Genetic Algorithms

- Introduction
- Genetic algorithms and optimization techniques
- Fitness function
- Genetic operators
- Case study: Job Shop Scheduling

Introduction

- Decision making is required in all fields including science & Technology, management etc.
- Many traditional optimization procedures have been developed to solve problems like linear and nonlinear programming, inventory, assignment and scheduling problems
- The main drawback of these traditional techniques is most of the time they may end up with local optima rather than global optima

Introduction contd..

- Some non traditional search and optimization techniques include
  - Genetic Algorithms
  - Simulated Annealing
  - Ant Colony optimization etc.
- The concept of GA is conceived by Prof. Holland of university of Michigan

Introduction contd..

- Genetic Algorithms are techniques to solve complex problems. The basis of these techniques relies on the natural process of human evolution over a period of time

4

Difference between Genetic Algorithm and traditional optimization techniques

- *GA works with a string coding of variables instead of variables*; the coding discretizes the search space even for continuous functions but only appropriate coding of the problem can converge to a right solution
- Since GA requires function values at discrete points, a discrete or discontinuous function can also be handled
- GA operators find the similarities in string structures and hence may find the global best solution

Difference between Genetic Algorithm and traditional optimization techniques contd..

- GA processes more than one string simultaneously and thus works with a population of solutions instead of one solution at a time; also previously found information is used efficiently in the next iteration
- Since GA is population based search algorithm, multiple optimal solutions can be obtained without much effort
- GA does not require any additional information except the object function value

Difference between Genetic Algorithm and traditional optimization techniques contd..

- Many real world optimization problems are multimodal problems and GA can handle such problems

  - GA skips the local minimum (due to the probabilities used) and explores all the solution space and hence can reach the global minimum

Theory of Darwinian Evolution

- Chromosomes are the basis of creation of life
- Every chromosome contains millions of genes
- Every gene contributes to a phenotype – a specific feature or characteristic of the child such as the color of the eye, blood group etc. that are inherited from parents/ancestors

8

Theory of Darwinian Evolution contd..

- Each gene for the blood type can have the values 0, 1, or 2 called alleles

- A set of alleles form a gene-pair called genotypes

For example for the blood group or phenotype A, the genotype is 01 or 11; for phenotype AB, the genotype is 21

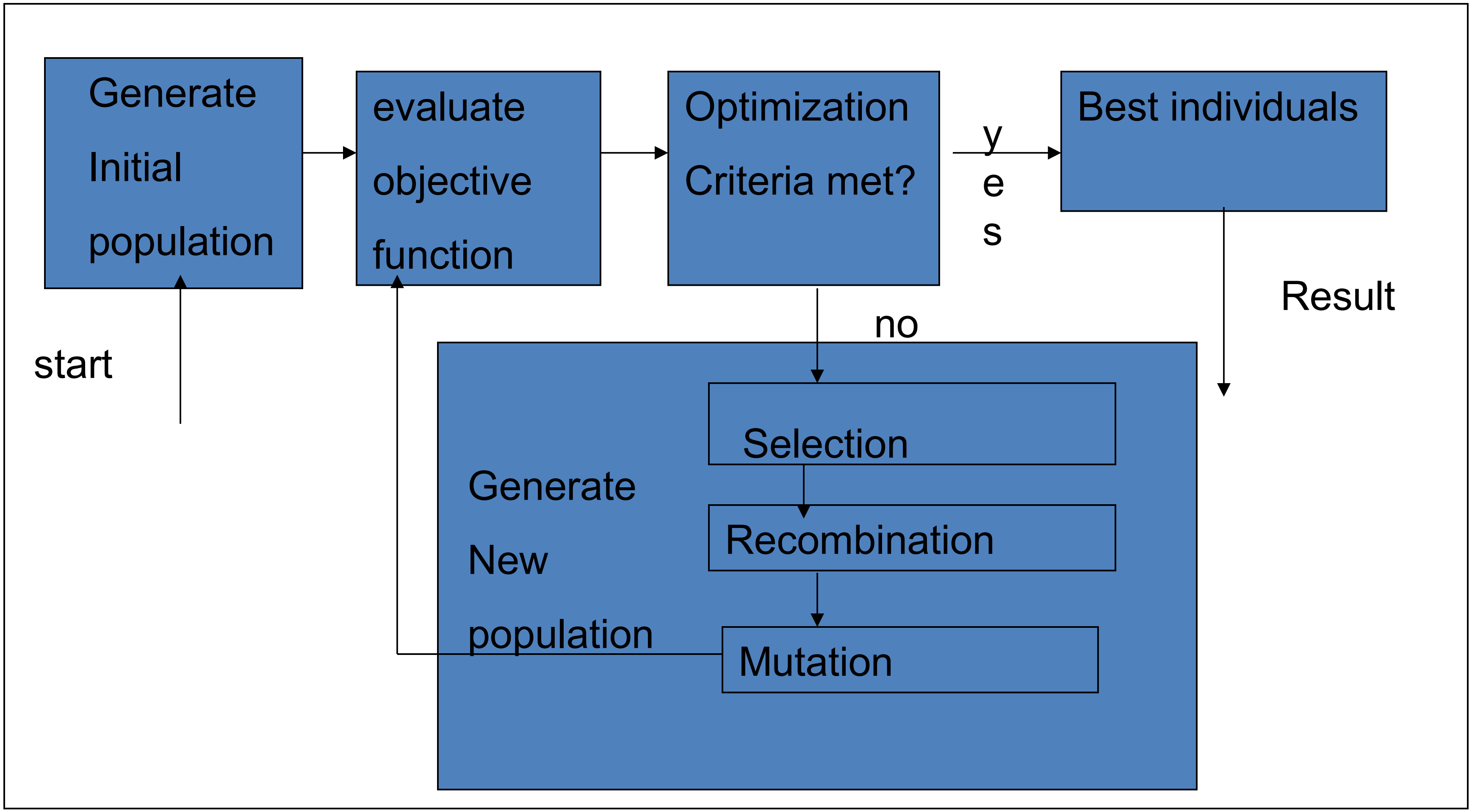
Theory of Darwinian Evolution contd..

