

Genetic Algorithm Based Routing Algorithm for Packet Switching Network

Olalekan Ogunbiyi ^{#1}, Wasiu Akande Ahmed ^{#2}

^{#1} Electrical and Computer Engineering, Kwara State University, Malete, Ilorin, Nigeria

^{#2} Space Research and Development, African Regional Centre for Space Science and Technology Education-English, OAU Campus, Ile Ife, Nigeria

Abstract— The distributed nature of routers in a packet switching network (PSN) and the need for them to send packet to one another following an optimal path make routing algorithm a necessity. In this paper, genetic algorithm (GA) is proposed to solve routing problem in a PSN. Computer simulation showed that GA approach obtain a result similar to Dijkstra's algorithm (used in routing information protocol (RIP)) but with some improvement. The GA technique finds the optimal path between any source and destination nodes in a network. The network may be simple or complex, such that a feasible path or optimal path seems intractable. The developed technique was able to determine an entire route right from the source node, quick adaptation to changes in the network. It can also handle a numerous number of different constraints in the network. The binary coding used will also make it easier to implement in hardware.

Keywords— Routing, packet switching network, metric, genetic algorithm and Dijkstra's algorithm.

I. INTRODUCTION

A communication network consists of interconnected devices that interact and communicate together with the objective of transmitting data from source to destination. Data transfer from source to destination is routed through the switching devices. Packet switching is a switching technique in which the data to be transmitted is divided into chunks called packet and each packet is switched from node to node until all the packets get to the destination. The switching device is usually a router. A router needs to be able to look at a packet destination address and determines the output port to best transmit the packet to the destination. The information about the best path is readily available in the forwarding table but how the router obtains the information in the forwarding table is an optimization problem called routing [1], [2].

Routing is the act of moving information across an internetwork from a source to a destination. One or more intermediate node (router) can be encountered along the way. In the case of virtual circuits, routing is an issue for the connection request packet alone, since all subsequent packets follow the same route as the request packet but in datagram, a packet must follow the optimal path as at the time it is being sent.

A router in an internet is the devices responsible for forwarding and receiving packets through the interconnected networks. It has two major groups of interfaces: WAN Interfaces which connect to the external network like ISP or another router and LAN Interfaces which connect to LAN network [3]. A router directs packet to its destination by examining a packet's destination address and determine the

best path with the aid of routing information located in the routing table.

Routing information is the topological condition of the network and the metric at the time the information is needed [4]. The routing table contains routing metric information. A metric is a numerical value used by routing protocols to determine the best path to a destination. The total metric of a particular route is equal to the sum of the metric of the network that comprises the route [2]. Some of the metrics used by routing protocols are the hop count (which is the number of routers a packet must travel through to get to its destination), the bandwidth (which is the speed of a link, also known as the data capacity of a link), network delay and reliability of a link [1], [2], [5].

Routing is achieved in most practical networks by running routing protocols among the nodes. All the widely used routing protocols use distributed routing algorithms, because there is a possibility that two routers will at one instant have different ideas about the shortest path to a destination. Complications or inefficiency can result if routers choose paths to optimize their own objectives, which may conflict with the objectives of other participants. In fact, each one may think that the other one is closer to the destination, and decide to send packets to the other one [3]. The routing objective is to select the least cost path, where the cost signifies the performance criteria. The distributed nature of routing algorithms is one of the main reasons why it has been such a rich field of research and development. There are a lot of challenges in making distributed algorithms work well.

An algorithm that has been used to solve the shortest path problem includes Dijkstra's algorithm, Bellman-Ford algorithm and Floy-Warshall algorithm [6]. These algorithms are deterministic in their search for an optimal path. They search for a particular solution in the feasible region.

Meta-heuristic algorithms are another approach to routing. They are stochastic algorithms that randomly search for a good solution in a search space. They may not be able to obtain the best solution but they usually arrive at a good solution. Popular Meta-heuristics for combinatorial problems include Genetic algorithms, Ant Colony Optimization (ACO) and Simulated Annealing (SA) [7].

