



Cairo University  
**Egyptian Informatics Journal**

[www.elsevier.com/locate/eij](http://www.elsevier.com/locate/eij)  
[www.sciencedirect.com](http://www.sciencedirect.com)



ORIGINAL ARTICLE

# A genetic algorithm for finding the $k$ shortest paths in a network

Ahmed Younes Hamed

*Department of Computer Science, Faculty of Science, Sohag University, Egypt*

Received 27 April 2010; accepted 22 September 2010

Available online 28 October 2010

## KEYWORDS

Computer networks;  
Shortest paths;  
Genetic algorithms;  
Multimedia

**Abstract** Most of the multimedia applications require the  $k$  shortest paths during the communication between a single source and multiple destinations. This problem is known as multimedia multicast routing and has been proved to be NP-complete. The paper proposes a genetic algorithm to determine the  $k$  shortest paths with bandwidth constraints from a single source node to multiple destinations nodes. The algorithm uses the connection matrix of a given network, and the bandwidth of the links to obtain the  $k$  shortest paths. Some examples are provided to illustrate the effectiveness of this algorithm over conventional algorithms.

© 2010 Faculty of Computers and Information, Cairo University.  
Production and hosting by Elsevier B.V. All rights reserved.

## 1. Introduction

The  $k$  shortest paths problem has several applications in others network optimization problems. One of them is the restricted shortest path, where the shortest path that verifies a specified condition is searched. This research will attempt to apply a genetic algorithm to solve this problem based on a real world

system. This is based on the analogy of finding the shortest path (i.e., the shortest possible bandwidth) between two nodes in the communication networks (assuming that each edge in the network has the bandwidth value). So, applying a genetic algorithm is an interesting idea. This is clearly different from traditional algorithms that try to compare every possibility to find the best solution, which might be a time consuming algorithm for a network containing a large number of nodes and edges.

Many papers study algorithms for  $k$  shortest paths [1–20]. Yen [20] cite several additional papers on the subject going back as far as 1957. One must distinguish several common variations of the problem. In many of the papers cited above, the paths are restricted to be simple, i.e., no vertex can be repeated. Several papers [5,18] consider the version of the  $k$  shortest paths problem in which repeated vertices are allowed, and it is this version that we also study.

This paper will attempt to apply a genetic algorithm to solve the  $k$  shortest paths problem based on the links bandwidth of the network. The paper is organized in the following

E-mail address: [a\\_y\\_hamed@yahoo.com](mailto:a_y_hamed@yahoo.com)

1110-8665 © 2010 Faculty of Computers and Information, Cairo University. Production and hosting by Elsevier B.V. All rights reserved.

Peer review under responsibility of Faculty of Computers and Information, Cairo University.

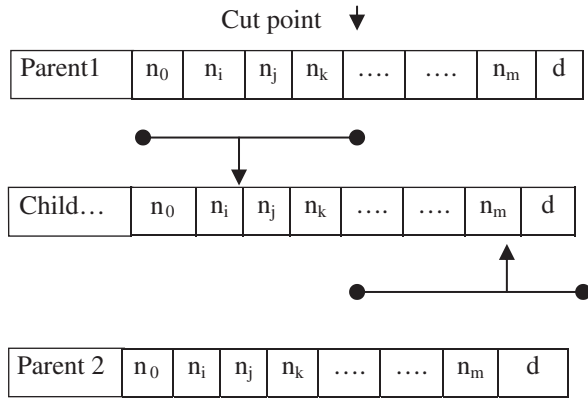
doi:[10.1016/j.eij.2010.10.004](https://doi.org/10.1016/j.eij.2010.10.004)



