The RestartPopulation procedure.

```
1  Procedure RestartPopulation (pop);
2  begin
3      Initialize newpop using EmptyPopulation();
4      #preserved ← popsize · %preserve;
5      for j ← 1 to #preserved do
6          i ← ExtractBestFromPopulation(pop);
7          InsertInPopulation individual i to newpop;
8      endfor
9      for j ← #preserved + 1 to popsize do
10         i ← GenerateRandomConfiguration();
11         i ← Local-Search-Engine (i);
12         InsertInPopulation individual i to newpop;
13     endfor
14     return newpop;
15 end
```

Memetic Algorithm

- **Flow Chart**



Memetic Algorithm

- **Conclusion**
  - In fact, MAs is a genetic algorithm framework, is a concept, in this framework, using different search strategies can constitute different MAs.
  - **Global search strategy** can be used genetic algorithms, evolution strategies, evolutionary programing, etc.
  - **Local search strategy** can be used to climb the search, simulated annealing, greedy algorithms, tabu search, guided local search.

Applications

Applications

- Machine learning and knowledge discovery
- Traditional combinatorial optimization
- Planning, scheduling, and timetabling
- Bioinformatics
- Electronics, engineering, and telecommunications

Thank You !