Slide 0

Statistical dynamics of evolving populations with recombination

Jonathan Shapiro

Department of Computer Science, University of Manchester

September 26, 1999

Slide 1

Outline

Part I — Perspective

Part II — The general approach

Part III — Cumulant dynamics in a simple, Wright-Fisher like model

Part IV — Cumulant dynamics in more complex models

Slide 2

Genetic Algorithms — forced computational evolution

Given a problem to be solved, tries to find a population of good solutions.

Sequence $\rightarrow$ representation of candidate solutions to the problem.

(Quantitative) trait $\rightarrow$ Objective function; thing to be optimized.

Selection: pick the better solutions, discard the worse ones

Mutation: make small changes to the candidate solutions

Recombination/Crossover: combine parts of one solutions with different parts of another solution.

Slide 3

Example: Filling a backup disk with files

(A knapsack problem; weakly NP.)

- There are L files of size f_i to be backed up.
- A set of backup disks of size D_I
- Task — back up as many files as possible with as little wasted space as possible.

Slide 4

As a genetic algorithm (assume a single disk)

Sequence:

$$x_i = \begin{cases} 0; & \text{file } i \text{ not on disk} \\ 1; & \text{file } i \text{ on disk} \end{cases}$$

Objective function: $E = \left(\sum_{i=1}^{L} x_i f_i - D \right)^2 \Theta \left(D - \sum_{i=1}^{L} x_i f_i \right).$

Population: of N strings. Fixed size.

Selection weight: N parents are chosen from the current population; probability of selecting string α is

$$p^\alpha = \frac{W(E^\alpha)}{\sum_\gamma W(E^\gamma)}.$$

Slide 5

Recombination/Crossover: This operation produces offspring from a pair of parents by *randomly* combining parts of each parent.

Uniform crossover: (Free recombination) Each site is independently chosen *at random* from parent 1 or parent 2.

$$\left.\begin{array}{r} 11111111111111 \\ 00000000000000 \end{array}\right\} \rightarrow 00101101101011$$

Single-point crossover:

$$\left.\begin{array}{r} 1111111|1111111 \\ 0000000|0000000 \end{array}\right\} \rightarrow 1111111|0000000$$

Slide 6

Multi-point crossover:

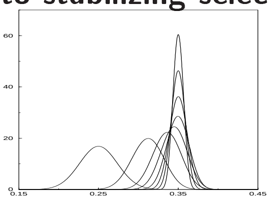
$$\left.\begin{array}{l} 111|111111|11111 \\ 000|000000|00000 \end{array}\right\} \rightarrow 111|000000|11111$$

Mutation: This operation simply changes characters in the offspring string at random. Each character is changed with a small probability.

Do these steps repeatedly until solutions of sufficient quality found.

Slide 7

Equivalent to stabilizing selection model



Like quantitative genetics, but asking different questions:

1. transient behaviour — how long to equilibrate
2. finite population effects — search properties are important
3. optimization of search parameters — how to get best improvement

Generic questions of *forced* evolution.