- Some of the genes mutate (change the values from one to another randomly) due to exposure to chemicals, or biological effects

- Over a number of generations of individuals, survival and reproduction are more predominant in the population

10

Structure of Genetic algorithm



Steps in Genetic Algorithm

- Genetic algorithms are based on the theory of selection

  1.A set of random solutions are generated
- Only those solutions survive that satisfy a fitness function
- Each solution in the set is a chromosome
- A set of such solutions forms a population

12

Genetic Algorithms contd..

2. The algorithm uses three basic genetic operators namely

(i)Reproduction

(ii)crossover and

(iii) mutation along with a fitness function to evolve a new population or the next generation

- Thus the algorithm uses these operators and the fitness function to guide its search for the optimal solution
- It is a guided random search mechanism

13

Significance of the genetic operators

- Reproduction or selection by two parent chromosomes is done based on their fitness
- Reproduction ensures that only the fittest of the solutions made to form offsprings
- Reproduction will force the GA to search that area which has highest fitness values

Crossover or recombination: Crossover ensures that the search progresses in the right direction by making new chromosomes that possess characteristics similar to both the parents

14

Significance of the genetic operators contd..

Mutation: To avoid local optimum, mutation is used

It facilitates a sudden change in a gene within a chromosome allowing the algorithm to see for the solution far way from the current ones

15

Genetic operator: Reproduction

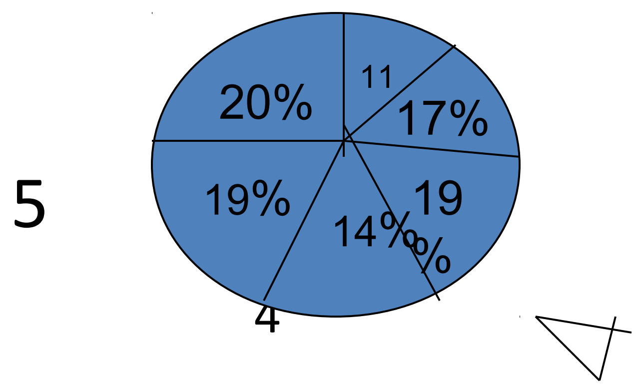
- There are many techniques for Reproduction or selection operator

1. Roulette wheel selection
2. Tournament selection
3. Ranked position selection
4. Steady state selection

1. Roulette wheel selection

- The circumference of Roulette wheel is divided into segments and marked for each string proportionate to the fitness value 6

1



-

- The wheel is spun n times; each time a segment is selected by the wheel pointer
- The segment with highest fitness value will have more probability for selection

Roulette wheel selection contd..

