21: Coevolutionary Algorithms

- Motivation
- Competitive coevolution
- Cooperative coevolution
- EC approaches and application examples
- Textbook chapter 15

Prof. Ting Hu, School of Computing, Queen's University

Motivation

- So far: problems with easy-to-measure fitness
- More difficult: fitness depends on <u>context</u>
  - solution represents a strategy that works in opposition to some competitor that is itself adapting, e.g., adversarial game-playing
  - solution being evolved does not represent a complete solution, but can only be evaluated as part of a greater whole, e.g., robotic controllers

Coevolution in nature

- Adaptation of a biological species
  - adaptive value is determined by the evolutionary niche which is determined by the other organisms
    - positive effect (mutualism/symbiosis)
    - negative effect (predation/parasitism)
  - coevolution: the landscape "seen" by each species is affected by the configuration of all other interacting species

Predator-prey relations

- Arms races of both sides, strong evolutionary pressure
- Success on one side is felt by the other side
- How to respond in order to maintain one's chances of survival
- Stepwise increase in complexity of both predator and prey

Competitive coevolution

- Species compete against each other
- Fitness is gained at each other's expense

Lifetime fitness evaluation (LTFE)

- Fitness of a solution is calculated on the basis of the tests it *encounters* during its lifetime
- As the number of tests satisfied by the solution during its last *k encounters*
- The fitness of a test is defined as the number of times the test was violated by the *k* solutions it encountered most recently

Jan Paredis: Coevolutionary computation, *Artificial Life*, 1995

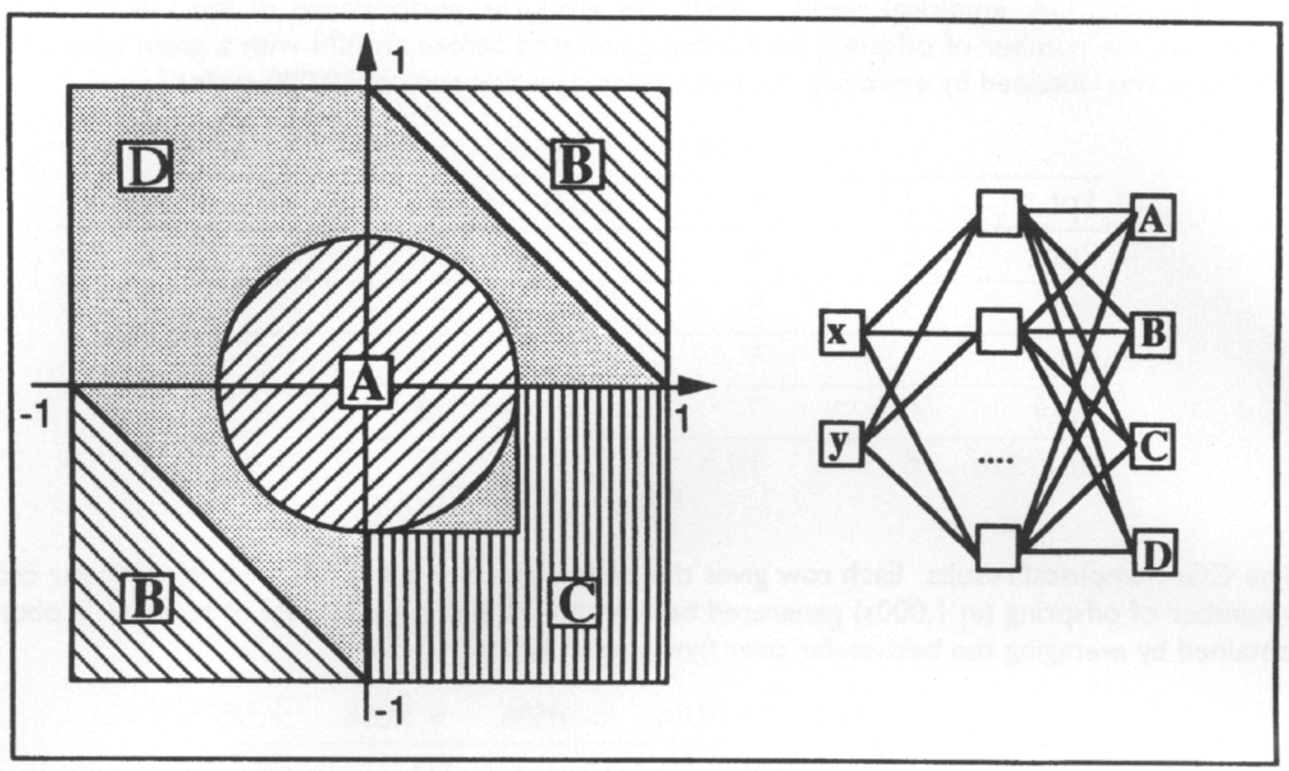
Lifetime fitness evaluation (LTFE)