This paper uses GA to obtain an optimal path in a PSN. It is considered as a good optimization technique because of its ability to intelligently search a very large space of solution. GA start with a set of solutions called population; it then applies genetic operators such as mutation and crossover to evolve another population of solutions. Solutions from one

population are taken and used to form a new population. This is motivated by a hope that the new population of solution will be better than the old one. Solutions which are selected to form new solutions (offspring) are selected according to their fitness; the fitter they are the more chances they have to reproduce. GA iterates until it converges to a good solution or a stopping criteria is met [8].

GA is a good optimization technique that has been applied to a wide range of problem. In [9], research was carried out on solving travelling salesperson problem (TSP) by using GA. They proposed a new representation method of chromosomes using binary matrix to find a near optimal solution. TSP is similar to routing in a PSN but the two problems are different. All the nodes must contribute to the salesperson path in TSP but it not a must for all the nodes to contribute to a routing path in PSN. Nevertheless, their ideas and concept were so useful in this research.

Vehicle routing problem has also been solved using GA, this was carried out by [10], [11]. This problem involves optimizing a fleet of vehicles that are to serve a number of customers from a central depot. There is a limited capacity that the vehicle can carry and each customer has a certain demand. Optimizing such a highly constraint transportation problem using GA was a good contribution to the ideas used in this study.

Research has earlier been carried out on network routing using GA [12]. The algorithm has to find the shortest path between the source and destination nodes. They also used a pattern search mechanism (permutation encoding), combined with binary encoding. They worked on six nodes where the nodes are numbered from one to six. Each number is coded in binary and the length of a chromosome will be the combination of all the bits (i.e. $6 * 4 = 24$). This makes their model more difficult to code because the GA will have to incorporate another algorithm with the fitness function.

In [11], the authors design their own metric table by imposing heavy cost on links having both the source and destination as the same node. Since the objective is to minimize the total cost of a path, any chromosome that considered such a link as part of the route will never feature at the last generation. This idea was a good suggestion to this study. Like some others, they used a real value encoding, permuted to form different chromosomes.

In this paper, a different approach is used to obtain an optimal path in a packet network. This approach uses a fixed size and bit string population. It incorporates a linear programming model of the network. Also, the algorithm will have nothing to do with the packet; it only optimizes the path for the packet to be routed. These make the search faster and the programming easier.

II. SYSTEM MODELING

A. Graphical Modeling of a PSN

To understand the complex issue of routing, it is better to think of PSN as a graph. A graph consists of a finite set of nodes (routers) and edges (communication links) that connect

the graph together. A router operates like a switch, it receives an incoming data and sends it to a destination, and therefore the real router network is a directed graph, as illustrated in Fig.1. The links are usually assigned a cost (C_{ij}), where i is the source and j is the destination. The costs are assigned to the links to support one or more design objectives. The cost may be inversely related to the bandwidth, proportional to the current queue delay or the cost of using the communication link between two nodes. It can also be the combination of the three [2].

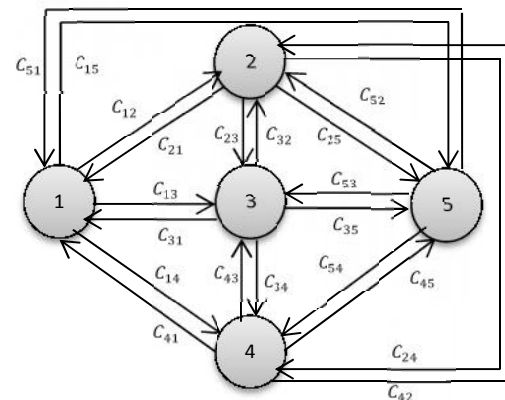


Fig. 1 Graphical modeling of a PSN

B. Dijkstra's Algorithm Approach

The Dijkstra's algorithm calculates the shortest path between two points on a network. The algorithm divides the nodes into two sets: temporary and permanent. It starts by selecting a node, makes it a temporary node and examines it if it possesses some criteria such that it can be changed to a permanent node.

If U_i is the shortest distance from source node 1 to node i , and C_{ij} is defined as the cost of arc (i, j) , Dijkstra's algorithm define the label for an immediate succeeding node j as:

$$[U_j, i] = [U_i + C_{ij}, i] \quad (1)$$

Where $[C_{ij} \geq 0]$ and $[U_j, i]$ means the shortest distance from node i to node j .