- In practice the procedure is as follows
- The individual fitness values are mapped to contiguous segments of a line
- Using fitness values of the string selection probability is computed using $p_i = F_i / \Sigma F_j$ $j=1,2,\ldots,n$

Roulette wheel selection contd..

- Cumulative probability for each string is computed; the i^{th} string in the population represents the cumulative probability from p_{i-1} to p_i
- A random number is generated and the string for which the random number lies in the cumulative probability range is selected for the mating pool
- The process is repeated until the desired number of individuals (called mating population) is obtained

Roulette wheel selection - Example

Table 1.1: Strings and fitness values

Number of Individual	1	2	3	4	5	6	7	8	9	10
Fitness value	2.0	1.8	1.6	1.4	1.2	1.0	0.8	0.6	0.4	0.2

Total fitness =11.0 and 2.0/11.0= 0.181~0.18

Table1.2: Computations of probabilities

Number of Individual	1	2	3	4	5	6	7	8	9	10
Selection Probability	0.18	0.16	0.15	0.13	0.11	0.09	0.07	0.06	0.04	0.02
Cumulative Probability	0.18	0.34	0.49	0.62	0.73	0.82	0.89	0.95	0.99	1.01

Roulette wheel selection – Example contd..

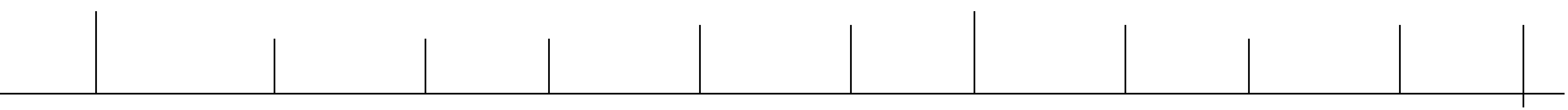
Selection Procedure:

Let the six generated random numbers be

0.81, 0.32, 0.96, 0.01, 0.65, and 0.42

1 2 3 4 5 6 7 8 9 10

0.0 0.18 0.34 0.49 0.62 0.73 0.82 0.89 0.95 0.99 1.01



0.81 falls between 0.73 and 0.82;so string 6 is selected

0.32 falls between 0.18 and 0.34;so string 2 is selected

0.96 falls between 0.05 and 0.99;so string 9 is selected

0.01 falls between 0.00 and 0.18;so string 1 is selected

0.65 falls between 0.62 and 0.73;so string 5 is selected

0.42 falls between 0.34 and 0.49;so string 3 is selected

Mating population after selection:1,2, 3,5, 6, 9

Crossover or recombination

- Crossover operator produces new offspring after combining the information contained in two parents
- Crossover is performed with a crossover probability P_c
- A random number say r is generated and compared with P_c
- If $P_c > r$ crossover occurs; otherwise the two strings are repeated without any change

Crossover methods

1. Real valued crossover

   The following methods are used to get a new offspring from two parent strings

   1.1 Discrete crossover

   1.2 Intermediate crossover

   1.3 Line crossover

2. Binary valued crossover

The following methods are used to get a new offspring from two binary encoded strings of parents selected from mating pool

   2.1 Single-point crossover

   2.2 Multi-point crossover

   2.3 Uniform crossover

   2.4 Shuffle crossover

Crossover or recombination

- Crossover operator produces new offspring after combining the information contained in two parents
- Crossover is performed with a crossover probability P_c
- A random number say r is generated and compared with P_c
- If $P_c > r$ crossover occurs; otherwise the two strings are repeated without any change

Binary valued Crossover-Single Point crossover

Single Point crossover method

- One crossover point is selected randomly between 1 and the number of bits in a chromosome
- The bits are exchanged between the parents about this point (after this point)
- Thus two new offspring are produced
- Example: Consider two parents with 11 binary variables each

