Genetic Algorithms

Presented at
Computer Society Meeting
5 June 1997, Ottawa

by

Prof. V. Rao Vemuri
Department of Applied Science
University of California, Davis
Livermore, CA

vemuri1@llnl.gov
• What are Genetic Algorithms?
• Where are They Useful?
• A Simple Example
• More Realistic Examples
WHAT ARE GAs?

• INSPIRED BY BIOLOGICAL EVOLUTION
• SEARCH AND OPTIMIZATION PROCEDURES
• METAPHORICAL EVOLUTIONARY PROCESSES
• GOOD IN RUGGED SEARCH SPACES
• GOOD FOR MULTIMODAL ENVIRONMENTS
• PROBLEM INDEPENDENT
MAIN IDEAS

• SOLUTIONS ENCODED AS CHROMOSOMES
• MAINTAINS POPULATION OF SOLUTIONS
• EVALUATE FITNESS OF ALL SOLUTIONS
• SELECT THOSE WITH DESIRABLE TRAITS
• MATE THEM VIA CROSSOVER (EXPLOITATION)
• MUTATE THEM (EXPLORATION)
• REPEAT UNTIL "CONVERGENCE"
CHARACTERISTICS OF GAs

- CAN SOLVE HARD PROBLEMS RELIABLY
- EASY TO INTERFACE WITH EXISTING MODELS
- EASY TO HYBRIDIZE WITH OTHER METHODS
- EXTENDIBLE
- NEEDS VERY LITTLE DOMAIN KNOWLEDGE
SEARCH USING GA

Fitness
Distribution of individuals in Generation 1

Fitness
Distribution of individuals in Generation 5
HOW A GENETIC ALGORITHM WORKS

- Reproduction
- Population of chromosomes
- Discarded chromosomes
- Dustbin
- Parents
- Evaluated children
- Chromosome alteration
- Altered children
- Chromosome evaluation
GENETIC OPERATORS

- SELECTION:
  SURVIVAL OF THE FITTEST

- CROSSOVER:
  COMBINES INDIVIDUALS TO GENERATE NEW COMBINATIONS OF PARTIAL SOLUTIONS

- MUTATION:
  INTRODUCES FEATURES THAT ARE NOT PRESENT IN THE PARENTS
SELECTION OPERATOR

• CALCULATE FITNESS OF ALL MEMBERS
• SELECT THOSE WITH BETTER FITNESS SCORES
• USE ONE OF MANY SELECTION SCHEMES
  - ROULETTE WHEEL SELECTION
    (A.K.A. Fitness Proportionate Reproduction, or FPR)
  - TOURNAMENT SELECTION
A simple crossover operation

Parent 1: 1 0 0 1 1 1 0 0
Parent 2: 0 1 0 0 0 1 1 0

Child 1: 1 0 0 0 0 1 0 0
Child 2: 0 1 0 1 1 1 0 0

Other types of crossovers exist.
MUTATION OPERATION

PARENT 1 0 0 _ 1 1 1 0 0

CHILD 1 0 0 _ 0 1 1 0 0
SAMPLE PROBLEM

• A FRANCHISER WANTS PROFIT-MAKING STRATEGY

• OPTIONS ARE: AFFORDABLE VS FANCY
  HOT DOGS VS CREPES
  SOFT DRINK VS WINE

• FORECAST ON COSTS AND SALES AVAILABLE
ENCODING THE PROBLEM

• DEFINE A 3-BIT CHROMOSOME

<table>
<thead>
<tr>
<th>BIT 3</th>
<th>BIT 2</th>
<th>BIT 1</th>
</tr>
</thead>
</table>

• ENCODE AS FOLLOWS

- BIT 3: 0 AFFORDABLE, 1, FANCY
- BIT 2: 0 HOT DOGS, 1, CREPES
- BIT 1: 0 SOFT DRINKS, 1, WINE
FITNESS EVALUATION

• EACH OPTION CAN BE REPRESENTED BY A 3-BIT SEQUENCE

• IN THIS SMALL PROBLEM THERE ARE ONLY 8 OPTIONS

• ASSUME THAT THE PROFIT MADE ON EACH OPTION CAN BE CALCULATED BY A FORMUL, SUCH AS

   PROFIT = DECIMAL VALUE OF 3-BIT SEQUENCE
A PEEK AT THE CORRECT SOLUTION

<table>
<thead>
<tr>
<th>OPTION</th>
<th>PROFIT</th>
<th>OPTION</th>
<th>PROFIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>$0</td>
<td>100</td>
<td>$4</td>
</tr>
<tr>
<td>001</td>
<td>$1</td>
<td>101</td>
<td>$5</td>
</tr>
<tr>
<td>010</td>
<td>$2</td>
<td>110</td>
<td>$6</td>
</tr>
<tr>
<td>011</td>
<td>$3</td>
<td>111</td>
<td>$7</td>
</tr>
</tbody>
</table>
EXHAUSTIVE ENUMERATION IMPrACTICAL