The label for the starting node is $[0, -]$, this is indicating that the starting node has no predecessor. The algorithm employs two types of node label: temporary label and permanent label. A temporary label is modified if a shorter route to a node can be found. If a better route could not be found, the status of the temporary label is changed to a permanent label [6].

C. Genetic Algorithm Approach

1) *Population Type*: This work uses a bit string (Binary Coding) population type. Bit string is chosen because of the model used. Hence a gene of the chromosome will correspond to a link in the network. The length of a chromosome is the square of the size of the network.

2) *Fitness Function Evaluation*: The GA based PSN routing evaluates the fitness value using the objective function. The objective function is derived from the linear programming model of the graph network for routing. The goal of the model is to minimize the path cost from source node to destination node. Using the network of Fig. 1, the linear programming model is as follows:

The objective function:

$$\text{Minimize } F(X) = \sum_{i=1}^N \cdot \sum_{j=1}^N C_{ij} X_{ij} \quad (2)$$

$F(X)$ = Cost function that is to be minimized

C_{ij} = Link cost from node i to node j

X_{ij} = Link between node i and node j

where N = number of nodes in the network,

Subject to the Constraints:

1. $\sum_{j=1}^N X_{ij} = 1$
 $\sum_{j=1}^N X_{ji} \approx 0$ for i = source node
2. $\sum_{j=1}^N X_{ij} = 0$
 $\sum_{j=1}^N X_{ji} = 1$ for i = destination node
3. $\sum_{j=1}^N X_{ij} - \sum_{j=1}^N X_{ji} = 0$
for $i \neq$ source or destination node
4. $X_{ij} = 0$ or 1 for $i \neq j$
5. $X_{ij} = 0$ for $i = j$
6. If $X_{ij} = 1$ and $j \neq$ destination,
then $\sum_{k=1}^N X_{jk} = 1$

Consider a network of five routers, such that all the routers are connected to each other.

Then, number of nodes = 5 and number of links will be 25. i.e. $X_{11}, X_{12}, X_{13}, \dots, X_{55}$.

The link cost will form a cost matrix C , where C_{ij} tends to infinity (1000) for all $i = j$.

1) *Rank Fitness Scaling*: This work uses rank scaling function to convert the value returned by the objective function into a value that can be passed to the selection function. The objective of the model is to minimize the cost $F(x)$, therefore the higher the cost is the poorer the fitness of the chromosome. Rank scales the raw scores based on the rank of each individual, rather than its score. The rank of an individual is its position in the sorted scores. The rank of the fittest individual is 1, the next fittest is 2, and so on [8]. Ranking fitness scaling removes the effect of the spread of the raw scores. Thus Ranking function is inversely proportional to the objective function.

2) *Stopping criteria*: Check whether the optimization stopping criteria is met. If yes, produce output else move to the next step. Assuming the stopping criterion has not been met, the move to the next step.

3) *Roulette Selection*: The GA based routing algorithm uses Roulette selection function to select parents for the next generation. It selects the fittest chromosome and eliminates the lowest fitness chromosome. Since the fitness function has been ranked, the idea is to select the better individuals of the population to be the parent that will produce offspring in the next generation. The chromosomes of the next generation are selected from the current population by the spinning roulette wheel method.

4) *Scattered crossover*: Crossover enables the algorithm to extract the best genes from different individuals and recombine them into potentially superior children. It combines sub parts of two parent chromosomes and produces offspring that contains some part of both the parent genetic material. Scattered crossover was used in our GA based routing, it creates a random binary vector, and then selects the genes where the vector is a 1 from parent 1, and the genes where the vector is a 0 from parent 2, and combines the genes to form the child [8].

5) *Adaptive Feasible Mutation*: Mutation children are created by applying random changes to a single individual in the current generation to create a child. This process is essential to GA; it prevents the algorithm from converging to a local optimal path. Mutation adds to the diversity of a population and thereby increases the likelihood that the algorithm will generate individuals with better fitness values. Adaptive feasible mutation was used in this research. It randomly generates directions that are adaptive with respect to the last successful or unsuccessful generation [7], [10].

