

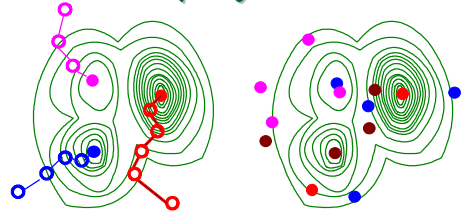
## Evolutionary Algorithms

Mohamed A. El-Sharkawi

Computational Intelligence Applications (CIA) Lab.  
 Department of EE, Box 352500  
 University of Washington  
 Seattle, WA 98195-2500

elsharkawi@ee.washington.edu  
<http://cialab.ee.washington.edu>

## Evolutionary Algorithm vs SST



**Single Search**  
 (e.g. steepest Descent)

**Genetic Search**

## Evolutionary Algorithm

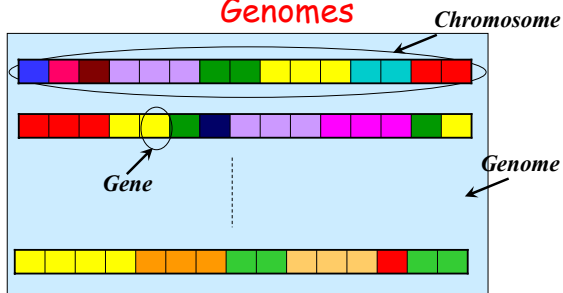


- **Single Search Technique (SST) could be trapped in undesirable minima**
- **Convergence of SST depends on the selected initial condition**

## Main Types of Evolutionary Algorithms

- **Genetic Algorithms**
- **Evolutionary Strategies**
- **Evolutionary Programming**

## Chromosome, Genes and Genomes



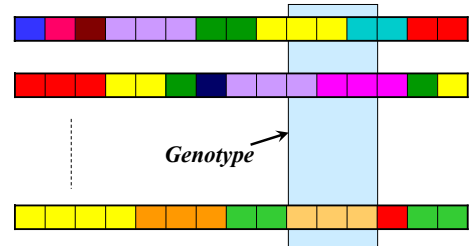
## Trail, Alleles and Locus

- **Trail:**
  - Features associated with a gene (hair color)
- **Allele:**
  - Possible settings of trail (black, blond, etc.)
- **Locus:**
  - Position of gene in the chromosome

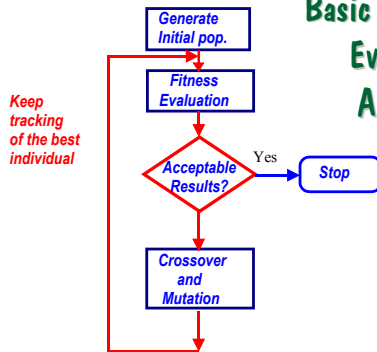
## Genotype and Phenotype

- **Genotype:**
  - Particular set of genes in a genome
- **Phenotype:**
  - Physical characteristic of the genotype (smart, beautiful, healthy, etc.)

## Genotype



## Basic Operations of Evolutionary Algorithms



## Basic Operations of Evolutionary Algorithms

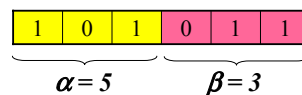
1. Selection of population pool
2. Selection of fitness function
3. Generate a new population
  - i. Selection of parents
  - ii. Cross over process
  - iii. Mutation process
4. Test
5. Loop or Stop

