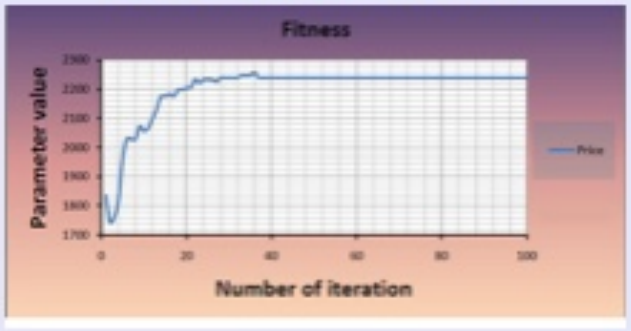
# What else?

# Knapsack Problem

Price and size of each item.

| Number of item | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Item's price | 77 | 88 | 67 | 14 | 36 | 10 | 68 | 36 | 87 | 68 | 54 | 45 | 53 | 77 | 36 | 76 | 9 | 39 | 84 | 22 |
| Item's weight | 39 | 5 | 74 | 12 | 84 | 16 | 87 | 83 | 34 | 50 | 39 | 66 | 89 | 76 | 5 | 79 | 13 | 4 | 73 | 13 |

Fitness vs. Generation.




The best combination with maximum total price.

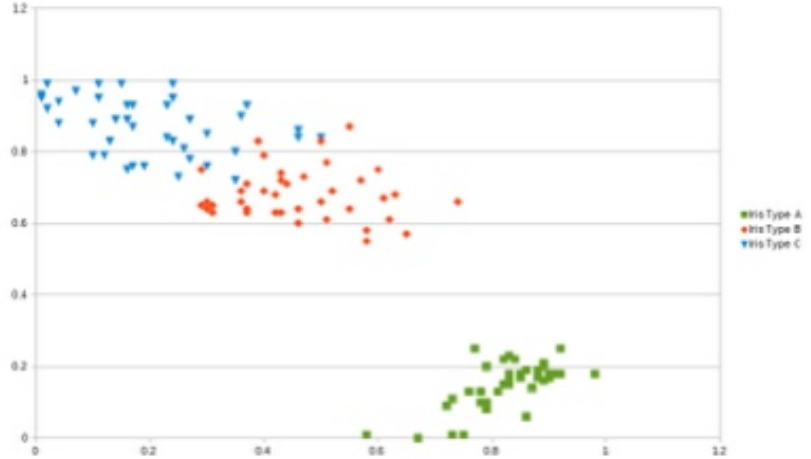
| Item | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Number of items | 4 | 5 | 0 | 2 | 0 | 1 | 0 | 0 | 4 | 5 | 4 | 0 | 0 | 1 | 4 | 0 | 0 | 4 | 1 | 4 |
| Item's price | 308 | 440 | 0 | 28 | 0 | 10 | 0 | 0 | 348 | 340 | 216 | 0 | 0 | 77 | 144 | 0 | 0 | 156 | 84 | 88 |

Yulia Lishko (2014)