6) *Replacement*: Re-insertion of the new offspring (crossover and mutation offspring) back into the population is done following these steps: If population size = 10, Elite count = 4 and Crossover fraction = 0.7; then, number of crossover children = $0.7 * 6 = 4.2 \approx 4$. The remaining 2 individuals are mutation children. Therefore, four elite, four crossover children and two mutation children produced from mutation makes up the total population of the second generation.

7) *Stopping criteria*: The process continues until the iteration converges or the stopping criteria are met. Condition for the termination of GA is determined by the stopping criteria. The GA based algorithm in this work terminates after a given generation (maximum number of iterations) or after a stall time limit (if there is no improvement in the best fitness value for an interval of time).

III. SIMULATIONS AND RESULTS

Computer simulation was carried out for both the Dijkstra's algorithm and the developed GA based routing algorithm for PSN. Codes were written in Matlab® programming environment to implement the two algorithms. Matlab® was chosen because of its extensive in-built function for graph, optimization and GA. It also has GA toolbox that extends the capabilities of the optimization toolbox. The capability of the in-built functions is extended by writing codes in m-files; the codes are to handle the objective function, the constraints and the GA operators: selection crossover, and mutation.

Simulations were carried out to obtain the effect of GA parameters on the performance of the algorithm. Different network scenarios were simulated using the program: when the network has three (3) nodes, four (4) nodes, five (5) nodes, six (6) nodes and seven (7) nodes. Comparison between the two routing algorithms were made and inferences drawn. Simulation was also carried out for situations when some links are not present, when the constraint does not permit the combination of some links to form a route.

A. Effect of GA Parameters on Path Determination

Table I to IV shows the result of simulations carried out to study the effect of GA parameters towards the determination of an optimal path. The parameters studied include: initial population, population size and maximum generation.

B. Effect of Population Initialization and Population Size

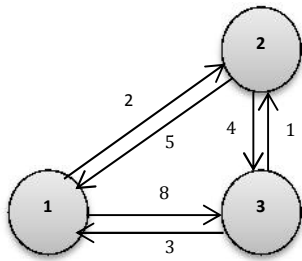


Fig. 2 PSN with three fully linked nodes

Table I Optimal Path in a Network with 3 Fully Connected Nodes and no Initialized Chromosomes

S/N	Population size	Maximum generation	Cost	Path
1	10	50	1000	-
2	10	50	1000	-
3	20	50	8	1-3
4	20	50	1000	-
5	50	50	6	1-2-3
6	50	50	8	1-3

Table II Optimal Path in a Network with 3 Fully Connected Nodes and one (1) Initialized Chromosomes

S/N	Population size	Maximum generation	Cost	Path
1	10	50	8	1-3
2	10	50	1000	-
3	20	50	8	1-3
4	20	50	8	1-3
5	50	50	8	1-3
6	50	50	8	1-3

Table III Optimal Path in a Network with 3 Three Fully Connected Nodes and Four (4) Initialized Chromosomes

S/N	Population size	Maximum generation	Cost	Path
1	50	50	8	1-3
2	50	50	6	1-2-3
3	50	50	6	1-2-3
4	50	50	6	1-2-3
5	50	50	6	1-2-3
6	50	50	6	1-2-3

C. Effect Of Maximum Generation

Table IV Optimal Path in a Network With 3 Three Fully Connected Nodes, Four (4) Initialized Chromosomes and Different Maximum Generation

S/N	Population size	Maximum generation	Cost	Path
1	50	5	1000	-
2	50	5	1000	-
3	50	10	6	1-2-3
4	50	10	1000	-
5	50	20	8	1-3
6	50	20	6	1-2-3
7	50	50	6	1-2-3
8	50	50	6	1-2-3

D. Simulations for Different Network Scenario

Fig. 3 shows a PSN with five fully linked nodes while Figure 4 shows a PSN with five partly linked nodes. The link cost is a generic value. When the source is equal to destination, the cost is assigned 1000 (which is far greater than other assigned cost) to show that such a path does not exist. This must be specified in the program used to compute the fitness function.