## Genetic Algorithm

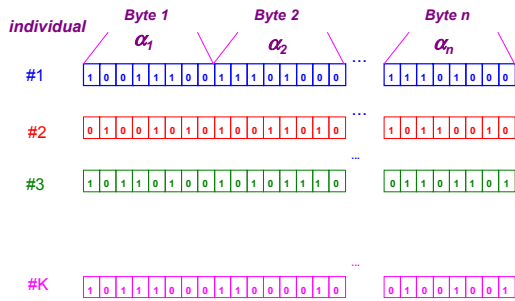
## Phenotype and Genotype

**Phenotype:**  $\alpha = 5$  and  $\beta = 3$

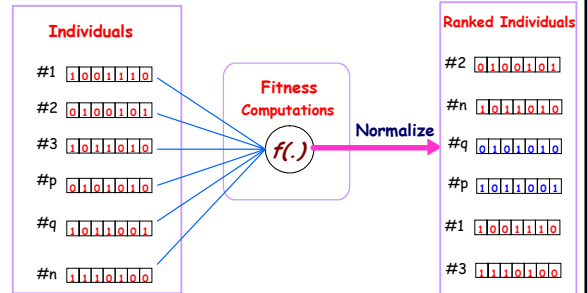
**Genotype (binary coding):**



## Coding of Population Pool



## Fitness Evaluation



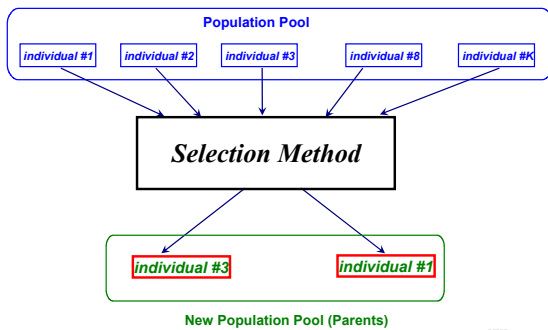
## Fitness Evaluation

Individuals	Fitness ( $f$ )	$f_{normalized} (f/f_{tot})$
1 0 1 1 1 1 0 1	.733	.144
1 1 0 1 1 0 0 0	.471	.093
0 1 1 0 0 0 1 1	.937	.184
1 1 1 0 1 1 0 0	.243	.048
1 0 1 0 1 1 1 0	.845	.166
0 1 0 0 1 0 1 0	.788	.155
0 0 1 0 0 0 1 1	.416	.082
0 0 1 1 0 1 0 1	.650	.128
Sum of Fitness ( $f_{tot}$ )	5.083	1.0

## Iteration (New Generation)

- **Parents Selection**
  - Highest fitness
  - Deterministic or probabilistic process
- **Crossover**
  - Generate new data for evaluation
- **Mutation**
  - Generate new data for evaluation

## Parents Selection



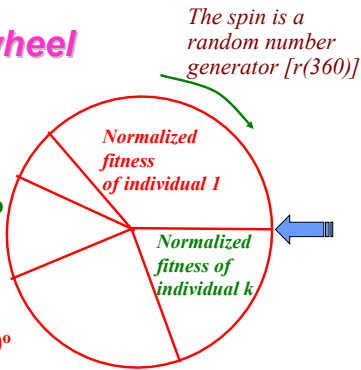
## Parents Selection

- **Selected methods**
  - Roulette wheel
  - Generation gap
  - Elitist strategy

## Roulette wheel

Size of roulette wheel's wedge, or arc, is proportional to the normalized fitness

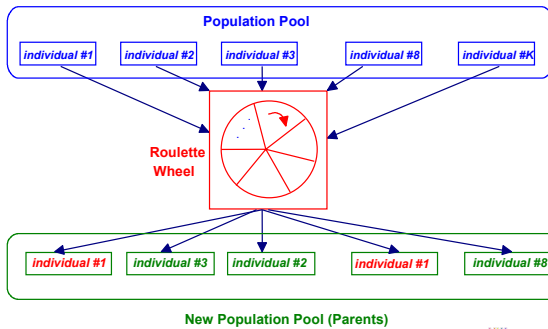
Total fitness=360°



## Roulette wheel

- Wheel is spun  $k$  times ( $k$  is the number of individuals in the population pool)
- An individual could be selected more than once
- It is possible that the best individual may not be selected

## Parents Selection using Roulette Wheel



## Generation Gap

- It is possible that an individual with the highest fitness value in a given generation may not survive the roulette wheel selection.
  - An  $x\%$  of the population with the best fitness values are kept for the crossover process
- It ensures that the population with the best fitness at each iteration are used for the cross over.

## Elitist Strategy

- Individuals with the **highest fitness** is always **copied** into the next generation.
- Unlike the generation gap technique, the individual in the elitist strategy is not changed after each generation.

## Crossover Process

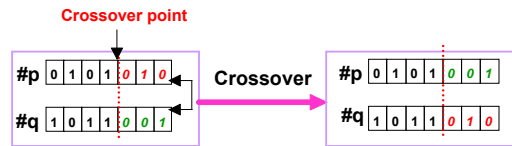
- Crossover rate
- One-point Crossover
- Two-point Crossover
- Uniform Crossover

## Crossover rate

- Selected individuals are arranged in random.
- An overall probability is assigned to the crossover process. This is called *crossover rate* (0.65-0.8).
- A random number is generated for each pair individual. If the number is lower than the crossover rate, crossover is performed.