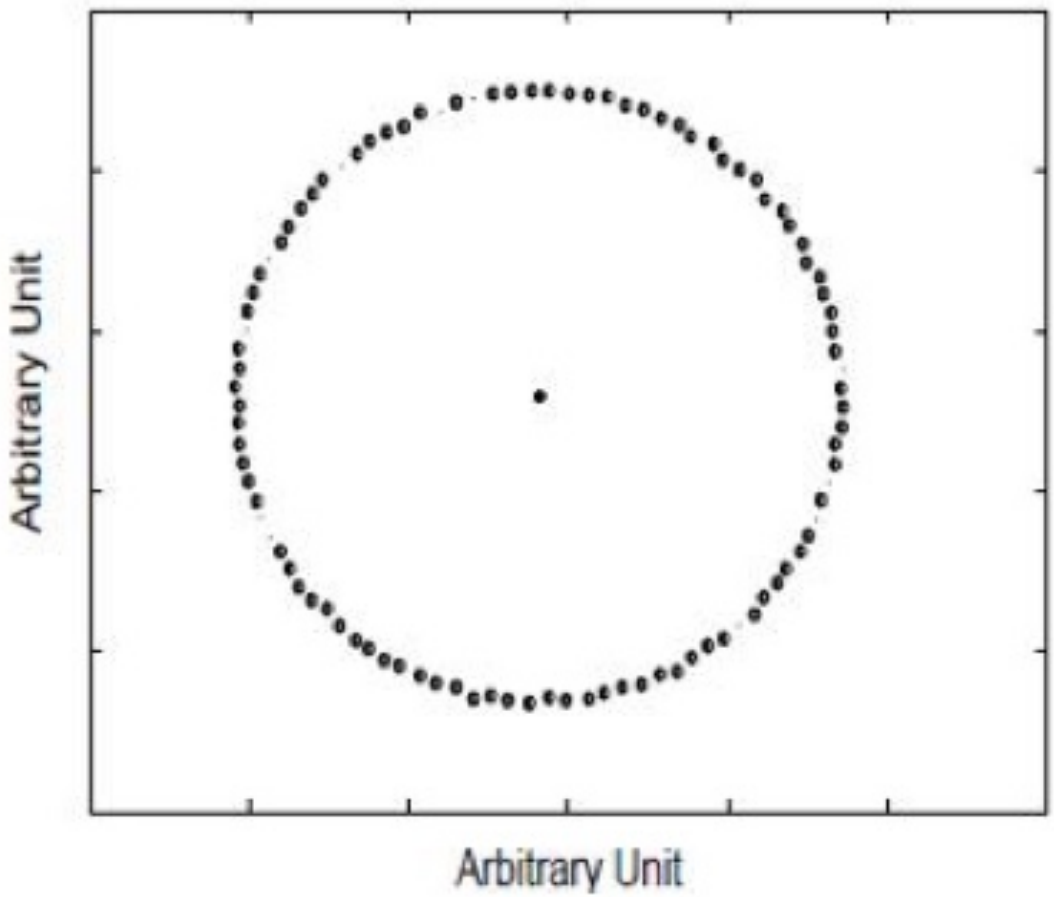
# Dimension Reduction

## (1) A classification of iris flowers

| $x_1$ | $x_2$ | $x_3$ | $x_4$ | class |
|---|---|---|---|---|
| 5.1 | 3.5 | 1.4 | 0.2 | 1 (Setosa) |
| 4.9 | 3.0 | 1.4 | 0.2 | 1 (Setosa) |
| 4.7 | 3.2 | 1.3 | 0.2 | 1 (Setosa) |
| ... | ... | ... | ... | ... |
| 7.0 | 3.2 | 4.1 | 1.4 | 2 (Versicolor) |
| 6.4 | 3.2 | 4.5 | 1.5 | 2 (Versicolor) |
| 6.9 | 3.1 | 4.9 | 1.5 | 2 (Versicolor) |
| ... | ... | ... | ... | ... |
| 5.8 | 2.7 | 5.1 | 1.9 | 3 (Virginica) |
| 7.1 | 3.0 | 5.9 | 2.1 | 3 (Virginica) |
| 6.3 | 2.9 | 5.6 | 1.8 | 3 (Virginica) |
| ... | ... | ... | ... | ... |



