Genetic Algorithms

Intelligent Robotics Seminar

Prasanth Sivakumar (#6517408)

Winter Semester 2013/14

20.01.2014

Outline

1. Introduction

- What is Genetic Algorithm (GA)?
  - Evolutionary Technique designed to solve Searching and Optimization problems.
  - Imitates Natural Selection

- Features of GA
  - Population of possible solutions (usually Binary)
  - Fitness Function
  - Selection of possible parents
  - Genetic Operations
  - Termination

2. Motivation

- Natural Selection – Theory of Evolution

Example - Owls inside mountain caves

- Initial Population with same fitness
- Selection
- Genetic Operators – Crossover and Mutation
- Fitness Evaluation
- Next Generation Population

3. Genetic Algorithm

```
Simple Genetic Algorithm ()
{
initialize population;
evaluate population ;
while convergence not achieved
        {
        scale population fitnesses ;
        select solutions for next population ;
        perform crossover and mutation ;
        evaluate population ;
        }
}
```

Fig. 1. Basic structure of a GA.

1. Initialize Population (Population size, Encoding)

    A_1 B_1 …

    0110100010, 1010001101

2. Evaluate Population; Solutions - Fitness Functions and Fitness Values

3. Genetic Algorithm (contd..)

3. Selection of Parents

4. Genetic Operators (p_c, p_m)

   1. Crossover

      A_1’ - 0110101101

      B_1’ - 1010000010

   2. Mutation

      A_2 - 0111101101

      B_2 – 1011000010

5. Repeat with New Population

6. Terminate (Convergence Conditions)

3.1 Variations of GA

1. Simple Chromosome Representation
   Represented Numerically (usually Bit Strings)
2. Elitism
   The Best fit solution survives
3. Hybrid Genetic Algorithms
   Memetic Algorithm, Parallel implementation of Darwinian evolution and local heuristics
4. Self-Organizing Genetic Algorithms
   GA parameters are subjected to evolution

- Adaptive Genetic Algorithms
  Changing Genetic Operators (Population size N, p_c, p_m) throughout the algorithm

3.2 Adaptive Genetic Algorithm

- Altering the GA Operators during the progress of the algorithm
- P_c and P_m have significant control over the performance of GA.
- Higher P_c -> Wide variety of new solutions in the population
- Significant P_m -> Less likely to Converge prematurely with a Suboptimal Solution
- Population Size
  - Smaller population affects Convergence rate
  - Large population increases the Algorithm runtime.

3.3 Applications of GA

- GA can be implemented to find optimized solutions for many problems ranged into many fields
- Widely applied in the field of Engineering – Designing networks, Scheduling, Database Mining,
- Robotics – Training Neural Networks, Navigation
- For solving Timetabling and Scheduling problems
- Solving Global Optimization Problems
- Pattern and Speech Recognition Systems
- Control Systems – Improving System Performance

3.4 Path Planning Applications

1. Travelling Salesman Problem [SP94]

$$p_c = k_1(f_{\max} - f')/(f_{\max} - \overline{f}), \quad f' \geq \overline{f},$$
$$p_c = k_3, \quad f' < \overline{f}$$

$$p_m = k_2(f_{\max} - f)/(f_{\max} - \overline{f}), \quad f \geq \overline{f},$$
$$p_m = k_4, \quad f < \overline{f}$$

Observed Performance of AGA against Simple GA

PERFORMANCE OF AGA AND SGA FOR TSP

Cities	Avg. Tour Length		Optimum Tour Located		Max. Gens.	Pop. Size
	SGA	AGA	SGA	AGA		
30 (424.0)	442.1	430.2	0	7	100	1000
105 (14383)	16344.3	14801.4	0	4	500	2000

3.4 Path Planning Applications (contd.)

2. Path Planner for Autonomous Underwater Vehicle[WZBS05]

   Proposed Probabilities

   Crossover Probability

$$P_{ctemp} = P_{cMax} \times 2^{(-n/N_{Gen})}$$

$$P_c(n) = \begin{cases} P_{ctemp} & if\,(P_{ctemp} > P_{cMin}) \\ P_{cMin} & else \end{cases}$$

   Mutation Probability

$$P_{mtemp} = P_{mMax} \times 2^{(-n/N_{Gen})}$$

$$P_m(n) = \begin{cases} P_{mtemp} & if\,(P_{mtemp} > P_{mMin}) \\ P_{mMin} & else \end{cases}$$

3.4 Path Planning Applications (contd.)

Proposed Optimal Paths

Fig.1 Planning path based on GA

Fig.2 Planning path based on AGA

Performance of AGA against GA

planner / item	GA planner	AGA planner
Total generations	162	69
Initial num of chromosome	76	253
Ultimate num of chromosome	8	5
The generation near stable status	84	19
Cost time (sec.)	41.860	25.187

4. Restrictions of GA

1. Cannot guarantee that GA provides the Best Solution. There almost always exists a better solution.
2. Evaluating the fitness functions and assigning fitness values could be tedious and complex when scaled.
3. Complexity increases with problems
4. Premature Convergence with suboptimal Solution if stuck in a local minimum of the function.
5. Cannot effectively be used to derive trivial solutions
6. For Simple GA, static Genetic operators increases the cost of the algorithm

5. Summary

- Genetic Algorithm
- Bio-Inspiration of Genetic Algorithm
- Structure of Simple Genetic Algorithm
- Variations of Genetic Algorithm
- Adaptive Genetic Algorithm
- Applications of Genetic Algorithm
- Path Planning using Adaptive Genetic Algorithm
- Limitations of Genetic Algorithm

Reference

Literature

- [SP94] - Adaptive probabilities of crossover and mutation in genetic algorithms

IEEE Systems, Man, and Cybernetics Society, (1994)

- [WZBS05] - An Improved Path Planner based on Adaptive Genetic Algorithm for Autonomous Underwater Vehicle

Mechatronics and Automation, 2005 IEEE International Conference, (2005)

- R. R. Sharapov (2007). Genetic Algorithms: Basic Ideas, Variants and Analysis, Vision Systems: Segmentation and Pattern Recognition, Goro Obinata and Ashish Dutta (Ed.)
- http://en.wikipedia.org/wiki/Genetic_algorithm
- http://www.obitko.com/tutorials/genetic-algorithms/index.php

- Questions and Discussions

- Feedback

Thank you for your attention