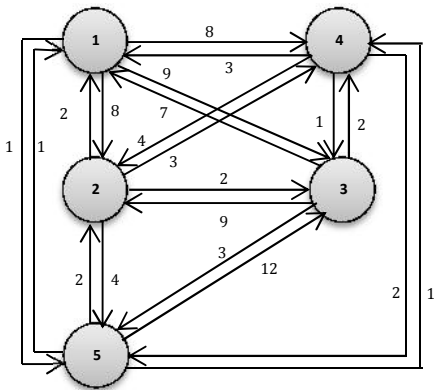


Fig. 3 PSN with five fully linked nodes

Table V Optimal Path in a Network with Five Fully Connected Nodes And Two (2) Initialized Chromosomes for GA Based Routing Algorithm

Source – Destination	Dijkstra's Algorithm		GA Based Routing Algorithm			
	Cost	Path	Simulation1	Simulation2	Cost	Path
1-2	3	1-5-2	14	1-4-3-5-2	3	1-5-2
1-3	3	1-5-4-3	9	1-3	10	1-2-3
1-4	2	1-5-4	2	1-5-4	2	1-4-5
1-5	1	1-5	1	1-5	1	1-5
2-1	2	2-1	1000	-	2	2-1
2-3	2	2-3	16	2-5-3	2	2-3
2-4	3	2-4	4	2-3-4	13	2-5-1-4
2-5	3	2-1-5	3	2-1-5	3	2-1-5
3-1	4	3-5-1	4	3-5-1	7	3-1
3-2	5	3-5-2	6	3-4-2	5	3-5-2
3-4	2	3-4	9	3-1-5-4	4	3-5-4
3-5	3	3-5	4	3-5-4	3	3-5
4-1	3	4-1	3	4-1	3	4-5-1
4-2	4	4-2	6	4-3-5-2	4	4-2
4-3	1	4-3	6	4-5-2	1	4-3
4-5	2	4-5	9	4-3-1-5	2	4-5
5-1	1	5-1	1	5-1	1	5-1
5-2	2	5-2	2	5-2	9	5-1-2
5-3	2	5-4-3	2	5-4-3	7	5-4-2-3
5-4	1	5-4	14	5-3-4	14	5-3-4

Table VI Optimal Path in a Network With Five Fully Connected Nodes And Eighty (80) Initialized Chromosomes for GA Based Routing Algorithm

Source – Destination	Dijkstra's Algorithm		GA Based Routing Algorithm Simulation1	
	Cost	Path	Cost	Path
1-2	3	1-5-2	3	1-5-2
1-3	3	1-5-4-3	3	1-5-4-3
1-4	2	1-5-4	2	1-5-4
1-5	1	1-5	1	1-5
2-1	2	2-1	2	2-1
2-3	2	2-3	2	2-3
2-4	3	2-4	3	2-4
2-5	3	2-1-5	3	2-1-5
3-1	4	3-5-1	4	3-5-1
3-2	5	3-5-2	5	3-5-2
3-4	2	3-4	2	3-4
3-5	3	3-5	3	3-5
4-1	3	4-1	3	4-5-1
4-2	4	4-2	4	4-5-2
4-3	1	4-3	1	4-3
4-5	2	4-5	2	4-5
5-1	1	5-1	1	5-1
5-2	2	5-2	2	5-2
5-3	2	5-4-3	2	5-4-3
5-4	1	5-4	1	5-3

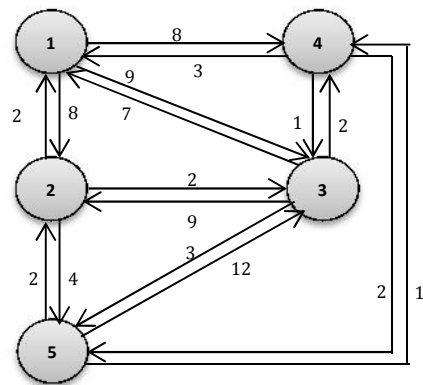


Fig. 4 PSN with five partly linked nodes

Table VII Optimal Path in a Network With Five Partly Connected Nodes and Eighty (80) Initialized Chromosomes for GA Based Routing Algorithm