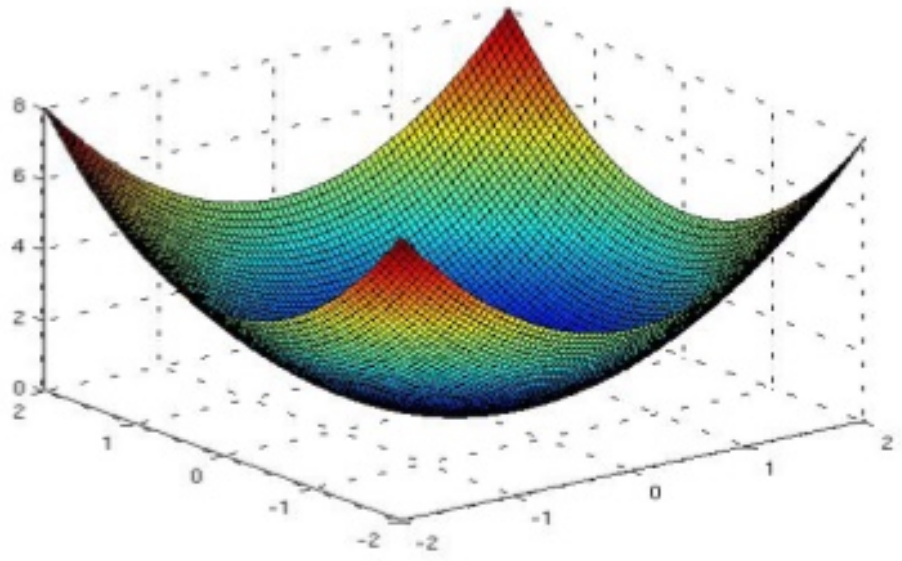
## (2) 100 points
## from surface of hyper sphere
## in 49^2 = 2401 dimensional space