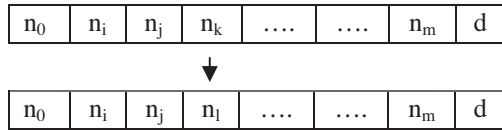
Production and hosting by Elsevier



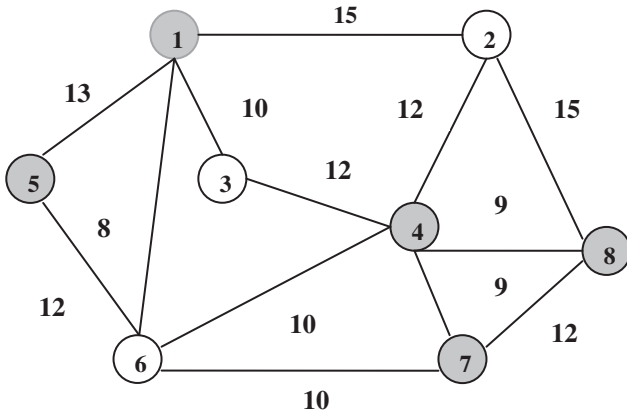
**Figure 1** A chromosome form (where  $n_i, n_j, n_k, \dots, n_m$  are the nodes between the source node  $n_0$  and destination node  $d$ ).



**Figure 2** Crossover operation.



**Figure 3** Mutation operation.

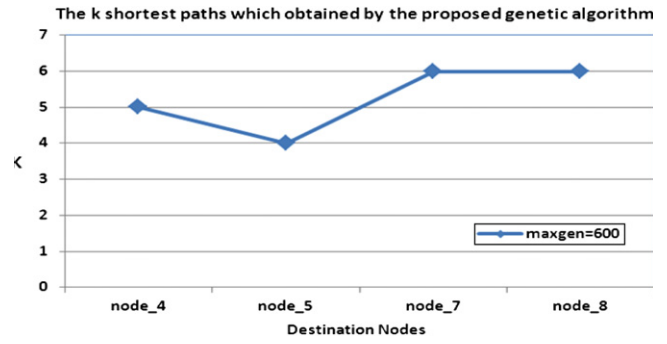


**Figure 4** The sample network.

sections: Section 2 presents the problem description and how it can be solved. The genetic algorithm and its operators are presented in Section 3. The proposed algorithm is presented in Section 4. Section 5 presents the experimental results and displays the obtained results.

## 2. Problem description

A network is usually represented as a weighted digraph,  $G = (N, E)$ , where  $N = \{1, \dots, n\}$  denotes the set of nodes and  $E = \{e_1, \dots, e_m\}$ , denotes the set of communication links connecting the nodes. Let  $M = \{n_0, u_1, u_2, \dots, u_m\} \subseteq N$  be a set of form source to destination nodes, where  $n_0$  is source



**Figure 5** The results given in Table 1.

**Table 1** The  $k$  shortest paths which obtained by the proposed genetic algorithm.

Destination node	The shortest paths	$k$	$Band(P)$
4	1 2 8 7 6 4	5	10
	1 3 4		10
	1 5 6 4		10
	1 2 4		13
	1 5 6 7 8 2 4		10
5	1 5	4	13
	1 2 4 6 5		10
	1 2 8 7 6 5		10
	1 3 4 6 5		10
7	1 3 4 2 8 7	6	10
	1 3 4 6 7		10
	1 2 4 6 7		10
	1 2 8 7		12
	1 5 6 7		10
	1 5 6 4 2 8 7		10
8	1 5 6 4 2 8	6	10
	1 3 4 2 8		10
	1 3 4 6 7 8		10
	1 5 6 7 8		10
	1 2 8		15
	1 2 4 6 7 8		10

node and  $U = \{u_1, u_2, \dots, u_m\}$  denotes a set of destination nodes.  $P(n_0, u_i)$  is a path from source node  $n_0$  to destination node  $u_i \in U$ .

The path  $P$  is the shortest path if the bandwidth of that path is equal to constant value  $B$  (this value is determined from the user or is a required value of the bandwidth). The bandwidth of  $P$  ( $Band(P)$ ) is the minimum value of link bandwidth ( $Band(e)$ ) in  $P$ . i.e.,

$$Band(P) = \min(Band(e), \quad e \in E_P) \quad (1)$$

Hence, the problem of bandwidth constrained  $k$  shortest path is to find all the paths from source node to each destination node which satisfy:

$$Band(P) \geq B.$$

## 3. The proposed genetic algorithm

Genetic algorithms, as powerful and broadly applicable stochastic search and optimization techniques, are the most

