Artificial Intelligence

Lab 10

Genetic Algorithms

Agenda

- Introduction
- Key terms
- Basic Genetic Algorithm
- Example
- Roulette Wheel Selection
- Crossover Operator
- Mutation Operator
- Hands On

Introduction

- A Genetic Algorithms (GAs): are search techniques used to find true or approximate solutions to optimization and search problems.

- GAs are evolutionary algorithms that use techniques inspired by evolutionary biology such as inheritance, mutation, selection, and crossover.

Introduction

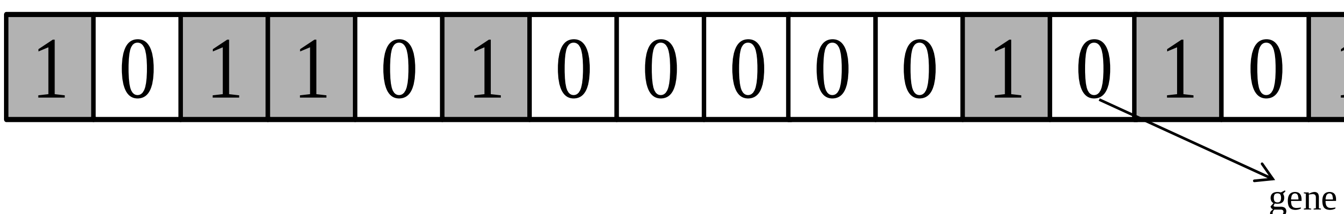
- GAs are implemented as a computer simulation in which a population of abstract representations (chromosomes) of candidate solutions (individuals) to an optimization problem evolves toward better solutions.
- In GAs, optimization iteratively improves the quality of solutions until an optimal, or near optimal, solution is found.
- As in nature, only the fittest species can survive, breed, and thereby pass their genes on to the next generation.
- Traditionally, solutions are represented in binary as strings of 0s and 1s, but other encodings are also possible.

Introduction

- The evolution usually starts from a population of randomly generated individuals.
- In each generation, the fitness of every individual in the population is evaluated, multiple individuals are selected from the current population based on their fitness, and modified (recombined and possibly mutated) to form a new population.
- The new population is then used in the next iteration of the algorithm.
- The algorithm terminates when a maximum number of generations has been produced.

Key terms

- Individual: Any possible solution
- Population(Search Space): All possible solutions to the problem
- Chromosome: A representation of a solution as an array of bits.
- Gene: Each chromosome consists of a number of "genes", and each gene is represented by 0 or 1.



A 16-bit binary string of an artificial chromosome

Key terms

- Fitness function: is used to measure the quality of a solution.
- The crossover operator: exchanges parts of two single chromosomes.
- The mutation operator: changes the gene value in some randomly chosen location of the chromosome.

Basic Genetic Algorithm

1. Represent the problem solution as a chromosome of a fixed length.
2. Choose the size of population N, the crossover probability P_c and the mutation probability P_m.
3. Define a fitness function to measure the fitness of an individual chromosome in the problem domain.
4. Randomly generate an initial population of chromosomes of size N: $X_1, X_2, \ldots, X_n$.
5. Calculate the fitness of each individual chromosome: f(X_1), f(X_2), . . . , f(X_n).

Basic Genetic Algorithm

6. Select a pair of chromosomes for mating from the current population. Parent chromosomes are selected with a probability related to their fitness.
7. Create a pair of offspring chromosomes by applying the genetic operators (crossover and mutation).
8. Place the created offspring chromosomes in the new population.
9. Repeat Step 6 until the size of the new population becomes equal to the size of the initial population (N).

Basic Genetic Algorithm

10. Replace the parent chromosome population with the new offspring population.
11. Go to Step 5, and repeat the process until the termination criterion is satisfied.

- After certain number of generations, we examine the best chromosomes in the population. If no satisfactory solution is found, the GA is restarted.