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

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

Randomly selected crossover point 5

Offspring 1: 0 1 1 1 0 **1 0 0 1 0 1**

Offspring 2: **1 0 1 0 1** 0 1 1 0 1 0

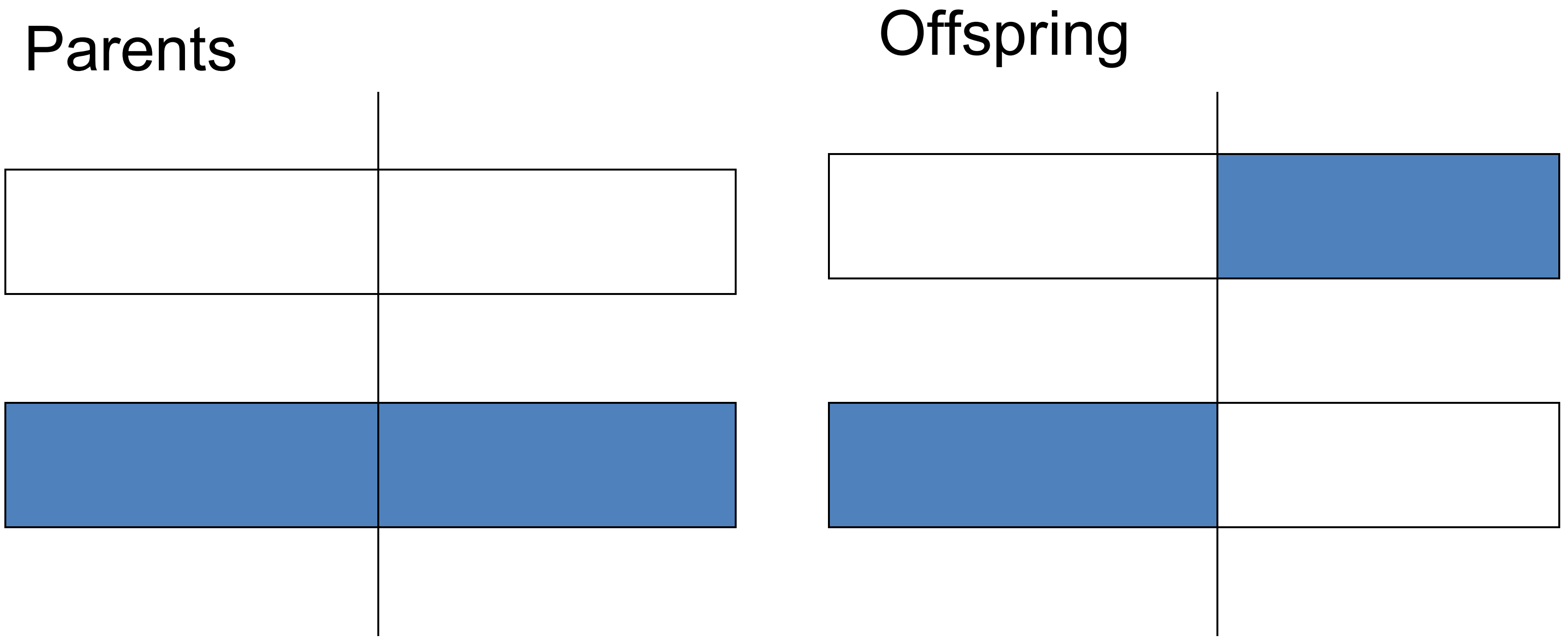


Figure 4. Single Point crossover

Mutation

- After crossover operation, the string is subjected to mutation operation
- Mutation prevents falling all solutions of the population into a local optimum
- Mutation operator alters a chromosome locally to create a better string
- Mutation causes movement in the search space and restores the lost information in the population
- Bit wise mutation is performed with a probability of mutation P_m
- A random number r is generated and compared to P_m
- If $P_m > r$, then mutation occurs; otherwise the bit is kept unchanged

Importance of mutation

- Mutation is used to maintain diversity in the population
- Consider the following strings

1110

1001

1100

1000

All the four strings have 1 in the left most position

’If the optimum solution requires a 0 in the left most position, the selection or crossover operators will not change the bit

The mutation operation will change its value

Mutation methods

1. Binary valued Mutation

   - This is widely used encoding mechanism

   - In this real value of the variables are transformed to binary codes and the genetic operators work on these coded strings