0	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	1	1	1	0
0	0	1	1	1	0	0	0	0	0	1	1	0	0	1	1	1	1	1	1
0	1	0	0	1	1	0	0	0	0	1	0	1	1	0	0	1	1	0	0
0	1	0	0	0	1	1	1	1	0	0	0	1	0	0	1	0	1	0	0
0	1	1	0	0	0	1	0	1	0	0	0	0	0	1	1	0	1	1	0
0	0	1	1	0	0	1	1	0	0	0	1	0	1	1	1	1	0	0	1
0	0	0	1	1	1	0	1	1	1	0	1	1	1	1	0	0	1	0	1
0	0	0	1	0	1	1	0	0	1	0	1	0	1	1	1	1	0	0	0
0	0	0	1	1	0	1	0	0	1	1	0	1	1	0	0	1	0	1	0
1	0	0	0	0	0	1	1	1	0	0	1	0	1	1	1	0	1	1	1
0	1	1	0	0	0	0	0	0	1	0	0	0	1	1	1	0	0	0	0
1	1	0	0	0	1	1	1	0	1	0	0	1	1	1	0	1	0	0	0
1	0	1	1	0	0	1	0	1	0	0	1	0	1	0	1	0	0	0	1
0	0	1	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	1	1
0	1	0	0	1	1	0	1	0	1	1	1	0	0	0	1	0	0	1	0
0	1	0	1	1	1	0	1	0	1	1	0	1	0	1	0	1	0	1	0
1	1	1	0	0	1	1	0	1	0	0	1	0	0	1	0	0	0	0	0
1	1	0	1	1	0	1	0	0	1	0	0	0	0	0	0	0	0	1	1
1	1	0	0	1	0	0	0	1	0	0	0	0	1	1	1	0	1	0	0
0	1	0	0	0	1	1	0	0	1	0	0	1	1	0	0	0	1	0	0

Figure 6 The connection matrix of the network (20 nodes).

0	0	0	0	0	0	0	0	0	13	0	9	13	0	0	0	12	3	3	0
0	0	8	13	9	0	0	0	0	0	3	2	0	0	4	13	9	6	8	2
0	8	0	0	7	9	0	0	0	0	1	0	9	13	0	0	10	1	0	0
0	13	0	0	0	15	13	16	10	0	0	0	1	0	0	12	0	6	0	0
0	9	7	0	0	0	3	0	12	0	0	0	0	0	16	9	0	2	2	0
0	0	9	15	0	0	8	14	0	0	0	15	0	9	2	3	5	0	0	7
0	0	0	13	3	8	0	10	13	11	0	7	14	2	0	0	0	11	0	3
0	0	0	16	0	14	10	0	0	9	0	6	0	12	6	5	15	0	0	0
0	0	0	10	12	0	13	0	0	4	6	0	6	5	0	0	11	0	9	0
13	0	0	0	0	0	11	9	4	0	0	11	0	0	2	2	0	8	1	13
0	3	1	0	0	0	0	0	6	0	0	0	0	1	10	12	0	0	0	0
9	2	0	0	0	15	7	6	0	11	0	0	11	1	7	0	5	0	0	0
13	0	9	1	0	0	14	0	6	0	0	11	0	3	0	14	0	0	0	6
0	0	13	0	0	9	2	12	5	1	1	1	3	0	0	0	0	0	12	10
0	4	0	0	16	2	0	6	0	2	10	7	0	0	0	5	1	0	7	0
0	13	0	12	9	3	0	5	0	2	12	0	14	0	5	0	7	0	8	0
12	9	10	0	0	5	0	15	11	0	0	5	0	0	1	7	0	0	0	0
3	6	1	6	2	0	11	0	0	8	0	0	0	0	0	0	0	0	5	6
3	8	0	0	2	0	0	0	9	1	0	0	0	12	7	8	0	5	0	0
0	2	0	0	0	7	3	0	0	13	0	0	6	10	0	0	0	6	0	0

Figure 7 The bandwidth values of the given network.

Table 2  $k$  shortest paths at  $N$  generations.