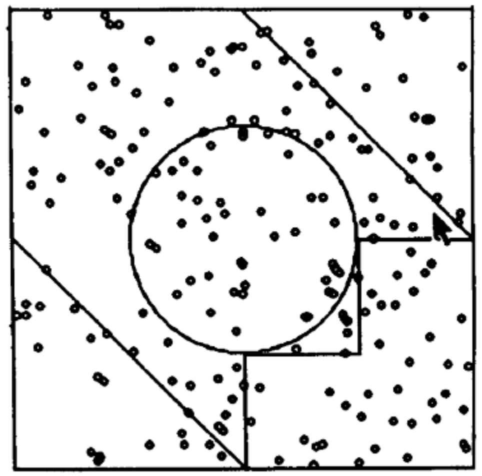
- Only Solution population truly evolves
- Tests are not changed but are evaluated
- More difficult tests will be picked in later stage of the search

Jan Paredis: Coevolutionary computation, *Artificial Life*, 1995

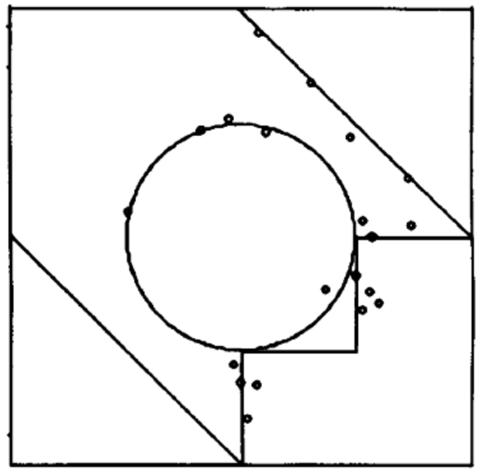
Application example

- Coevolutionary neural net for classification





all 200 tests



final 20 fittest tests

Jan Paredis: Coevolutionary computation, *Artificial Life*, 1995

Cooperative coevolution

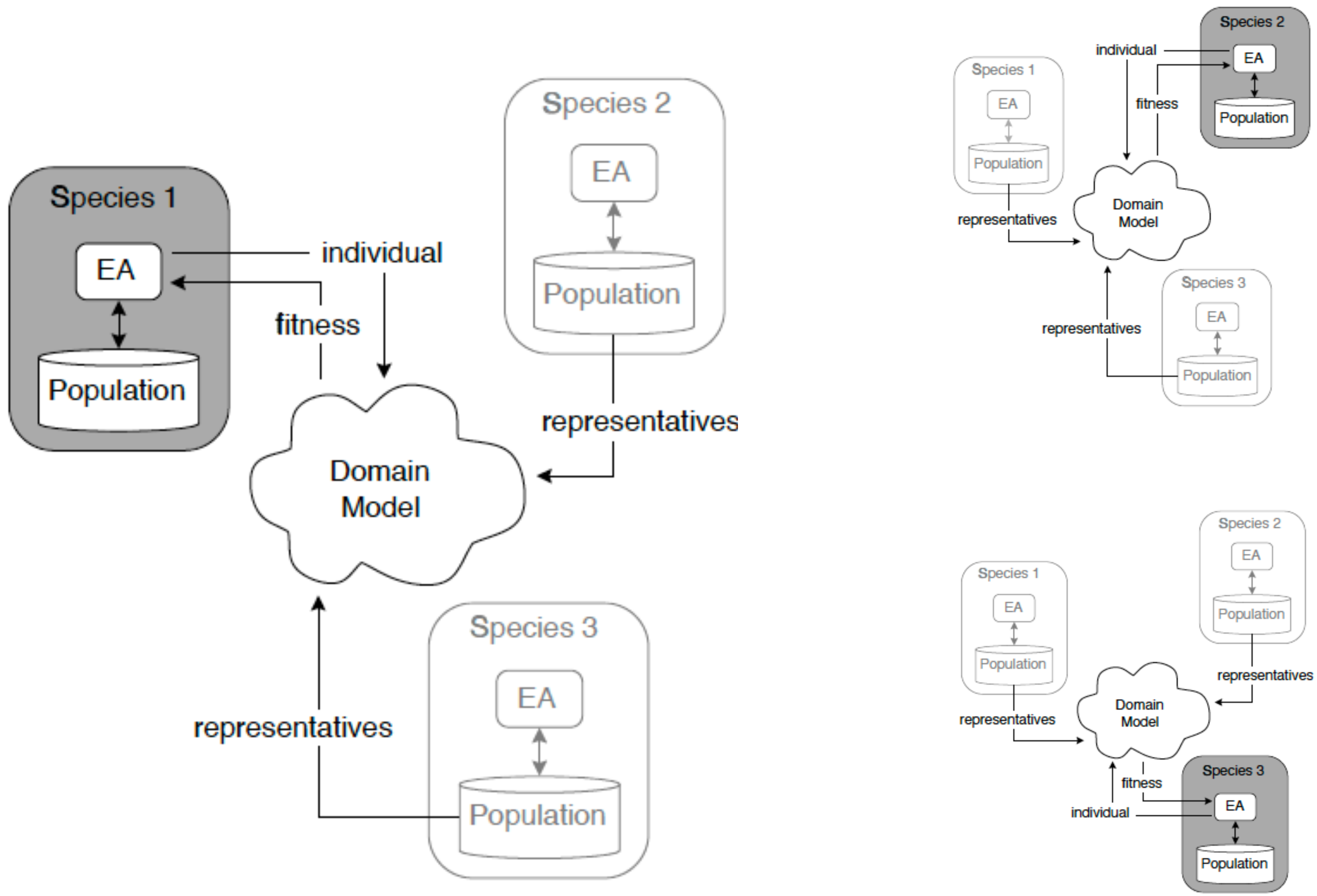
- Different species each represent part of a problem
- Cooperate in order to solve a larger problem
  - high-dimensional function optimization
  - job shop scheduling
- Challenges
  - how to divide the problem
  - how to pair the subpopulations