# 3-D Function Optimization

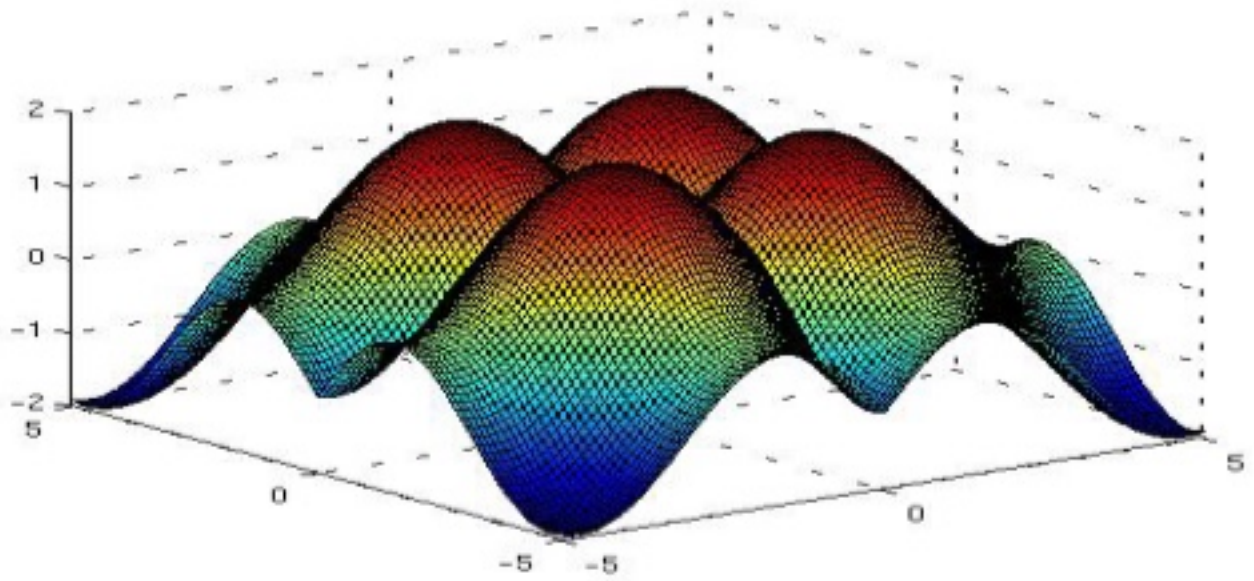
## (1) Sphere model

$$z = x^2 + y^2$$

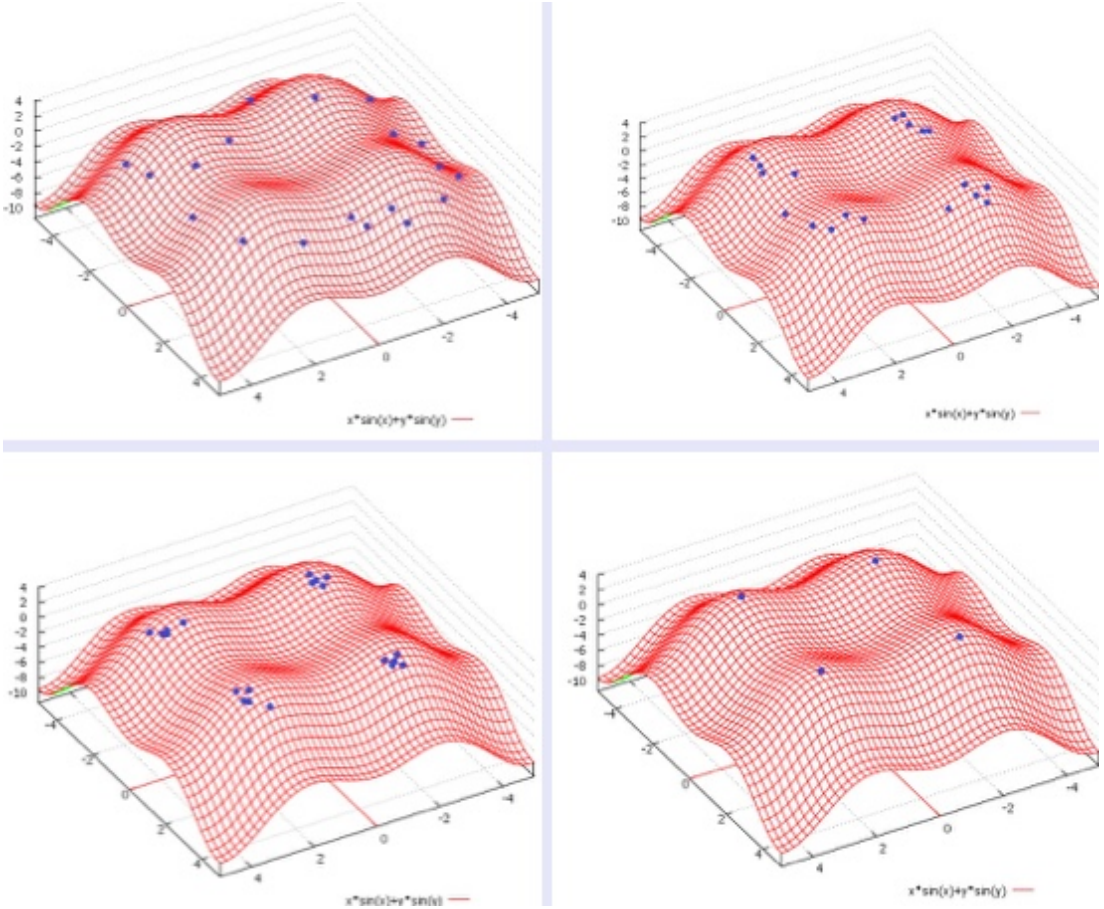


## (2) 3-D Schwefel function

$$z = x\sin(|x|) + y\sin(|y|)$$



# (3) An example



# Random Mutation Hill-climbing

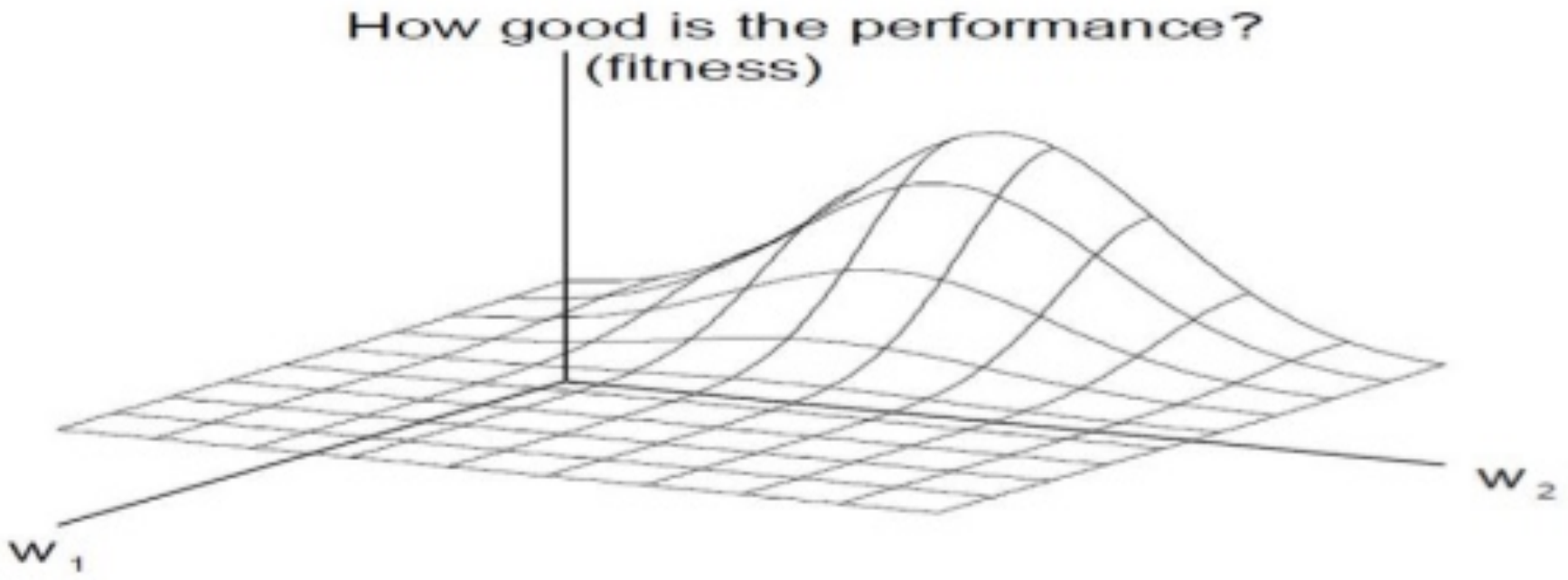
(1) choose a string at random and call this current-hilltop

(2) choose a locus at random to flip. If the flip leads to an equal or higher fitness then set current-hilltop to the resulting string

(3) goto step (2) until an optimum string has been found or until a maximum number of evaluations have been performed.

