ISTANBUL UNIVERSITY –
JOURNAL OF ELECTRICAL & ELECTRONICS ENGINEERING

YEAR : 2006 (245 - 250)
VOLUME : 6
NUMBER : 2

COMPUTER NETWORK OPTIMIZATION USING GENETIC ALGORITHM

Olcay AKBULUT[1] Onur OSMAN[2] Osman N. UCAN[3]

[1,3]Istanbul University, Engineering Faculty, Department of Electrical and Electronics Engineering, 34850, Avcılar, Istanbul,TURKEY
[2]Istanbul Commerce University, Engineering Faculty, Vocasional School, Istanbul, Turkey.

[1]E-mail: oakbulut@istanbul.edu.tr [2]E-mail: oosman@iticu.edu.tr [3]E-mail: uosman@istanbul.edu.tr

ABSTRACT

In this paper, Genetic Algorithm (GA) is proposed as optimization software to find the shortest path of various computer networks. It deals with different method concerning the placement routers, routes the packages. The genetic based algorithm defines an optimum way when a computer network system is constructed. Genetic Algorithm gives better results regarding other classical methods as the number of nodes of the network increases.

Keywords: *Genetic Algorithm, node number, optimization.*

1. INTRODUCTION

Computer network is one of the fastest growing technologies today. A computer network is called a group of computers linked in such a way that they can send information back and forth between themselves. The link can be established with copper, fiber cables, radio-link systems or infrared for the short ranges. In this study the shortest distance or optimum cost is calculated using genetic algorithm for the all possible combination of paths starting at a node and ending at a final node. At the beginning stage, matrixes representing distances between nodes and cost for each node is filled with random values.

2. METHOD

Genetic algorithms constitute the inceasingly large part of evolutionary calculation techniques, which form the artificial intelligence. As it's obvious in its name, genetic algoritm, forming evolutionary calculation technique, inspired from evolution theory of Darwin. Any kind of problems which involves genetic algoritm will be solved through the application of artificial evolution technique. Genetical algorithm is used to solve problems that are impossible or hard to be solved by applying conventional methods [4-6]. In general terms, genetical algorithm has three field of application. These are; optimization, practical industrial applications, and categorization systems.

Genetic Algorithm starts with a set of solution, which is identified with chromosomes and known as population. Resolutions that has come out from a population is applied to the next one with the expectation of positive improvements. The selected group is used for creation of a new population according to their compatibility. Nevertheless, it's likely that the compatible ones produces better solutions. This would be continued until the expected solutions will be obtained.

Received Date : *10.01.2004*
Accepted Date: *23.06.2006*

The phases for genetic algorithm:
1. Introduction: creation of a population which includes n number of chromosomes (appropriate solution of the problem)
2. Compatibility: evaluation of each x chromosomes with f (x)
3. New population: repetition of the following steps until a new population have been emerged.
- Selection: selection of two parental chromosomes according to compatibility (increases the chance for better compatibility)
Crossover: To create a new member, parents are cross-fertilized according to possibility of crossover. If cross-fertilization is not applied, new member will be a copy of a mother or father.
-Mutation: the place of the new member will be changed according to the possibility of fission.
- Addition: adding the new member to the new population
4. Alteration: Using the new population that emerged when algorithm is re-applied.
5. Test: If the result is convincing, algorithm will be concluded and the last population will be presented as the solution.
6. Cycle: Return to the second step.
As seen above, the structure of the genetic algorithm is quite general and can be applied to any kind of problem. Identification of chromosomes is generally done by the numbers in double set. Members who are used for the crosswise must be selected among the best ones. If a problem will be solved with the application of GA, the user would decide the completion of the algorithm. There is no completion criterion of GA. Having a satisfactory result or guaranteeing the convergence could be used as criteria for completion of the algorithm.

The most important parts of GA are the processes of crosswise and mutation. These processes are started with a unit of probability, and in most of the cases applied randomly. This helps to get satisfactory results.

The representation of a chromosome with double numbers:
Chromosome1 1101100100110110,
Chromosome2 1101111000011110

A chromosome should include information on solution that it represents. Each chromosome is set up with binary series. Each number that named bit in this series can represent a characteristic of the solution. Or, a serial, on its own, would indicate a number. Expressing the chromosomes with the set of numbers in the binary series is the most common representation form; however, also integer and real numbers can be used. The reason for selecting for binary series is as the following: first of all it is simple, and secondly, it is processed by the computer easier and faster.