- CONSIDER TRAVELLING SALESPERSON PROBLEM
- ASSUME

  20-CITY PROBLEM TAKES ABOUT 1 HOUR
  THEN
  21-CITY PROBLEM TAKES ABOUT 20 HOURS
  22-CITY PROBLEM TAKES ABOUT 20 DAYS

  25-CITY PROBLEM TAKES ABOUT 6 CENTURIES!!
A POSSIBLE SOLUTION BY GA

- Try 4 randomly selected strategies at 4 different outlets
- Observe how outlets perform for a week
- Compare their performance
- Pick those that are doing well: mating pool
- Generate new strategies from mating pool using crossover and mutation
- Go back to first step
<table>
<thead>
<tr>
<th>$x(i)$</th>
<th>$F(i)$</th>
<th>M.p</th>
<th>$F(i)$</th>
<th>$X(i)$</th>
<th>$F(i)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>011</td>
<td>3</td>
<td>011</td>
<td>3</td>
<td>111</td>
<td>7</td>
</tr>
<tr>
<td>001</td>
<td>1</td>
<td>110</td>
<td>6</td>
<td>010</td>
<td>2</td>
</tr>
<tr>
<td>110</td>
<td>6</td>
<td>110</td>
<td>6</td>
<td>110</td>
<td>6</td>
</tr>
<tr>
<td>010</td>
<td>2</td>
<td>010</td>
<td>2</td>
<td>010</td>
<td>2</td>
</tr>
<tr>
<td>Total F</td>
<td>12</td>
<td></td>
<td>17</td>
<td></td>
<td>17</td>
</tr>
<tr>
<td>Worst</td>
<td>1</td>
<td></td>
<td>2</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>Ave</td>
<td>3</td>
<td></td>
<td>4.5</td>
<td></td>
<td>4.25</td>
</tr>
<tr>
<td>Best</td>
<td>1</td>
<td></td>
<td>6</td>
<td></td>
<td>7</td>
</tr>
</tbody>
</table>
WHAT DID THE FRANCHISER LEARN?

(1) $3 IS AN ESTIMATE OF THE AVERAGE FITNESS (PROFIT) OF THE SEARCH SPACE
   - Estimate Based on 4 Samples

(2) THAT
   - 110 is 200% better than estimated average
   - 010 is 2/3 as good as estimated average
   - 001 is 1/3 as good as estimated average
WHAT DO WE DO NEXT?

OPTION 1: BRUTE FORCE METHOD

• CONTINUE SEARCHING FOR BETTER SOLUTIONS

DRAWBACK:

• UNIVERSE CREATED 15 BILLION YEARS AGO

• AT BILLION SOLUTIONS/SEC, WE WOULD HAVE SEARCHED ONLY $2^{90}$ SOLUTIONS

• THIS MEANS THAT OUR CHROMOSOMES CANNOT BE LONGER THAN 90 BITS!!
OPTION 2: GREEDY METHOD

• BECAUSE 110 IS 200% BETTER THAN THE AVERAGE, GRAB IT. DO NOT WORRY ABOUT POSSIBLE BETTER SOLUTIONS

• EXPLORING EVERY NEW POINT COSTS $6 - $3 = $3, on the average

• NOT EXPLORING COSTS $7 - $6 = $1, on the average

• THIS IS EXPLORATION VS EXPLOITATION!
SUMMARY: A SIMPLE GA

GENERATE RANDOMLY AN INITIAL POPULATION

EVALUATE FITNESS OF THE POPULATION

for GENERATION = 1 TO MAX_GENERATIONS

SELECTION: CREATE MATING POOL

MATING: CROSSOVER AND MUTATE

REPLACEMENT: REPLACE ENTIRE POPULATION WITH OFFSPRING

end for
OTHER EXAMPLE PROBLEMS

1. Automatically Generating Computer Programs (Koza)
2. Prisoner’s Dilemma (Axelrod)
3. Designing a Sorting Network Using Diploid Chromosomes (Hillis)
4. Prediction of Protein Secondary Structure (Koza)
5. Multi-objective Optimization (Cedeno and Vemuri)
6. DNA Fragment Assembly (Cedeno and Vemuri)
WHAT IS GENETIC PROGRAMMING?