Example

- Find the maximum value of the function $(15X - X^2)$ where parameter X varies between 1 and 15.
- For simplicity, we may assume that X takes only integer values.
- Thus, chromosomes can be built with only four genes:

Integer	*Binary code*	*Integer*	*Binary code*	*Integer*	*Binary code*
1	0 0 0 1	6	0 1 1 0	11	1 0 1 1
2	0 0 1 0	7	0 1 1 1	12	1 1 0 0
3	0 0 1 1	8	1 0 0 0	13	1 1 0 1
4	0 1 0 0	9	1 0 0 1	14	1 1 1 0
5	0 1 0 1	10	1 0 1 0	15	1 1 1 1

Example

- The size of the chromosome population N is 6.
- The crossover probability P_c equals 0.7.
- The mutation probability P_m equals 0.001.
- The fitness function is defined by: F $(X) = 15X - X^2$
- The GA creates an initial population of chromosomes by filling six 4-bit strings with randomly generated ones and zeros.

Chromosome label	*Chromosome string*	*Decoded integer*
X1	1 1 0 0	12
X2	0 1 0 0	4
X3	0 0 0 1	1
X4	1 1 1 0	14
X5	0 1 1 1	7
X6	1 0 0 1	9

Example

- The next step is to calculate the fitness of each chromosome.

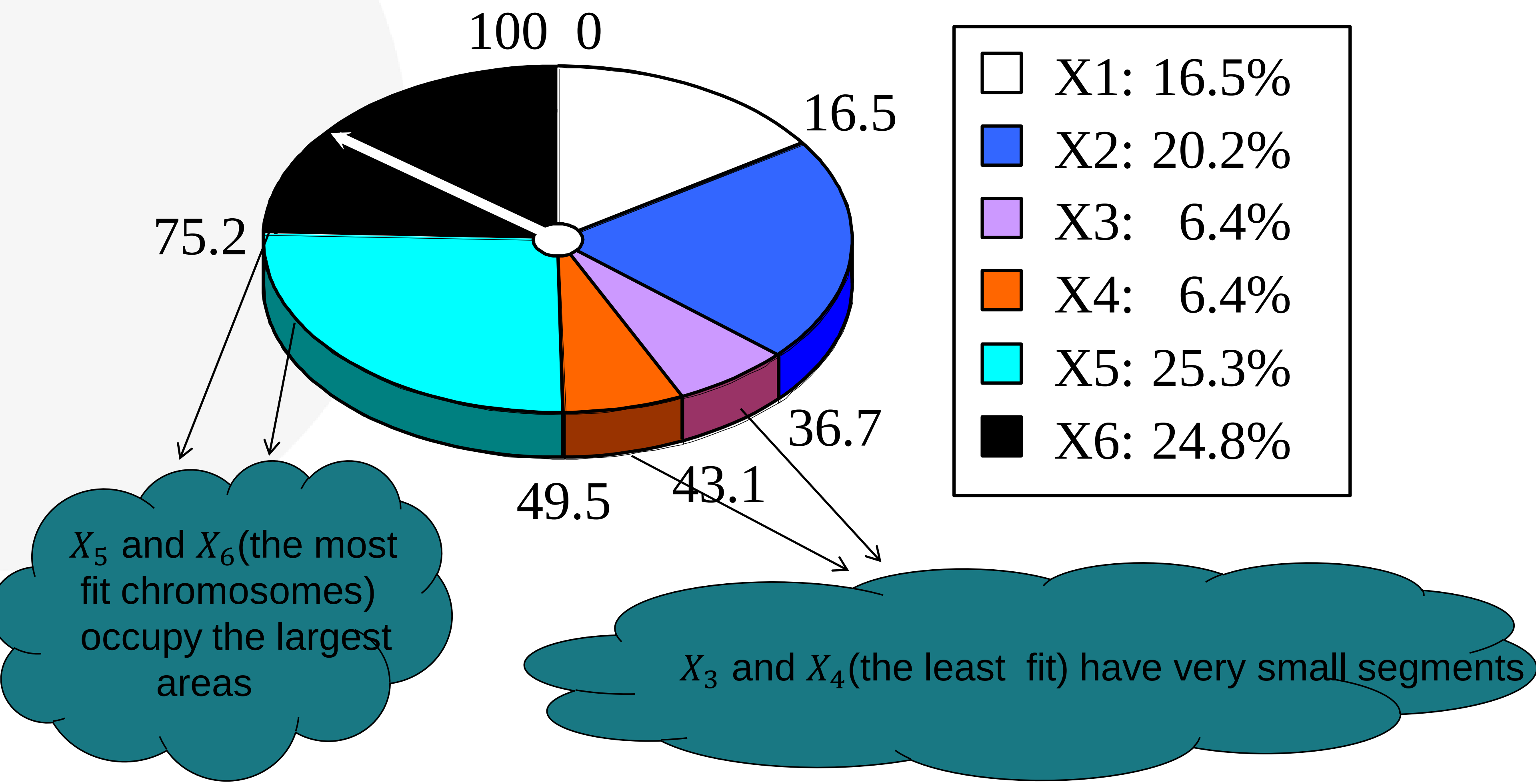
Chromosome label	*Chromosome string*	*Decoded integer*	*Chromosome fitness*	*Fitness ratio, %*
X1	1 1 0 0	12	36	16.5
X2	0 1 0 0	4	44	20.2
X3	0 0 0 1	1	14	6.4
X4	1 1 1 0	14	14	6.4
X5	0 1 1 1	7	56	25.7
X6	1 0 0 1	9	54	24.8