The reproduction process is a process which is applied according to certain selection criteria to reproduce new generation. A selection criterion takes the compatibility as a basis and selects the compatible members. At later stage, it is possible that more compatible new members will emerge from those members who will be subjected to crosswise and fission. All members may be selected in terms of compatibility or some are selected randomly and transferred to next generation.

Crossover can be applied after the decision for representation of chromosomes taken. Crossover is a process which is applied through the deduction of some genes from parents to create new members.

Chromosome1	11011 | 00100110110,
Chromosome2	11011 | 11000011110
Member1	11011 | 11000011110,
Member2	11011 | 00100110110

A position for crossover is selected randomly. Newly emerged member would have taken some characteristics of parents, and becomes their copy. Crossover process can also be applied in other ways. For instance, more than one crossover point can be selected. In order to reach to a better performance, different crossover methods can be applied. Mutation will be realized following the realization of crossover. Mutation is applied in order to prevent new solutions to copy previous solution and to reach a solution faster. Mutation randomly changes the bit of the new member (if it was identified in binary serial).

Original member1	1101111000011110,
Original member2	1101100100110110
Transformed member1	1100111000011110,
Transformed member2	1101101100110110

Some of the members which has the highest compatibilities after these processes, are transferred to new generation to keep the life of best members. This is named as elitism.

GA technic has two parameters which has the possibility of crosswise and mutation.

The possibility of crossover determines how often crossover will be applied. If new crossover is not applied (the possibility of crosswise is 0%), new members becomes the copy of ex-members but, this does not mean new generation will be the copy of old generation. When this percentage is 100%, new members are entirely obtained by crosswise. Crossover is applied with the expectation to obtain better new members that are created by deducting the positive sides of the old members.

Mutation probability indicates the frequency of the mutation to be performed. If mutation doesn't occur, a new crossbreed or copy remains as it is. If mutation does occur, a part of the new individual will be altered. If this ratio is 100%, offspring in that generation change completely. If it is 0%, without any transformation, GA technique will contain other parameters as well. The size of the population is one of the most significant of the factors. This parameter denotes the number of chromosomes, entity, (solely in one generation) among the population. In the case the number of the chromosomes is not sufficient, GA can travel merely some part of the space, and there will be no option for crossbreeding. GA functions in much slower fashion, if the number of the chromosomes is too many. It is discovered from the researches that beyond a particular point, there is no use in increasing the population.

The incoming individuals may be chosen randomly or based on adaptability/harmony. In the case the new arrivals are picked randomly, the convergence may be getting harder. In the case all individuals are selected based on harmony/adaptability, in the new generation, regional convergences may come into place. In order to overcome these issues, the selection can be made partially based on adaptability, and partially randomly. This ratio can be explained as with the Generational Difference. When the Generational Difference is 100%, all the new individuals are decided based on the adaptability criterion.

There is a need of selection of individuals to constitute parents. According to theory the fittest individuals must be survive to leave descendants. This selection can be based on several criteria. Examples are Roulette selection, Boltzman selection, tournament selection, sorted selection [1-6].

3. APPLICATION OF COMPUTER NETWORKS

We have applied GA for computer networks with 10x10 and 20x20 nodes. For n=10, simulation parameters are found as in Table 1.

Table1. *GA simulation parameters*

Simulation parameters	
Population growth	40
Node number	10
Iteration number	300
Mutation rate	0,3

At the beginning weight coefficients between nodes are selected randomly as in Equation (1).

$$A=\begin{bmatrix} 0 & 2 & 3 & 1 & 4 & 5 & 7 & 8 & 10 & 12 \\ 2 & 0 & 1 & 4 & 5 & 7 & 8 & 10 & 12 & 6 \\ 3 & 1 & 0 & 5 & 7 & 8 & 10 & 12 & 6 & 9 \\ 1 & 4 & 5 & 0 & 8 & 0 & 12 & 6 & 9 & 11 \\ 4 & 5 & 7 & 8 & 0 & 12 & 6 & 9 & 11 & 17 \\ 5 & 7 & 8 & 10 & 12 & 0 & 9 & 11 & 17 & 4 \\ 7 & 8 & 10 & 12 & 6 & 9 & 0 & 17 & 4 & 8 \\ 8 & 10 & 12 & 6 & 9 & 11 & 17 & 0 & 8 & 3 \\ 10 & 12 & 6 & 9 & 11 & 17 & 4 & 8 & 0 & 1 \\ 12 & 6 & 9 & 11 & 17 & 4 & 8 & 3 & 1 & 0 \end{bmatrix} \quad (1)$$

Here, nodes are represented by row and columns and the weights are labelled as matrix coefficients. In Table 2, simulation results and corresponding total cost is given in Table 2.