$N$ generations	$k$ shortest paths for each destination node							
	9	11	12	14	16	17	19	20
1000	3	2	1	1	3	1	1	1
10,000	13	8	18	5	19	15	3	9
20,000	18	15	24	12	29	20	10	13
30,000	24	18	28	17	31	22	7	9
80,000	31	27	38	29	39	32	14	19
90,000	33	34	34	30	45	34	18	18
120,000	35	28	36	34	45	32	16	24
140,000	38	30	40	33	50	32	20	19
180,000	39	34	41	36	53	35	20	21
200,000	40	38	45	37	54	35	23	26

widely known types of evolutionary computation methods today. In general, a genetic algorithm has five basic components as follows: (1) an encoding method that is a genetic representation (genotype) of solutions to the program. (2) A way to create an initial population of chromosomes. (3) The objective function. (4) The genetic operators (crossover and mutation) that alter the genetic composition of offspring during reproduction.

### 3.1. Encoding method

Given a network  $G(N, E)$  with  $N$  nodes and  $E$  is the set of communication links connecting the nodes. Also, we consider the source node  $n_0$  and destination nodes set  $U = \{u_1, u_2, \dots, u_m\}$ .

The chromosome can be represented by a string of integers with length  $N$ . The genes of the chromosome are the nodes between the source node  $n_0$  and destination node  $u_i$ . Each

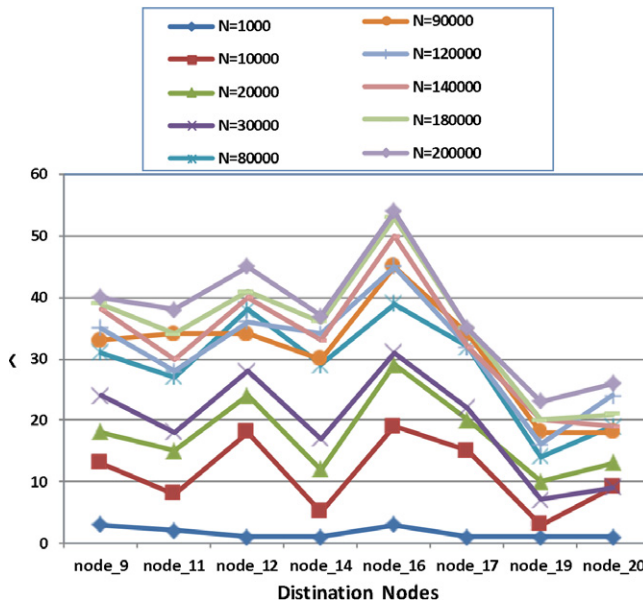


Figure 8  $k$  shortest paths at  $N$  generations.

chromosome in population denotes the shortest path. Obviously, a chromosome represents a candidate solution for the  $k$  shortest path problem since it guarantees the shortest path between the source node and any of the destination nodes.

### 3.2. Initial population

The initial population is generated according to the following steps:

1. A chromosome  $x$  in the initial population can be generated in a form as indicated in Fig. 1.
2. If the generated chromosome in Step 1 fails to meet 2-connectivity conditions, discard it and go to Step 1.
3. Repeat Steps 1 to 2 to generate  $\text{pop\_size}$  number of chromosomes.

### 3.3. The objective function

The objective function is to find the shortest paths from the source node to the destination nodes which satisfy

$$\text{Band}(P) = \min(\text{Band}(e), e \in E_P) \geq B$$

### 3.4. Crossover operation

The crossover operation is performed by one-cut point. In the proposed approach, the crossover operation will perform if the

crossover ratio ( $P_c$ ) is verified. The value of  $P_c$  is 0.9. The cut point is selected randomly. The offspring generated by crossover operation is shown in Fig. 2.

### 3.5. Mutation operation

The mutation operation is performed on bit-by-bit basis. In the proposed approach, the mutation operation will perform if the mutation ratio ( $P_m$ ) is verified. The mutation ratio,  $P_m$  in this approach is 0.2. The mutated bit is selected randomly. The offspring generated by mutation is shown in Fig. 3.