- Thus, the chromosomes X_5 and X_6 have a high chance, while the chromosomes X_3 and X_4 have a very low probability of being selected.
- So, The average fitness improves from one generation to the next. (The average fitness of the initial population is 36)

Roulette Wheel Selection

- The most commonly used chromosome selection techniques is the roulette wheel selection.
- Each chromosome is given a slice of a circular roulette wheel proportional to its fitness.
- To select a chromosome for mating, a random number is generated in the interval [0,100], and the chromosome whose segment spans the random number is selected.

Roulette Wheel Selection



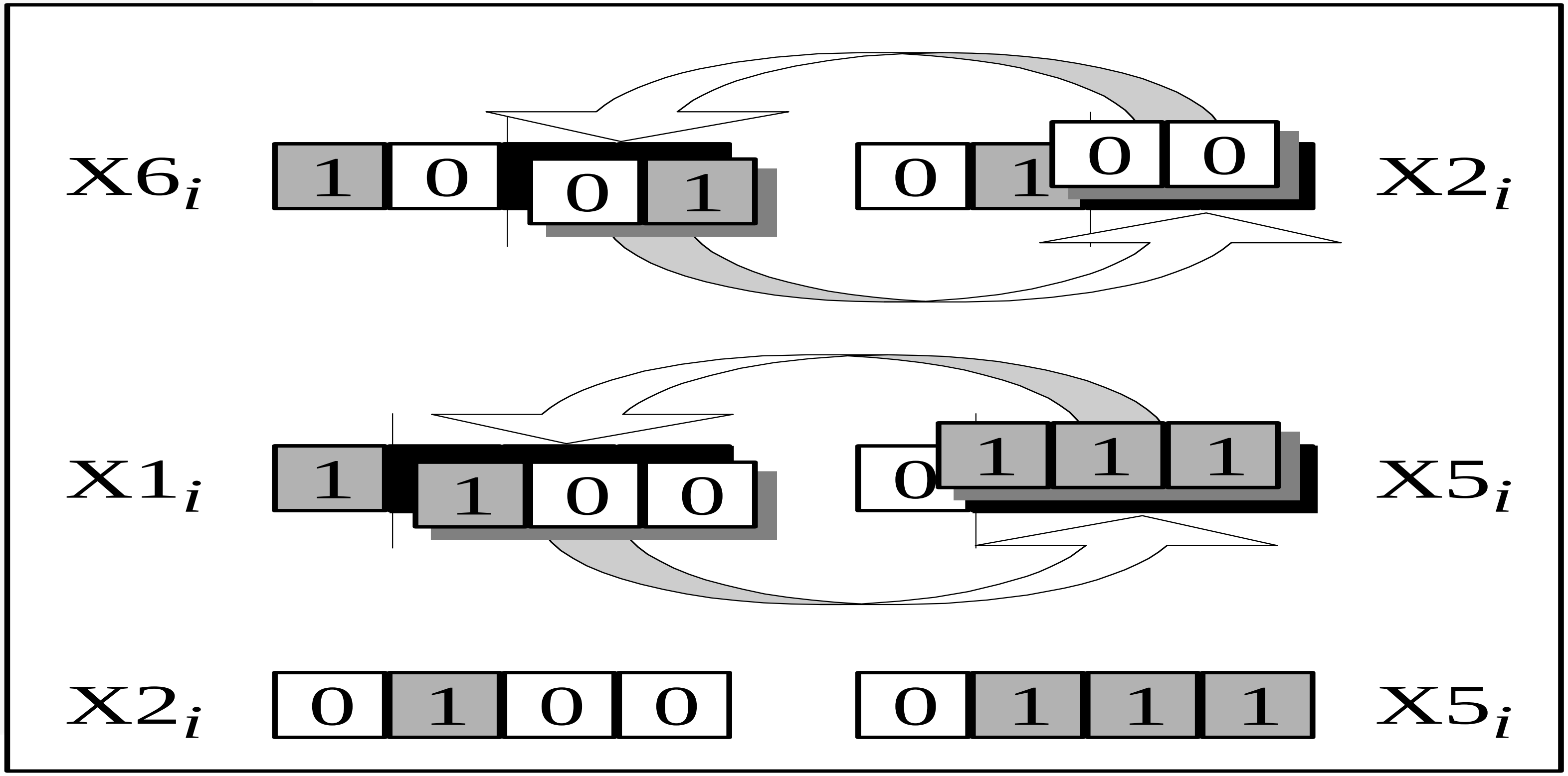
Roulette Wheel Selection

- In our example, we have an initial population of 6 chromosomes.
- Thus, we apply roulette wheel selection three times.
- The first two selected chromosomes are X_6 and X_2 to become parents.
- The second selected pair are chromosomes X_1 and X_5.
- The last two selected chromosomes are X_2 and X_5.
- Once a pair of parent chromosomes is selected, the crossover operator is applied.

Crossover Operator

- First, the crossover operator randomly chooses a crossover point where two parent chromosomes break, and then exchanges the chromosome parts after that point. As a result, two new offspring are created.
- For example, the chromosomes X_6 and X_2 could be crossed over after the second gene in each to produce the two offspring.
- If a pair of chromosomes does not crossover, then the offspring are created as exact copies of each parent.
- For example, the parent chromosomes X_2 and X_5 may not crossover. Instead, they create the offspring that are their exact copies.