Source – Destination	Dijkstra's Algorithm		GA Based Routing Algorithm Simulation1	
	Cost	Path	Cost	Path
1-2	8	1-2	8	1-2
1-3	9	1-3	9	1-4-3
1-4	8	1-4	8	1-4
1-5	10	1-4-5	10	1-4-5
2-1	2	2-1	2	2-1
2-3	2	2-3	2	2-3
2-4	4	2-3-4	4	2-3-4
2-5	4	2-5	4	2-5
3-1	5	3-4-1	5	3-4-1
3-2	5	3-5-2	5	3-5-2
3-4	2	3-4	2	3-4
3-5	3	3-5	3	3-5
4-1	3	4-1	3	4-1
4-2	4	4-5-2	4	4-5-2
4-3	1	4-3	1	4-3
4-5	2	4-5	2	4-5
5-1	4	5-4-1	4	5-2-1
5-2	2	5-2	2	5-2
5-3	2	5-4-3	2	5-4-3
5-4	1	5-4	1	5-4

IV DISCUSSION OF RESULTS

Tables I to VII contain the results obtained from the different simulations that were made. This section gives the observations and inferences that can be drawn from the tables, in relation to the determination of an optimal path in a PSN. Table V to VII shows the result of simulations of Dijkstra's algorithm and GA based routing algorithm. The reason for this is to check the performance of the GA based routing algorithm with an already established routing algorithm.

A. Factors Affecting the Determination of an Optimal Path

From Table I, it can be observed that GA based routing algorithm may not perform well without population initialization and enough population size. With the method adopted in this work, if there are 'n' number of nodes, the length of a chromosome represented by k will be n^2 ($k = n^2$) and the number of possible solutions will be 2^k . Hence the algorithm could not arrive at any feasible path with zero number of chromosomes initialized and small sizes of the population. Nevertheless, increasing the size of the population produced a better result as shown in Table II and III.

Another GA parameter that can affect the performance of GA based routing algorithm is the maximum number of generation. This is one of the stopping criteria used in this work. Table IV shows that the algorithm will not produce a good result until after some iteration. The number of iterations depends on the problem at hand. 20 generations is ok for a

network with 3 nodes but was not ok for network with five nodes and above.

B. Other Observations and Inferences

Table VI to VII shows the potentials of GA based routing algorithm as being able to obtain an optimal or near optimal path in a fully linked and partly linked network. The results in simulation1 and simulation 2 shows that the developed algorithm will perform better in a PSN, since convergence to least cost path is not a must. This is an advantage in traffic engineering because Dijkstra's algorithm will continue to search the solution space till the shortest path is obtained.

Another observation from Table V to VII is that GA based routing algorithm does not converge to the same value all the time. That is why the simulation was done twice. Sometimes, it converges to the least cost path and at times a near value. In any case, once the parameters are well set, GA based routing algorithm will always converge to a good solution.

In Table V, when the source is 4 and destination is 1, the solution can be either 4-5-1 or 4-1. The GA based routing algorithm was able to choose any of these paths; this shows the flexibility of the algorithm. There are several cases of this in the result of the simulation.

Three types of simulations were made when the network has 5 nodes. The results in Table VI are better than that of Table V, this show that the number of initialized chromosome should increase as the number of nodes increases. Table VII contains an acceptable result, showing that GA based routing algorithm is a promising algorithm in a fully connected or partly connected network.

According to Cisco [13], the performance of a routing algorithm can only be studied in terms of the design goal. The design goal includes: optimality, simplicity, cost overhead, robustness and stability, rapid convergence and flexibility. Simulation results show that GA based routing algorithm will achieved these to a greater extent.

V. CONCLUSION

The potential of GA in solving the shortest path routing problem in a PSN by minimizing the cost of a routing path has been studied and simulated in the Matlab® environment. The most critical part of the research work was the formulation of the linear programming model with its associated constraints for routing, and the application of Matlab® resources for the simulation. Simulation was also carried out to determine how best to set the GA parameters for routing. The results are very encouraging, since our developed GA for routing can obtain an optimal solution at any instance, reduces congestion, handle a complex network, it is simple and can eliminate many of the manual configuration tax (once they are coded as a constraint), it is therefore satisfied to be a potential alternative for routing in a Packet Switching Network.

REFERENCES

- [1] S. Willian "Data and Computer Communication", 8th Edition, Pearson Prentice Hill, Pearson Education Inc.: Upper Saddle River, 2003.
- [2] S. Y. Anddrew, "Computer Networks", Prentice – Hall, Inc.: New Jersey, 2004.