## 4. The proposed algorithm

This section presents the proposed GA for solving the  $k$  shortest paths problem. The steps of this algorithm are as follows:

**Algorithm:** Genetic algorithm for finding the  $k$  shortest paths

**Input:**  $\text{pop\_size}$ ,  $\text{maxgen}$ ,  $P_m$ ,  $P_c$ ,  $n_0$ , the destination nodes  $U$ ,  $B$ .

**Output:**

1. Generate the initial population as in Section 3.2.
2.  $\text{gen} \leftarrow 1$ .
3. **While** ( $\text{gen} \leq \text{maxgen}$ ) **do**
4.  $P \leftarrow 1$
5. **While** ( $p \leq \text{pop\_size}$ ) **do**
6. Obtain chromosomes of the new population, select two chromosomes from the parent population according to  $P_c$ . Apply crossover, and then mutate the new child according to  $P_m$  parameter.
7. Compute the bandwidth of the new child ( $\text{Band}(P)$ ) according to Eq. (1).
8. If  $\text{Band}(P) \geq B$  then Save this child as a candidate solution.
9.  $P \leftarrow p + 1$ .
10. **End if**
11. **End**
12. Print all obtained solutions.
13. **End**

## 5. Experimental results

In this section, we show the effectiveness of the above algorithm by applying it on two examples as follows.

### 5.1. First example

We consider a network with eight nodes as shows in Fig. 4. Each link has a corresponding bandwidth.

The parameters setting in this algorithm are:  $\text{pop\_size} = 20$ ,  $P_m = 0.2$ ,  $P_c = 0.9$ ,  $\text{maxgen} = 600$ . The source node  $n_0$  is the node No. 1 and the destination nodes are  $U = \{4, 5, 7, 8\}$ , and the objective value of  $B$  is equal to 10 (Fig. 5).

Table 3 Effect of mutation on  $k$  shortest paths.

Mutation rate $P_m$	$k$ shortest paths for each destination node							
	9	11	12	14	16	17	19	20
0.9	22	19	30	18	29	23	4	9
0.7	14	15	23	15	23	17	4	9
0.5	9	8	16	5	15	10	1	6
0.3	7	8	9	10	10	11	1	1
0.1	4	3	6	1	4	7	1	4

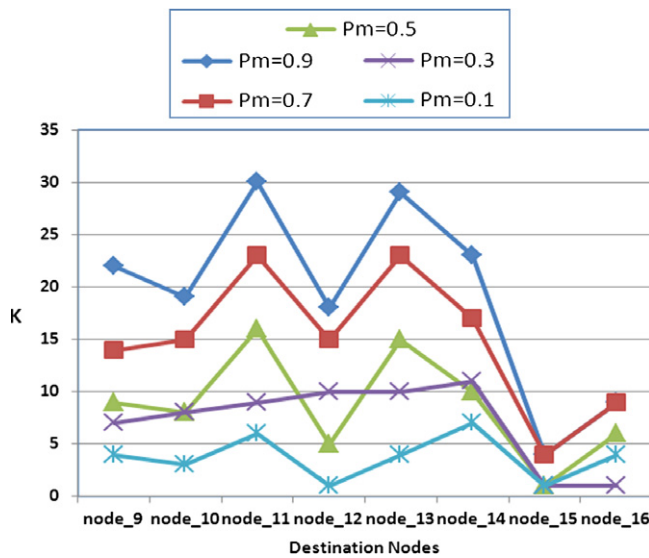


Figure 9 Study the effect of the mutation probability.

The  $k$  shortest paths which obtained by the proposed genetic algorithm are shown in Table 1. These results indicate that the proposed algorithm is finding the  $k$  shortest paths with bandwidth constraints from a single source node to multiple destinations nodes for any given network topology.

The following figure represents the results given in Table 1.

### 5.2. Second example

We consider another example with 20 nodes. The connection matrix of that example is shown in Fig. 6. The corresponding bandwidth of each link is shown in Fig. 7.

The parameters setting in this algorithm as:  $\text{pop\_size} = 25$ ,  $P_m \geq 0.1$ ,  $P_c = 0.9$ ,  $\text{maxgen} = 2000,000$ . The source node  $n_0$  is the node No. 1 and the destination nodes are  $U = \{9, 11, 12, 14, 16, 17, 19, 20\}$ , and the objective value of  $B$  is equal to 10.

The  $k$  shortest paths for each destination node at  $N$  generations are shown in the Table 2.