Crossover Operator

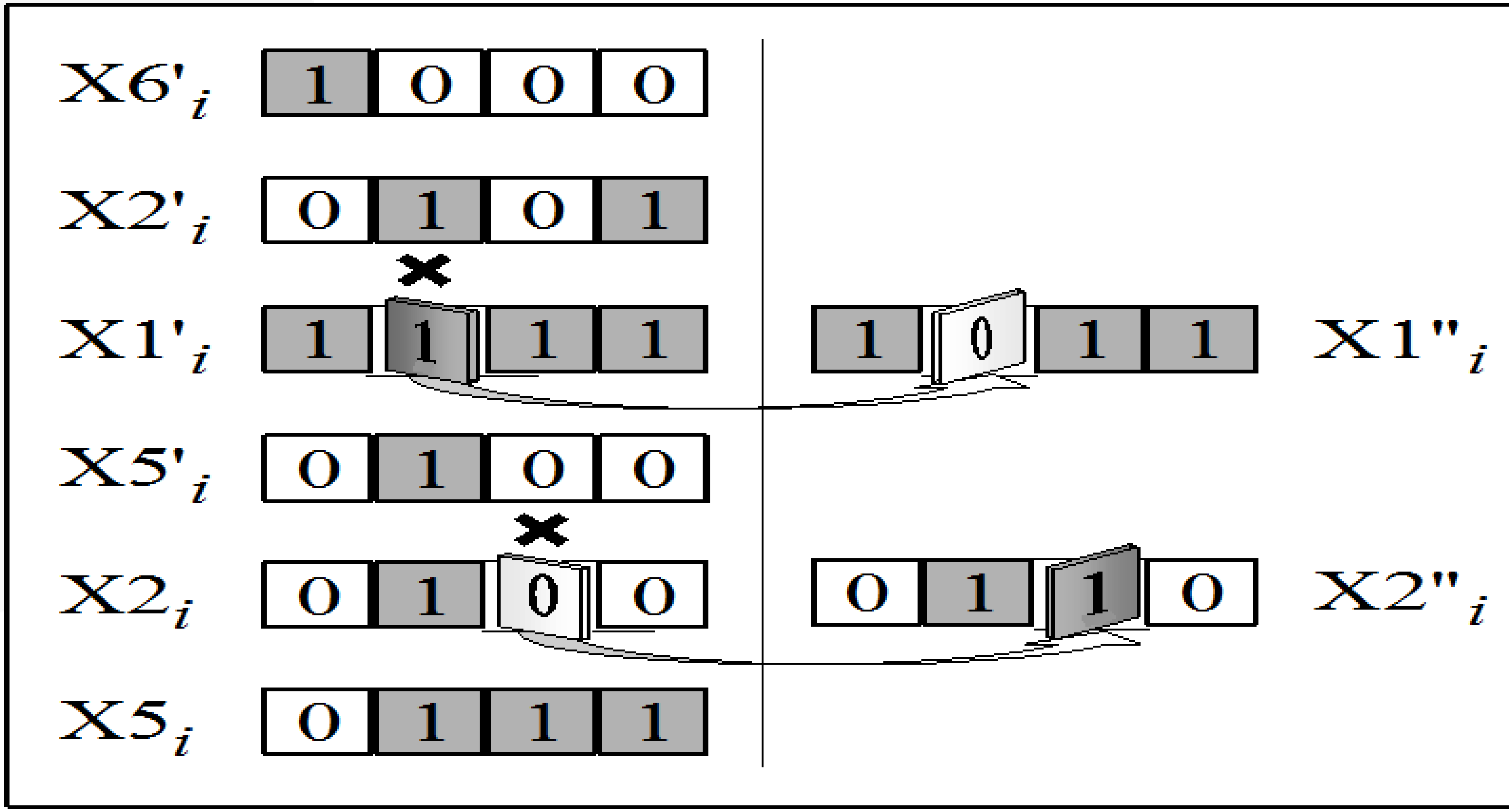


- After selection and crossover, the average fitness of the chromosome population has improved from 36 to 42.

Mutation Operator

- Mutation, which is rare in nature, represents a change in the gene. It may lead to a significant improvement in fitness, but more often has rather harmful results.
- Its role is to provide a guarantee that the search algorithm is not trapped on a local optimum.