Tablo 2. *GA computer simulation results*.

Iteration number	Total cost
1	50
7	42
17	38
20	34
182	33

The optimum path is found as in Table 3, after 182 iterations.

Tablo 3. *The optimum way in 182nd iteration*

Optimum way									
6	8	4	1	3	2	7	10	5	9

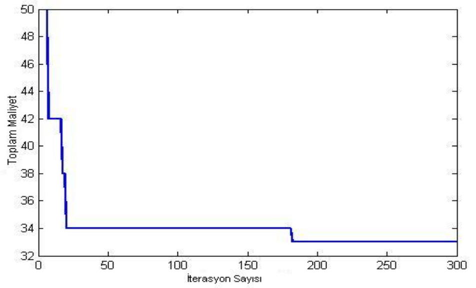

Fig. 1. *Total cost graphic according to iteration number,*

For n=20x20 nodes, GA simulation parameters are found as in Table 4.

Table 4. *GA simulation parameters*.

Simulation parameters	
Population width	50
Node number	20
Iteration number	20000
Mutation ratio	0.4

Computer network with 20x20 nodes is modelled by matrix below.

Here, nodes are represented by row and columns and the weights are labelled as matrix coefficients. In Table 5, simulation results and corresponding total cost is given in Table 6

$$A=\begin{bmatrix}
0 & 2 & 3 & 1 & 4 & 5 & 7 & 8 & 10 & 12 & 6 & 9 & 11 & 17 & 4 & 8 & 3 & 1 & 13 & 5 \\
2 & 0 & 1 & 4 & 5 & 7 & 8 & 10 & 12 & 6 & 9 & 11 & 17 & 4 & 8 & 3 & 1 & 13 & 5 & 2 \\
3 & 1 & 0 & 5 & 7 & 8 & 10 & 12 & 6 & 9 & 11 & 17 & 4 & 8 & 3 & 1 & 13 & 5 & 2 & 18 \\
1 & 4 & 5 & 0 & 8 & 10 & 12 & 6 & 9 & 11 & 17 & 4 & 8 & 3 & 1 & 13 & 5 & 2 & 18 & 7 \\
4 & 5 & 7 & 8 & 0 & 12 & 6 & 9 & 11 & 17 & 4 & 8 & 3 & 1 & 13 & 5 & 2 & 18 & 7 & 19 \\
5 & 7 & 8 & 10 & 12 & 0 & 9 & 11 & 17 & 4 & 8 & 3 & 1 & 13 & 5 & 2 & 18 & 7 & 19 & 9 \\
7 & 8 & 10 & 12 & 6 & 9 & 0 & 17 & 4 & 8 & 3 & 1 & 13 & 5 & 8 & 18 & 7 & 19 & 9 & 20 \\
8 & 10 & 12 & 6 & 9 & 11 & 17 & 0 & 8 & 3 & 1 & 13 & 5 & 2 & 18 & 7 & 19 & 9 & 20 & 14 \\
10 & 12 & 6 & 9 & 11 & 17 & 4 & 8 & 0 & 1 & 13 & 5 & 2 & 18 & 7 & 19 & 9 & 20 & 14 & 15 \\
12 & 6 & 9 & 11 & 17 & 4 & 8 & 3 & 1 & 0 & 5 & 2 & 18 & 7 & 19 & 9 & 20 & 14 & 15 & 8 \\
6 & 9 & 11 & 17 & 4 & 8 & 3 & 1 & 13 & 5 & 0 & 18 & 7 & 19 & 9 & 20 & 14 & 15 & 8 & 1 \\
9 & 11 & 17 & 4 & 8 & 3 & 1 & 13 & 5 & 2 & 18 & 0 & 19 & 9 & 20 & 14 & 15 & 8 & 1 & 10 \\
11 & 17 & 4 & 8 & 3 & 1 & 13 & 5 & 2 & 18 & 7 & 19 & 0 & 20 & 14 & 15 & 8 & 1 & 10 & 3 \\
17 & 4 & 8 & 3 & 1 & 13 & 5 & 2 & 18 & 7 & 19 & 9 & 20 & 0 & 15 & 8 & 1 & 10 & 3 & 6 \\
4 & 8 & 3 & 1 & 13 & 5 & 2 & 18 & 7 & 19 & 9 & 20 & 14 & 15 & 0 & 1 & 10 & 3 & 6 & 11 \\
8 & 3 & 1 & 13 & 5 & 2 & 18 & 7 & 19 & 9 & 20 & 14 & 15 & 8 & 1 & 0 & 3 & 6 & 11 & 4 \\
3 & 1 & 13 & 5 & 2 & 18 & 7 & 19 & 9 & 20 & 14 & 15 & 8 & 1 & 10 & 3 & 0 & 11 & 4 & 16 \\
1 & 13 & 5 & 2 & 18 & 7 & 19 & 9 & 20 & 14 & 15 & 8 & 1 & 10 & 3 & 6 & 11 & 0 & 16 & 21 \\
13 & 5 & 2 & 18 & 7 & 19 & 9 & 20 & 14 & 15 & 8 & 1 & 10 & 3 & 6 & 11 & 4 & 16 & 0 & 3 \\
5 & 2 & 18 & 7 & 19 & 9 & 20 & 14 & 15 & 8 & 1 & 10 & 3 & 6 & 11 & 4 & 16 & 21 & 3 & 0
\end{bmatrix} \quad (2)$$