The bit-wise operators are as below

1.1 One’s complement operator

1.2 Logical Bit-wise Operator

1.3 Shift Operator

1.4 Swap Operator

1.5 Inverse Operator

2. Real valued Mutation

One's complement operator

- This unary operator causes the bits of the string to be inverted; 1 becomes 0 and 0 becomes 1
- Example:

String before mutation: 1100 1110

A random number r between 0 and 1 is selected and compared with mutation probability P_m

If $P_m > r$, then mutation occurs; otherwise the bit is kept unchanged

String after mutation : 0011 0001

with 1's complement operator

Job Shop Scheduling problem

- Suppose there are three manufacturing units M1, M2 and M3, each of which is capable of manufacturing a different product

- A schedule could be a set of numbers 20,35,15 where each number indicates the quantity of the concerned product manufactured by M1, M2, and M3 respectively

- The schedule could yield a profit say 40 computed by some equation

- The problem is to optimize (maximize) the profit
- The function used to compute the profit is called the fitness function

31

Job Shop Scheduling problem contd..

- The set {20,35,15} forms the chromosome; its members form the genes, Alleles are the values that these genes can assume

- The problem can be simplified in terms of binary numbers by assuming a unit is ON (manufacturing) or OFF (not manufacturing)

32

Job Shop Scheduling problem contd..

- A schedule { 1,0,1} yields a chromosome that indicates that M1 and M3 are on and M2 is off

The fitness function could be the binary number formed from the individual genes in the chromosome; for example the schedule {1,0,1} has fitness of $(101)_2$ or decimal 5

The problem is to find the best schedule to maximize the value of the numbers formed by the genes comprising the chromosome

The solution {1,1,1} could be found using genetic algorithm approach

33

Using GA to solve Job Shop Scheduling problem

1. Represent a solution as a chromosome and find a suitable fitness function

   Define the chromosomes like {1,0,1}, {1,1,0} etc., and define the fitness function as

   $f(\text{chromosome}) = (\text{concat}(\text{Gene1},\text{Gene2}, \text{Gene3}))_{10}$

2. Choose a population of solutions and initialize it

   Let the population size N be 4; let the initialized population be [{0,0,1}, {0,1,0}, {1,1,0}, {1,0,1}]

3. Apply the genetic operators one by one

34

3. Determine the fitness of each solution fi and the total fitness of the population ∑fi as follows

Compute the normalized fitness or fitness probability pi, and the expected count of the solution (number of solutions that could occur) E_c

The computed values are shown in Table 1.

35

Using GA to solve Job Shop Scheduling problem contd..

Solution No.	Solution	fi=concat(Gene1,Gene2, Gene3))$_{10}$	$\%pi = fi / \sum fi * 100$	$E_c = N * \%pi$ (N=4)
1	{0,1,0}	1	7.14	28.6
2	{0,1,0}	2	14.29	57.1
3	{1,1,0}	6	42.86	171.4
4	{1,0,1}	5	35.71	142.9
$\sum$		14	100	400
Maximum		6	42.86	171.43

Table 1. Fitness and expected count of a generation

Using GA to solve Job Shop Scheduling problem contd..

(b) Reproduction

Four random numbers are generated to select four chromosomes for the next generation

Let the random numbers be 265, 801, 515 and 85 Then the four chromosomes are selected based on the fitness probability and shown in Table 2. These four chromosomes are called the mating pool

Table 2.

Value generated	Selected Solution
265	{1,1,0}
801	{1,0,1}
515	{1,1,0}
85	{0,0,1}

Applying genetic operators

Reproduction or selection operator is applied on the mating pool and two solutions are selected. There are {1,0,1} and {1,1,0}

(c) Crossover is applied on {1,0,1} and {1,1,0} choosing crossing site 2 and the resultant chromosomes are {1,0,0} and {1,1,1}

(d) Mutation occurs rarely. Mutation flips the value of a randomly selected gene in a chromosome

For example the chromosome {0,1,0} could mutate to any of {1,1,0} or {0,0,0} or {0,1,1}

Applying genetic operators

- As we have not transgressed through many generations, assume that mutation will not occur
- The resultant next generation population along with fitness function are shown in Table 3.

resultant solution Si	fi
{0,0,1}	1
{1,0,0}	4
{1,1,1}	7
{1,1,0}	6
$\sum$	19
Maximum	7

Thus the solution is {1,1,1} that has highest fitness value