- The mutation operator flips a randomly selected gene in a chromosome.
- The mutation probability is quite small in nature, and is kept low for GAs, typically in the range between 0.001 and 0.01.

Mutation Operator



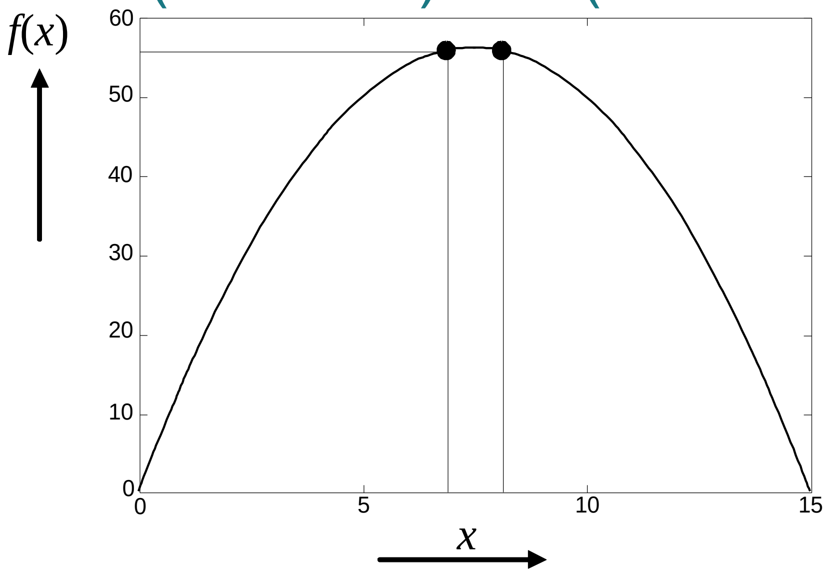
The Second Generation

Generation $(i + 1)$

$X1_{i+1}$	1	0	0	0	$f = 56$
$X2_{i+1}$	0	1	0	1	$f = 50$
$X3_{i+1}$	1	0	1	1	$f = 44$
$X4_{i+1}$	0	1	0	0	$f = 44$
$X5_{i+1}$	0	1	1	0	$f = 54$
$X6_{i+1}$	0	1	1	1	$f = 56$