Table 5. *GA computer simulation results.*

Iteration Number	Total Cost
1	117
12	103
42	99
49	97
50	95
96	88
111	81
120	80
192	79
216	70
1716	65
2479	61
4364	57
6941	54
10845	51
14406	50

Table 6. *The optimum way in 14406th iteration*

Optimum way																			
20	11	10	8	9	13	6	16	5	17	2	3	19	12	4	1	18	15	7	14

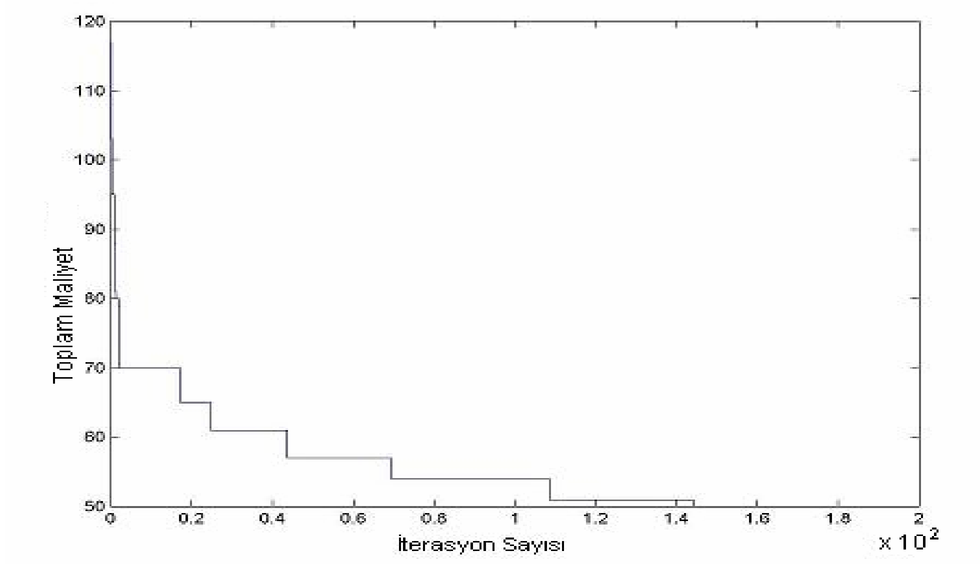

Fig. 2. *Total cost graphic of 20x20 nodes according to iteration number*

4. RESULT

In this paper, Genetic algorithm is applied for optimization of any number of nodes and best fit path is found from any pair of computers. The results are especially important for first establishment of a network with path costs are only known. We have studied on 10x10 and 20x20 computer networks, but it can be generalised for any dimension. It is shown that GA is a strong tool in computer network optimization.

REFERENCES

[1] Uçan, O.N., Osman, O.,"*Bilgisayar Ağları ve Haberleşme Teknikleri*", Istanbul University Press, 2003.

[2] Wattenhofer, R., *Ad-hoc Networks Mobile Computing. Distributed Computing Group*, 19. 2002

[3] Kurose, J.F., Ross, K.W., , *Computer Networking a Top-down Approach Featuring The Internet,* Addision Wesley, United States of America, 752, 2003.

[4] Gen M., Cheng R., Wang, D., "Genetic Algorithms for Solving Shortest Path Problems", *Proceedings of IEEE International Conference on Evolutionary Computing*, 401-406, 1997.

[5] Buckles, B.P., Petry, F.E., *Genetic Algorithms,* IEEE Computer Press, Los Alamitos,1994.

[6] Dejong, K., *Learning With Genetic Algorithms: An Overview, Machine Learning,* Vol.3, 121-138, Kluwer Academic, Hingham, MA, 1998.