Artificial Intelligence

Evolutionary Algorithms

Lesson 9:
Parallel Evolutionary Algorithms

Vincenzo Piuri

Università degli Studi di Milano

Contents

- Parallelization of evolutionary algorithms
- Which steps can be parallelized
- Island model
- Cellular evolutionary algorithms

Parallelization of Evolutionary Algorithms (1)

Evolutionary Algorithms are computationally-expensive optimization methods

- Large population
  (a few thousand up to several tens of thousands of individuals)
- Large number of generations
  (a couple of hundreds)

- Advantage: better solution quality compared to other approaches
- Disadvantage: long execution time

Parallelization of Evolutionary Algorithms (2)

Parallelization

- Which steps can be parallelized?
- What additional, specialized techniques are inspired by a parallel organization of the algorithm?

Which Steps Can Be Parallelized (1)

Initial population

- Easy to parallelize
  - Usually the chromosomes of the initial population are created randomly and independently
- Attempt to prevent duplicate individuals may pose obstacles to a parallel execution
  - Little importance overall, because the initial population is created just once

Which Steps Can Be Parallelized (2)

Evaluation of chromosomes

- Easily parallelizable
  - Usually an individual is evaluated independently of any other ones

Which Steps Can Be Parallelized (3)

Computing the (relative) **fitness** values or a **ranking** of the individuals

- Central agent that collects and processes evaluations

Which Steps Can Be Parallelized (4)

Selection

- Expected value model, Elitism
  - Require to consider the population as a whole
  - Difficult to parallelize
- Roulette-wheel, Rank-based selection
  - Can be parallelized after the initial step of computing the relative fitness values
- Tournament selection
  - Best suited for a parallel execution
  - All tournaments are independent and thus can be held in parallel

Which Steps Can Be Parallelized (5)

Applying **genetic operators**

- Can be applied in parallel
- They affect only one (mutation) or two chromosomes (crossover)
- Are independent of any other chromosomes

Which Steps Can Be Parallelized (6)

Termination criterion

- Need a central agency that collects information about the individuals
  - Best individual of the population exceeds a user-specified fitness threshold
  - Best individual has not changed (a lot) over a certain number of generations

Island Model (1)

- Selection method that causes some troubles for parallelization
  - Parallel execution by simply processing several independent populations
  - Each population can be seen as inhabiting an island, therefore **island model**
- Pure island
  - Executing the same evolutionary algorithm multiple times
  - Results that are somewhat worse than those of a single run with a larger population

Island Model (2)

Migration

- Exchanging individuals between the island populations at certain fixed points in time
- Not in every generation
- No direct recombination of chromosomes from different islands
- After migration: recombination of genetic material of one island with another

Island Model (3)

Control the migration

- Random model
  - Pairs of islands chosen randomly
  - Exchange some of their inhabitants
  - Any two island can exchange individuals
- Network model
  - Islands are arranged into a network or graph
  - Individuals can migrate only between islands that are connected in the graph
  - Along which of the edges individuals are exchanged is determined randomly

Island Model (4)

- Evolutionary algorithms applied on the islands differ in approaches and/or parameters
- Population size of an island is increased or decreased according to the average fitness of its individuals
- Usually a lower bound for the population size is set

Cellular Evolutionary Algorithms (1)

- Processors are arranged in a rectangular grid

- Selection and crossover are restricted to adjacent processors (connected by edge)

- Mutation restricted to single processors

Cellular Evolutionary Algorithms (2)

- Each processor is responsible for one chromosome
  - Selection: processor chooses the best chromosome of the (four) processors adjacent to it
  - Crossover: processor performs crossover of the selected chromosome with its own and may also mutate the chromosome
    - The better child resulting from such a crossover replaces the chromosome of the processor
    - Groups of adjacent processors that maintain similar chromosomes mitigates the usually destructive effect of the crossover