Genetic Algorithms

Michael J. Watts

http://mike.watts.net.nz

Lecture Outline

- Genetic algorithms
- Jargon
- Advantages of GAs
- Disadvantages of GAs
- Simple genetic algorithm
- Encoding schemata
- Fitness evaluation

Lecture Outline

- Selection
- Creating new solutions
- Crossover
- Mutation
- Replacement strategies
- Word matching example

Genetic Algorithms

"Genetic algorithms are search algorithms based on the mechanics of natural selection and natural genetics" Goldberg, 1989

Genetic Algorithms

- A kind of evolutionary algorithm
- Also known as GA(s)

Jargon

- Locus - a position on a chromosome
- Gene - a portion of a chromosome representing a parameter of the solution set
- Alleles - different values of a particular gene. Members of the domain of the gene's value
- Chromosome - string of genes

Jargon

- Genotype - coding of the solution
- Phenotype - expression of the genotype
- Population - group of individuals capable of interbreeding

Advantages of GA

- Efficient means of investigating large combinatorial problems
  - can solve combinatorial problems many orders of magnitude faster than exhaustive ‘brute force’ searches

Disadvantages of GA

- GAs are not ‘silver bullets’ for solving problems
- Must be able to assess the quality of each attempt at a solution
  - can’t crack PGP with a GA

Disadvantages of GA

- Computationally expensive
  - some problems require many days or weeks to run
  - often still faster than brute force, however
- Blind, undirected search
  - difficult to direct a GA towards optimal solution area if known

Disadvantages of GA

- Can be sensitive to initial parameters
  - parameters such as mutation can significantly influence the search
- Stochastic process
  - not guaranteed to find an optimal solution, just highly likely to

Simple Genetic Algorithm

1. Select an encoding schema
2. Randomly initialise chromosome pool
3. Evaluate each individual in the population
4. Select fit individuals to breed new population
5. Create new population from parents
6. Replace old population with new population
7. Repeat steps 3 - 6 for each generation

Encoding Schemata

- Two competing principles
- Meaningful building blocks
  - "user should select a coding so that short, low order schemata are relevant to the underlying problem"
- Minimal alphabets
  - "user should select the smallest alphabet that permits a natural expression of the problem"

Fitness Evaluation

- Method dependent upon problem
- Involves quantifying performance of the phenotypes

Fitness Evaluation

- Normalisation or scaling of fitness values required to prevent good solution overwhelming later generations
  - known as "premature convergence"
  - similar to the 'local minima' problem with neural networks

Selection

- Involves selecting the parents of the next generation
- Many methods in existence
- All based upon the fitness of the individual

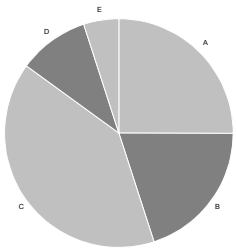
Selection

- Roulette selection
  - each individual is given a slice of a virtual roulette wheel
  - size of each slice is proportional to fitness
  - spin the wheel once to select each individual

Roulette Selection

Individual	Fitness
A	0.25
B	0.2
C	0.4
D	0.1
E	0.05

Roulette Selection



Create New Solutions

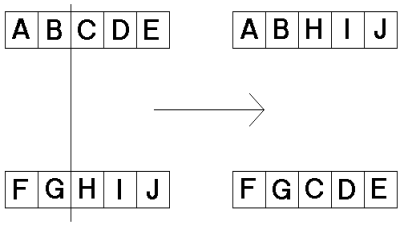
- Creates new individuals from selected parents
- Two operators come into play
  - crossover, and
  - Mutation

Crossover

- Two chromosomes join at one or more points and exchange genes
- Types of crossover include

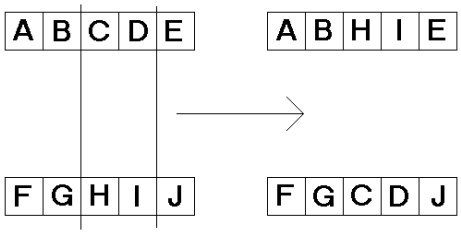
Crossover

- One point
  - chromosomes join at only one locus



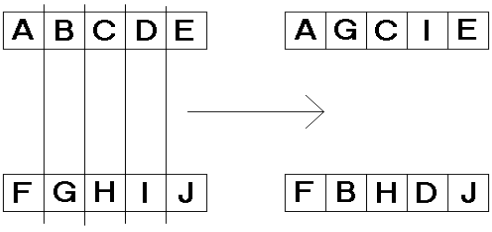
Crossover

- Two point
  - chromosomes join at two loci



Crossover

- Uniform
  - crossover at each locus determined by a “coin toss”



Mutation

- One or more allele is randomly chosen and it's value changed
- Method of change depends upon coding schema used
- Best rate of mutation subject of much current research

Mutation

- Some dispute it's necessity
- Effect of mutation dependent upon size of alphabet
  - the higher the cardinality of the alphabet, the lower the benefit of mutation

Replacement Strategies

- Replace some or all of the parent population with children
- Many different replacement strategies available
- Most concerned with preserving 'good' genes and purging 'bad' genes

Word Matching Example

- Problem is to 'guess' a word
- Difficult to solve with a brute force approach
  - assuming case sensitivity, to investigate every possible combination of letters in a seven letter word would require 1,028,071,702,528 attempts

Word Matching Example

- Assume trying to guess the word 'genetic'
  - assign fitness based on number of correct letters in the correct place
- Step one: select an encoding schema
  - use characters

Word Matching Example

- Step two: initialise the population
  1. kdjirid
  2. ginddcc
  3. nmugjyb
  4. zezezez
  5. uhjklyt
  6. wojikli
  7. kladonn
  8. flortik

Word Matching Example

- Step three: evaluate the population

individual	genotype	fitness
1	kdjirid	1
2	ginddcc	3
3	nmugjyb	0
4	zezezez	2
5	uhjklyt	0
6	wojikli	0
7	kladonn	0
8	flortik	2

Average fitness = 1

Word Matching Example

- Step four: select breeding population
  - selection based on fitness, so breeding population is

individual	genotype	fitness
2	ginddcc	3
2	ginddcc	3
2	ginddcc	3
4	zezezez	2
4	zezezez	2
8	flortik	2
8	flortik	2
1	kdjirid	1

Word Matching Example

- Step five: create new population
  - crossover comes into play
  - no mutation used here to keep things simple
  - eg. cross individual 2 with individual 4
    - individual 2 genotype is: ginddcc
  - individual 4 genotype is: zezezez
  - crossing over produces : genedec

Word Matching Example

- New population is:

individual	genotype	fitness
1	genedec	5
2	glnrdic	4
3	fdoitik	2
4	zdziziz	1
5	gdnidic	4
6	feoetek	3
7	kijdrcd	0
8	zlzrez	0

average fitness = 2.375

Summary

- Genetic algorithms are a class of evolutionary algorithm
- Able to efficiently investigate large search spaces
- Have two major requirements
  - representation
  - Evaluation

Summary

- Population evolves by creating more offspring from fitter members
  - fitness based selection
- Offspring created using crossover and mutation operators
- Must be mindful of the disadvantages