Nature-Inspired Computing

Evolutionary Programming

Dr. Şima Uyar
September 2006

Overview

- developed in the USA in the 1960's
- typically applied to:
  - traditional EP: machine learning tasks by finite state machines
  - contemporary EP: (numerical) optimization
- features:
  - very open framework: any representation and mutation op's OK
  - crossbred with ES (contemporary EP)
  - consequently: hard to say what "standard" EP is
  - no recombination
  - self-adaptation of parameters standard (contemporary EP)

Overview (current EP)

Representation	Real-valued vectors
Recombination	None
Mutation	Gaussian mutation
Parent selection	Deterministic
Survivor selection	Probabilistic ($\mu+\mu$)
Specialty	Self-adaptation of mutation step sizes (in meta-EP)

Prediction by Finite State Machines

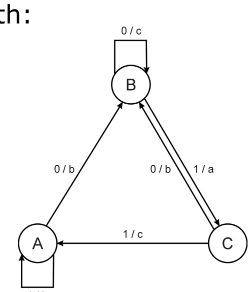
- finite state machine (FSM):
  - states S
  - inputs I
  - outputs O
  - transition function $\delta : S \times I \rightarrow S \times O$
  - transforms input stream into output stream

Prediction by Finite State Machines

- can be used for predictions, e.g. to predict next input symbol in a sequence

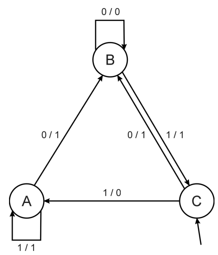
FSM example

- consider the FSM with:
  - S = {A, B, C}
  - I = {0, 1}
  - O = {a, b, c}
  - δ given by a diagram



FSM as Predictor

- consider the following FSM
- task: predict next input
- quality:
  % of $in_{(i+1)} = out_i$
- given initial state C
- input sequence 011101
- leads to output 110111
- quality: 3 out of 5



Modern EP

- no predefined representation
- no predefined mutation (must match representation)
- often applies self-adaptation of mutation parameters