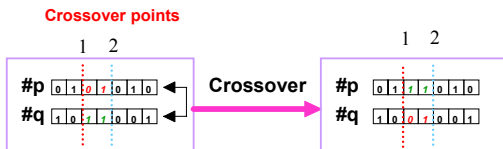
## One-point Crossover

- One crossover point is obtained by a random number generator



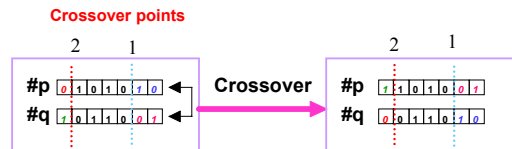
## Two-point Crossover

- Two crossover points are obtained by a random number generator



## Two-point Crossover

- The two crossover points could wrap around



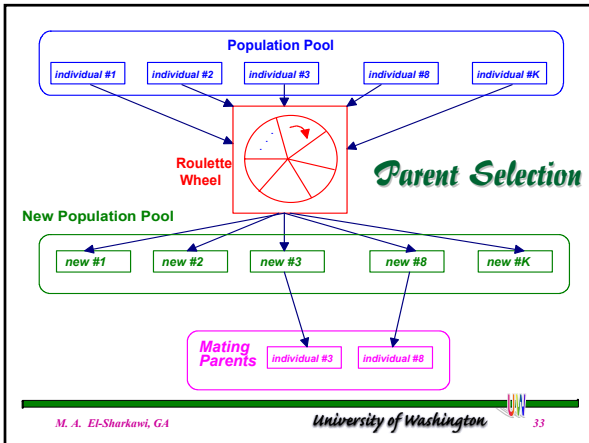
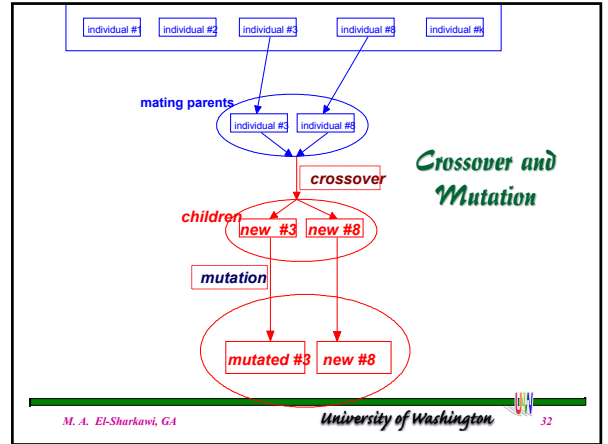
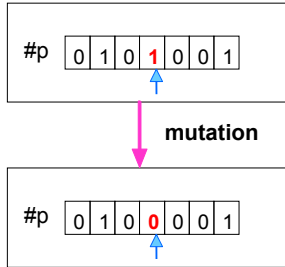
## Uniform Crossover

- A random decision is made at each bit position in the string as to whether or not to crossover bits between the parent strings.
- If the output of the random number generator is above a threshold, the bits are exchanged

## Mutation

- It is possible that the roulette wheel selects the same or similar individuals.
- This could lead to a cross over between identical or similar genes. The next generation will have genes similar to their parents.
- This stagnation can be prevented by the mutation process

## Mutation



## Comments on GA

Binary encoding may require a large # of genes to change status when a small change in the parameter is needed.

$$\begin{array}{|c|c|c|c|} \hline 0 & 1 & 1 & 1 \\ \hline \end{array} = 7$$

$$\begin{array}{|c|c|c|c|} \hline 1 & 0 & 0 & 0 \\ \hline \end{array} = 8$$

## Other Encoding Methods

### Value Encoding: Chromosome

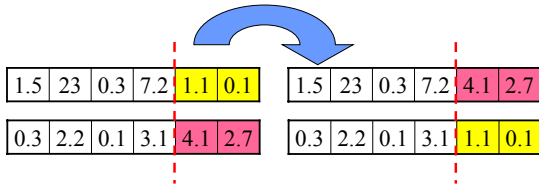
**Phenotype:**  $\alpha = 1.5, \beta = 23, \mu = 0.3, \dots$

**Chromosome:**

1.5	23	0.3	7.2	1.1	0.1
-----	----	-----	-----	-----	-----

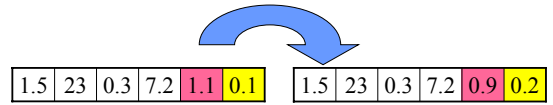
## Value Encoding: Crossover

Same as in binary encoding



## Value Encoding: Mutation

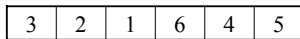
Add small noise



## Permutation Encoding: Chromosome

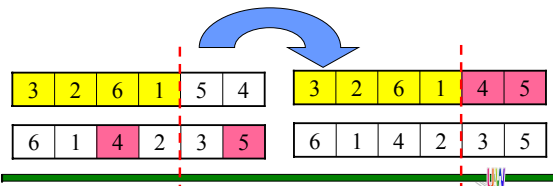
**Phenotype (Ranking):**  $\alpha = 3, \beta = 2, \mu = 1, \dots$   
Suitable for TSP type

**Chromosome:**



## Permutation Encoding: Crossover

One crossover point is selected, till this point the permutation is copied from the first parent, then the second parent is scanned and if the number is not yet in the offspring it is added



## Permutation Encoding: Mutation

Two numbers are exchanges