Evolving coadapted subcomponents

- Increasingly complex problems
  - Solutions evolve in the form of interacting coadapted subcomponents
  - Behavior learning, modularity
- Models an ecosystem consisting of two or more species
  - Species evolve on their own
  - Fitness is assessed by within a shared domain model

Potter and De Jong: Cooperative coevolution, *Evolutionary Computation*, 2000

Evolving coadapted subcomponents



Potter and De Jong: Cooperative coevolution, *Evolutionary Computation*, 2000

Application example

- Problem of coevolving string covers
  - finding a set of *N* binary vectors that match as strongly as possible another set of *K* binary vectors, *K*>>*N*
  - *match set* must contain patterns shared by multiple garget strings to cover the *target set* optimally, i.e., generalize
  - the match strength S between two binary vectors of length L is

$$S(\vec{x},\vec{y}) = \sum_{i=1}^{L} \begin{cases} 1 & \text{if } x_i = y_i \\ 0 & \text{otherwise.} \end{cases}$$

  - the strength of a match set M is $S(M) = \frac{1}{K}\sum_{i=1}^{K} \max(S(\vec{m}_1,\vec{t}_i),\ldots,S(\vec{m}_N,\vec{t}_i))$

Potter and De Jong: Cooperative coevolution, *Evolutionary Computation*, 2000

Target set

- Various schemata
  - half-length schemata

```
11111111111111111111111111111111################################
################################11111111111111111111111111111111
```

  - quarter-length schemata

```
1111111111111111################################################
################1111111111111111################################
################################1111111111111111################
################################################1111111111111111
```

  - eighth-length schemata

```
11111111########################################################
########11111111################################################
################11111111########################################
########################11111111################################
################################11111111########################
########################################11111111################
################################################11111111########
########################################################11111111
```

Potter and De Jong: Cooperative coevolution, *Evolutionary Computation*, 2000

Final species representatives

Half-length

Species 1: 111111111111111111111111111111111010011000100011111100010110111
Species 2: 0010000001001110110100001000100011111111111111111111111111111111

Quarter-length

Species 1: 1111111111111111000110011001010011111111111111110111111111110111
Species 2: 1111111110111111111111010111111111111111111111111000010100010000
Species 3: 0101010101101101111111011111110100001010001011101111111111111111
Species 4: 1001101110011000111111111111111110110100010100011010101101101111

Eighth-length

Species 1: 1100100011111110001011110100000011001011111111110011010011110010
Species 2: 1011111001010001111111110010101110010000101101111110111011010010
Species 3: 1010111100000111111101111001000011100110011110111101111100000111
Species 4: 0000111011111111101110010111101011111111010001101100010111111101
Species 5: 1101100100100010110000111100101111111111100101011001000111111111
Species 6: 0001011011111011011010001111111100110011111111101111111100110000
Species 7: 1111001011111110100000101011010100100100100100010111000111111111
Species 8: 1111111110001101111000001111111101011010110111101100001111101010

Potter and De Jong: Cooperative coevolution, *Evolutionary Computation*, 2000