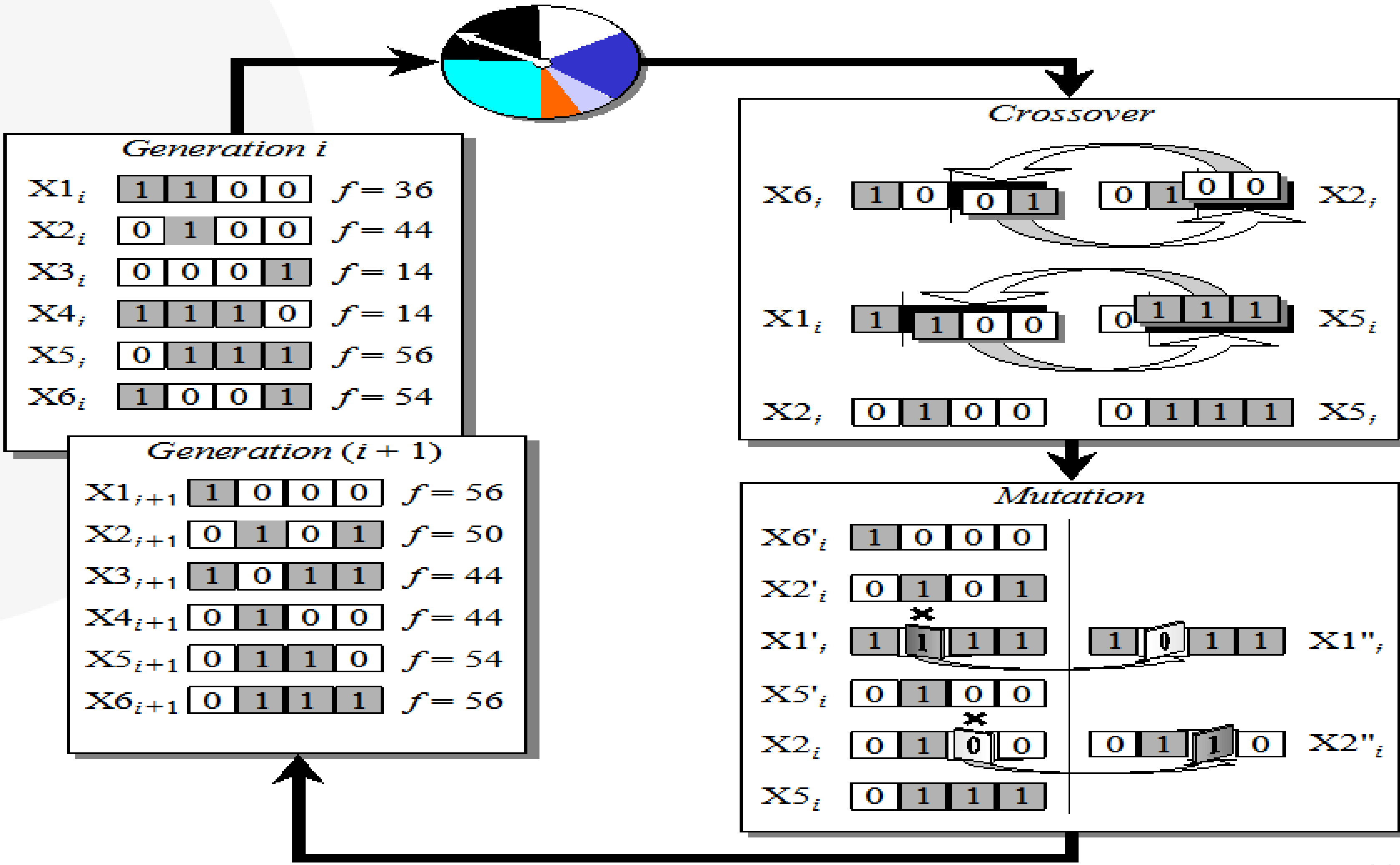
The Final Generation

- Genetic algorithms guarantees the continuous improvement of the average fitness of the population.
- After a number of generations (typically several hundreds), the population evolves to a near-optimal solution.
- In our example, the final population would consist of only chromosomes (0111 = 7) and (1000 = 8)



Chromosome final locations

The Genetic Algorithm Cycle



Hands On

Implement a genetic algorithm that:

- Starts with a base population of randomly generated strings.
- Iterates over a certain number of generations while implementing 'natural selection‘.
- Prints out the most fit string.
- The optimal string is **"Hello, World”.**
- Each chromosome has the same length as the optimal string.
- The population size is 20 individuals.
- The number of generations is 5000.
- P_c equals 1 P_m equals 0.01.

Questions?