(4) return the current-hilltop

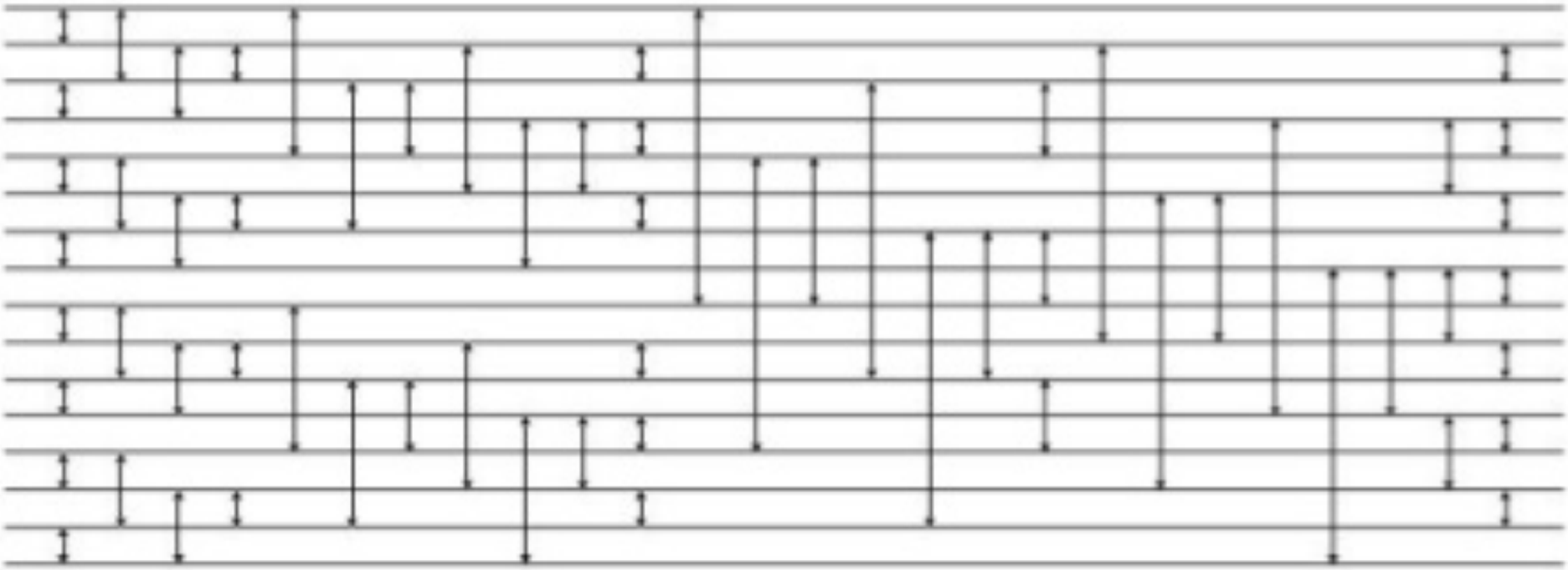
# A conceptual plot of fitness value defined on a fictitious 2-D space



