An Algorithm for Calculating the Long-Run Behavior of Genetic Algorithms in Economic Modeling

Ludo Waltman and Nees Jan van Eck
Erasmus University Rotterdam, The Netherlands
{lwaltman,nvaneck}@few.eur.nl

14th International Conference on
Computing in Economics and Finance

Paris, June 28, 2008

Introduction

- Genetic algorithms (GAs) can be used to model boundedly rational behavior
- We focus on GAs for modeling social learning
- The general idea is as follows:
  - There is a population of chromosomes
  - Each chromosome represents a strategy
  - Selection models imitating successful strategies
  - Crossover models combining elements from successful strategies
  - Mutation models experimenting or making errors
- GA models are almost always analyzed using computer simulation
- Our aim is to analyze GA models without using computer simulation

Existing theoretical results

- A GA is an ergodic Markov chain
  - Hence, in principle, the long-run behavior of a GA can be calculated exactly
  - However, in practice, this is infeasible even for small problems
- In the limit as the mutation rate approaches zero, the stationary distribution of a GA Markov chain has all its mass on uniform populations (Dawid, 1996)
  - Hence, in the long run, the probability that the population in a GA is uniform approaches one as the mutation rate approaches zero
  - However, calculating the relative likelihood of the different uniform populations seems infeasible even for small problems

New theoretical result

- In the limit as the mutation rate approaches zero, the stationary distribution of a GA Markov chain depends only on a small portion of the transition probabilities
  - Consequently, for small and medium-sized problems, exact calculation of the long-run probabilities of uniform populations becomes feasible
  - The long-run probabilities do not depend on the crossover rate

Algorithm

- Based on our theoretical result, we have derived an algorithm for calculating the long-run behavior of GAs in the limit as the mutation rate approaches zero
- Due to memory constraints, on today's personal computers, the algorithm can be run for chromosomes with length up to 23 or 24 bits
- Large population size (e.g., 1000 chromosomes) is no problem

Advantages over simulation

- Exact results, no statistical uncertainty
- Analysis of truly long-run behavior
- Less arbitrary choice of the mutation rate
- Less computing time needed than for a comprehensive simulation study

Application (1)

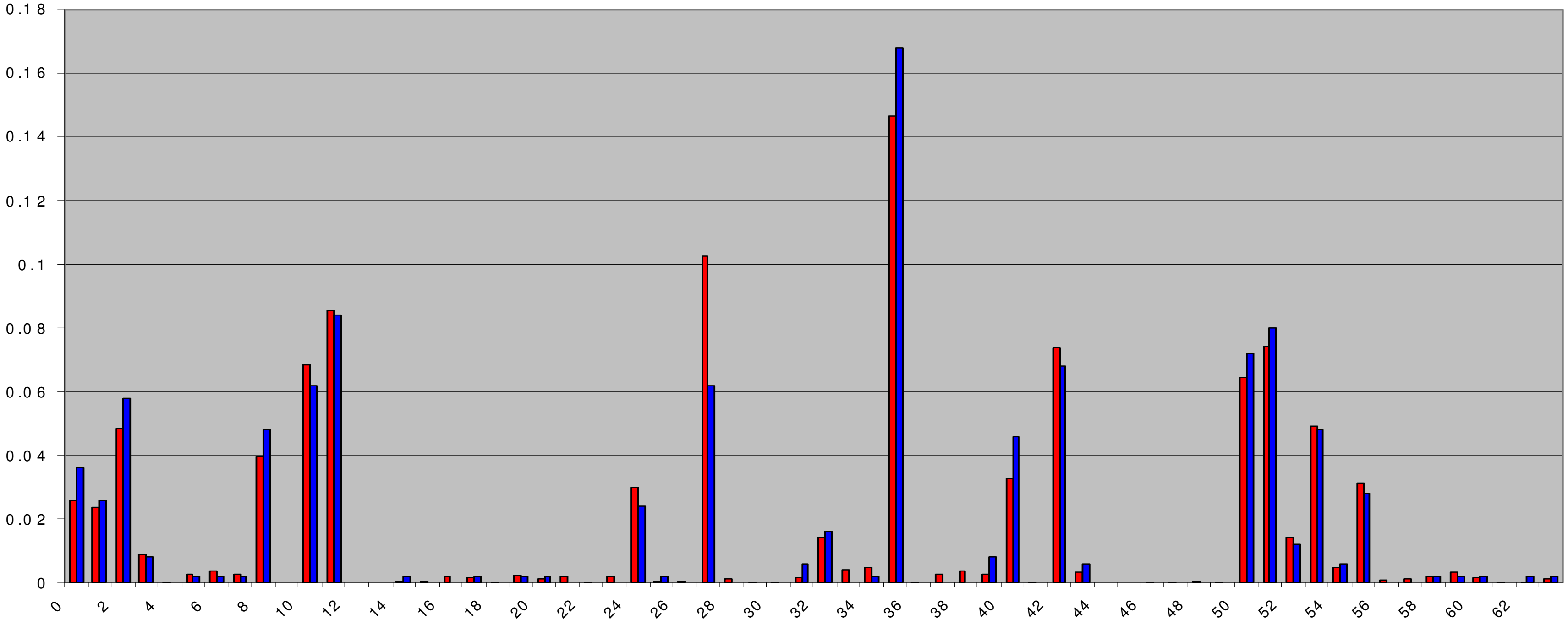
- Based on a well-known paper by Axelrod (1987)
- Chromosomes represent strategies in the iterated prisoner's dilemma (IPD)
- Memory lengths of 1, 2, and 3 periods are considered
- Chromosome lengths equal, respectively, 6, 20, and 70 bits
- Population consists of 20 chromosomes
- Fitness of a chromosome equals the mean payoff obtained by playing one IPD game of 151 periods against each chromosome in the population

Application (2)

- The payoffs in a single period of an IPD game are as follows:

	Cooperate	Defect
Cooperate	(3, 3)	(0, 5)
Defect	(5, 0)	(1, 1)

Results (1)



- Long-run distribution over the uniform populations: exact results (red) vs. simulation results (blue)
- Memory length equals 1
- Simulation results are based on 500 GA runs of 100,000 iterations with mutation rate 0.0001

Results (2)

		Memory length		
		1	2	3
Mutation rate	0.01000	2.78 ± 0.05	2.66 ± 0.05	2.68 ± 0.04
	0.00100	2.21 ± 0.08	2.32 ± 0.07	2.64 ± 0.05
	0.00010	1.89 ± 0.09	2.35 ± 0.07	2.50 ± 0.06
	0.00001	1.84 ± 0.09	2.27 ± 0.07	2.47 ± 0.06
	lim → 0	1.84	2.29	???

- Simulation results are based on 500 GA runs of 100,000 iterations
- Mean fitness at the end of a GA run together with a 95% confidence interval is reported

Conclusions

- Our theoretical result makes it possible to calculate the long-run behavior of a GA rather than to estimate it using computer simulation
- There are two restrictions:
  - Only the limit case in which the mutation rate approaches zero can be analyzed
  - Chromosomes must be no longer than 23 or 24 bits

If you want to know more about our theoretical result or about the algorithm derived from it, please let us know!