Fig. 8 represents the results given in Table 2.

Table 3 and Fig. 9 show the effect of varying the mutation probability.

It is clearly from the above table, the  $k$  shortest paths decrease when the mutation rate decrease in the proposed algorithm

## 6. Conclusions

The paper proposes a genetic algorithm to determine the  $k$  shortest paths with bandwidth constraints from a single source node to multiple destinations nodes. The algorithm uses the connection matrix of a given network, and the bandwidth of

the links to obtain the  $k$  shortest paths. The proposed GA has been applied on two examples network topology and the produced results are obtained by a less number of generations.

The proposed algorithm is considered to be the first algorithm that uses the genetic algorithms to obtain the  $k$  shortest paths from a single source node to multiple destinations nodes.

## References

- [1] Bako A, Kas P. Determining the  $k$ -th shortest path by matrix method. *Szigma* 1977;10:61–6 [In Hungarian].
- [2] Brander AW, Sinclair MC. A comparative study of  $k$ -shortest path algorithms. In: Proc. 11th UK performance engineering workshop for computer and telecommunications Systems; September 1995.
- [3] Carraresi P, Sodini C. A binary enumeration tree to find  $K$  shortest paths. In: Proc. 7th Symp. operations research. p. 177–88. Athinaum/Hain/Hanstein, Methods Oper. Res. 1983;45.
- [4] Consiglio A, Pecorella A. Using simulated annealing to solve the  $K$ -shortest path problem. In: Proc. conf. Italian assoc. operations research; September 1995.
- [5] Fox BL.  $K$ -th shortest paths and applications to the probabilistic networks. In: ORSA/TIMS joint national meeting, vol.23; 1975. p. B263.
- [6] Horne GJ. Finding the  $K$  least cost paths in an acyclic activity network. *J Oper Res Soc* 1980;31:443–8.
- [7] Katoh N, Ibaraki T, Mine H. An  $O(Kn^2)$  algorithm for  $K$  shortest simple paths in an undirected graph with nonnegative arc length. *Trans Inst Electron Commun Eng Jpn* 1978;E61:971–2.
- [8] Katoh N, Ibaraki T, Mine H. An efficient algorithm for  $K$  shortest simple paths. *Networks* 1982;12(4):411–27.
- [9] Kumar N, Ghosh RK. Parallel algorithm for finding first  $K$  Shortest paths. *Comput Sci Inform* 1994;24(3):21–8.
- [10] Law AG, Rezazadeh A. Computing the  $K$ -shortest paths, under nonnegative weighting. In: Proc. 22nd Manitoba Conf. Numerical Mathematics and Computing, Congr. Numer., vol. 92; 1993. p. 277–80.
- [11] Lawler EL. Comment on computing the  $k$  shortest paths in a graph. *Commun Assoc Comput Mach* 1977;20:603–4.
- [12] Minioka E. The  $K$ -th shortest path problem. In: ORSA/TIMS joint national meeting, vol. 23; 1975. p. B/116.
- [13] Perko A. Implementation of algorithms for  $K$  shortest loopless paths. *Networks* 1986;16:149–60.
- [14] Ruppert E. Finding the  $k$  shortest paths in parallel. In: Proc. 14th symp. theoretical aspects of computer science; February 1997.
- [15] Shibuya T. Finding the  $k$  shortest paths by AI search techniques. *Cooperative Res Rep Model Algorithms* 1995;7(77):212–22 [Inst. of Static Mathematics].
- [16] Shier DR. Algorithms for finding the  $k$  shortest paths in a network. In: ORSA/TIMS joint national meeting; 1976. p. 115.
- [17] Shier DR. Iterative methods for determining the  $k$  shortest paths in a network. *Networks* 1976;6(3):205–29.
- [18] Shier DR. On algorithms for finding the  $k$  shortest paths in a network. *Networks* 1979;9(3):195–214.
- [19] Weigand MM. A new algorithm for the solution of the  $k$ -th best route problem. *Computing* 1976;16:139–51.
- [20] Yen JY. Another algorithm for finding the  $K$  shortest-loop less network paths. In: Proc. 41st mtg. operations research society of America, vol. 20; 1972. p. B/185.