• THINK OF EACH SOLUTION IN THE GA AS A COMPUTER PROGRAM

• START WITH A FAMILY OF RANDOMLY GENERATED COMPUTER PROGRAMS

• RUN EACH PROGRAM AND DETERMINE HOW WELL IT SOLVES THE PROBLEM AT HAND

• CROSSOVER AND MUTATE PROGRAMS TO GENERATE OFFSPRING PROGRAMS

• AFTER A FEW GENERATIONS ENJOY A CORRECT COMPUTER PROGRAM
• PROBLEM: LOCATE TRANSMEMBRANE SEGMENTS OF BACTERIORHODOPSIN

• TM PROTEINS CROSS CELL WALLS SEVERAL TIMES

• SHORT LOOPS ON EITHER SIDE OF THE MEMBRANE

• GOAL: IDENTIFY SEGMENTS THAT ARE WITHIN THE MEMBRANE, THE TRANSMEMBRANE DOMAINS

• OBJECTIVE: EVOLVE A COMPUTER PROGRAM FOR PREDICTING WHETHER OR NOT A GIVEN SEGMENT LIES IN TM DOMAIN
WINNING PROGRAM

(prog (looping-over-residues
    (SETM3 (- (+ (- (F?) (K?)) ((+ (-M3 (P?))
    (+ (I?) (SETM2 (SETM3 (L?)))))))) (SETM2 (SETM2 (H?)))))
(values (* (IFLTE (IFLTE (+ -5.606 M3) (* L M2) (% -2.786
    (IFLTE M1 M3 M2 M2)) (* M2 M0 ) (*% (+M2 M3)
    (+M3 L) (%M2 L )) (* (+ (+M2 M1) (*M2 M0))
    (%M2 M2))))))
PROJECTS AT DAS/LLNL

- GP WITH SISAL, A PARALLEL LANGUAGE
- GA TO TRAIN NEURAL NETS
- GA TO PROTEIN STRUCTURE PREDICTION
- GA TO DNA SEQUENCING ANALYSIS
- GA TO GROUND WATER REMEDIATION
- GA TO INFORMATION FILTERING
- GA TO COMMUNICATIONS AND NETWORKING
A FLOW CHART

1. **GEN**:= 0
2. GENERATE INITIAL RANDOM POPULATION
3. EVALUATE FITNESS OF EACH INDIVIDUAL
   - yes
     - TERMINATION CONDITION SATISFIED
     - no
6. **GEN**:= **GEN**+1
7. **S.O.I**
   - **Pr**
   - REPRODUCE
   - COPY INTO NEW POPUL.
8. **S.P.I**
   - **Pc**
   - XOVER
   - C.I.N.P.
9. **M** = POP_SIZE
10. **I** := 0
11. yes
12. **IS I** := **M**?
13. no
14. **Pm**
15. SELECT ONE GENETIC OPERATION PROBABILISTICALLY
16. **I** := **I**+1
17. yes
18. END
SUMMARY

• GAs ARE RANDOMIZED OPTIMIZATION PROCEDURES

• GAs ARE EASY TO LEARN AND IMPLEMENT

• GAs CAN BE APPLIED TO A WIDE RANGE OF PROBLEMS

• GAs CAN BE COMBINED WITH NEURAL NETS

• GAs CAN BE USED TO AUTOMATICALLY GENERATE CORRECT COMPUTER PROGRAMS
REFERENCES

• Holland, J. Adaptation in Natural and Artificial Systems, MIT Press, 1975
• Proceedings of the International Conferences on Genetic Algorithms, Lawrence Erlbaum Assoc. 1985, 87, 89, 91
• Davis, L. Genetic Algorithm and Simulated Annealing, Pitman, 1987
• Goldberg, D. Genetic Algorithms in Optimization, Search, and Machine Learning, Addison Wesley, 1989
• Davis, L. Handbook of Genetic Algorithm, Van Nostrand.1990
• Koza, J. Genetic Programming: On the Programming of Computers by Means of Natural Selection, MIT Press
• Koza, J. Genetic Programming II: Automatic Discovery of Reusable Programs, MIT Press, 1994