- [3] M. W. Curt, "Data Communication and Computer Network", Thomas place: Boston Massachusetts, 2007.
- [4] L. Peterson and D. L. Bruces, "Computer Network", 3rd Edition, Morgan Kaufmann Publishers: USA, 2003.
- [5] A. F. Behrouz, Data Communication and Networking, McGraw-Hill Companies: NY, 2003.
- [6] A. T. Hamdy, "Operation Research: An Introduction", 8th Edition, Pearson Pentice Hill: Delhi India, 2007.
- [7] L. Rhyd, "An Introduction to Metaheuristic Algorithms and the Problems they (try to) Solve", *Talk at the British Computing Society Lecture, Cardiff School of Mathematics / Cardiff Business School*, 2008.
- [8] The Math Works, "Genetic Algorithm and Direct Search Toolbox", The Math Works, Inc.: 3 Apple Hill Drive Natick, 2004
- [9] H. K. Foza, K. Nasiruddin, I. Syed & T. N. Shaikh, "Solving Travelling Salesman Problem(TSP) by Using Genetic Algorithm", *International Journal of Basic and Applied Science IJBAS*, Vol: 9 No:10, Page 79-88, 2009.
- [10] S. B. Aslaug, "Solving the Vehicle Routing Problem with Genetic Algorithm", *Information and Mathematical Modeling*, IMM technical University of Denmark, pp. 1-127, 2004.
- [11] K. Rakesh and K. Mahesh, "Exploring Genetic Algorithm for Shortest Path Optimization in Data Networks", *Global Journal of Computer Science and Technology (GJCST)*, Vol. 10, Issue 11 (Ver. 1.0), pp. 8-12, 2010.
- [12] N. Gihan and G. A. Wahied, "Network Routing Protocol using Genetic Algorithms", *International Journal of Electrical & Computer Sciences IJECS-IJENS* Vol:10 No:02, 2010.
- [13] K. Amit, "Computational Intelligence: Principles, Techniques and Applications", Springer-Verlag: Berlin Heidelberg, New York, 2005.
- [14] C. Kelly and C. Kelly, "Guide to cisco networking", 3rd edition, Thomas learning: Massachusetts, 2004.
- [15] M. R Masillamani, V. S. Avinankumar, P. Rajesh, and G.V. Uma, "Genetic Algorithm for Distance Vector Routing Technique", *AIML 06 International Conference*, 13 - 15 June 2006, pp. 171-174.
- [16] M. Mitchell, "An Introduction to Genetic Algorithms", Fifth Printing, MIT Press Cambridge: Massachusetts, 1999.
- [17] G. L. David, "Linear and Nonlinear Programming", Third Edition, Springer Science + Business Media: New York, 2008.
- [18] K. Erwin, Advance Engineering Mathematics, 9th Edition, John Wiley & Son: NJ, 2006.
- [19] A. M. Saad, M. Mohammad and N. A. Amer, "Hybrid Dynamic Routing Protocol For Finding an Optimal Path to Routers", *International Journal Of Academic Research*, Vol. 3, No. 2, Part IV, 2011.
- [20] N. Selvanathan and J. T. Wee, "A Genetic Algorithm Solution to Solve the Shortest Path Problem in OSPF and MPLS", *Malaysian Journal of Computer Science*, Vol. 16, No. 1, pp. 58-67, 2003.
- [21] L. Suihong, A. N. Zincir, and I. H. Malcolm, "Intelligent Packets for Dynamic Network Routing Using Distributed Genetic Algorithm", *Proceeding of the Genetic and Evolutionary Computation Conference*, Morgan Kaufmann Publishers Inc.: San Francisco, CA, 2002.
- [22] M. Zbigniew, "Genetic Algorithm + Data Structures + Evolution Programs", Springer-Verlag Berlin Heidelberg: New York, 1996.
- [23] W. A. Chang and R. S. Ramakrishna, "A Genetic Algorithm for Shortest Path Routing Problem and the Sizing of Populations", *IEEE Transactions on Evolutionary Computation*, Vol. 6, No. 6, pp. 566-579, 2002.