# A needle in a hay stack problem

# Ex. Sorting algorithm by Knuth et. al (1964)

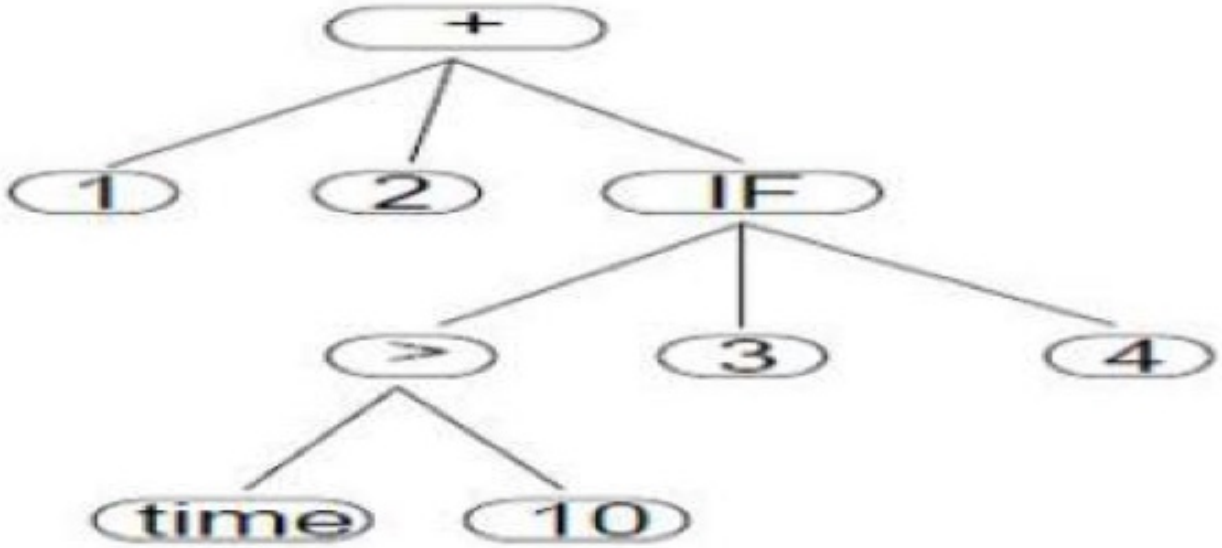
## 63 Comparisons



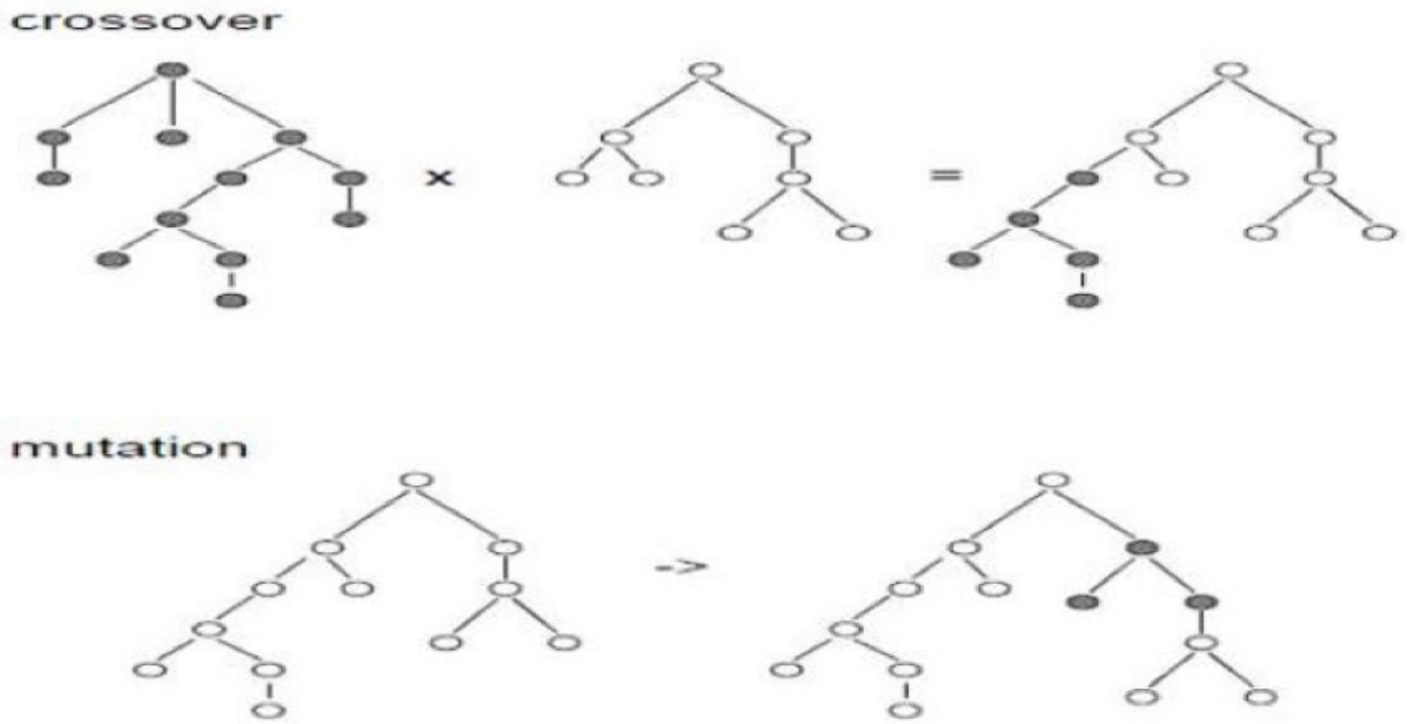
# Evolution of Tree Structure

Prgramming in LISP which can be represented by tree such as

```
(+ 1 2 (IF (> time 10) 3 4))
```



# Its crossover & mutation



# Evolution under two Fitness Functions