International Journal of Mathematical Archive-7(8), 2016, 76-84
Available online through www.ijma.info ISSN 2229 – 5046

SHORTEST PATH IDENTIFICATION USING THE MINIMUM HOP COUNT AND MAXIMUM BANDWIDTH CONSTRAINED BASED ALGORITHM IN NOC

M. MUTHULAKSHMI*

Research Scholar, Karpagam University, Coimbatore, Tamil Nadu, India.

M. M. SHANMUGAPRIYA

**Assistant professor, Department of Mathematics,
Karpagam University, Coimbatore-641021, Tamil Nadu, India.**

(Received On: 04-08-16; Revised & Accepted On: 30-08-16)

ABSTRACT

The Shortest path problems are one of the network flow optimization problems with large number of applications in various fields. Conventional techniques cannot be used for solving the shortest paths in a large network as they are too slow and require large number of storage. Finding the multiple shortest path gives the optimal solution in case of any fault occurred in a single shortest path in the run-time dimension. The proposed algorithm uses the fuzzy, shortest path algorithm and widest shortest path algorithm to identify the multiple shortest paths. Increased reliability, high throughput and high packet delivery ratio are achieved using the proposed algorithm.

Keywords: *Shortest path, minimum hop count, bandwidth, and throughput.*

1. INTRODUCTION

The shortest path problem is one of the most studied problems in the fields of network optimization. The main idea behind the shortest path problem is to find the path between the two nodes and optimize the path based on their time, distance and weight. The Shortest path is divided into two ways. They are single shortest path and multiple shortest paths. Multiple shortest paths are more efficient than a single shortest path. The multiple path routing has the benefits of fault tolerance, increased bandwidth; improved reliability, load balancing and quality of service [1]. Lots of approaches are developed to deal with the fuzzy shortest path problem [2-8].

The shortest path problems are handled with fuzzy numbers and are called as fuzzy shortest path algorithm. The shortest path in a network: discrete fuzzy shortest length method [9], Fuzzy shortest path problem as a bi-level programming model [10], uncertain environment [11], and fuzzy arc lengths[12], The Fuzzy Physarum algorithm is one of the biological inspired algorithm for the shortest path [13].

In the paper, framework is formulated for the multiple shortest paths using a combination of Fuzzy shortest path and widest shortest path algorithm. The framework consists of two phases; the first phase finds the multiple paths with less traffic using the fuzzy shortest path algorithm; the second phase finds the minimum hop count, increased bandwidth in the respective single path in the multiple paths using the widest shortest path algorithm. The benefits of the proposed framework easily reach the destination without loss of any information using the increased bandwidth, less traffic and less loss rate.

The rest of the paper is organized as follows. In Section 2, literature survey on the shortest path algorithm. Section 3 describes the proposed fuzzy shortest path problem and the widest shortest path algorithm. Demonstration of the proposed framework is given in the section 4. The concluding remarks are given in the Section 5.

Corresponding Author: M. Muthulakshmi*
Research Scholar, Karpagam University, Coimbatore, Tamil Nadu, India.

2. LITERATURE SURVEY

The solutions for the unconstrained shortest path are proposed in [14]. The solution involves splitting of algorithm into two sets. The first sets rely on the optimality principle and the second set decides the set with k-shortest paths. Dijkstras and bellman-ford-moore algorithms are used in the selecting the shortest path. The first set algorithm follows an optimality principle that tracks the shortest path which is defined by the shortest sub-paths. For the second set of algorithms, it forms a tree using the identified k-shortest paths.

The main idea of [15] is to translate the problem for one with two points, source and destination to a problem with one point. Path is found from the source to destination by finding a path from source to any node and then finding the shortest path from that node to destination. It is done by using the priority queue of paths. The breadth is used for the first search that contains zero edged paths that points itself and updates the list of the output paths repeatedly by removing the shortest paths. Then add the one edge extensions to the list of output paths of the priority queue. Binary heap for each vertex is constructed and the edges that do not follow the shortest path are listed and that are reached by following the shortest-path tree edges.

An equal cost multipath is proposed [16] for forwarding the packets in the multipath that possesses equal cost. The approach reduces load balancing of the multi-path routing by reducing itself to the shortest path. The reduction results in reducing the ability to decrease the congestion. The approach [17] uses ECMP, but it does not consider the cost of the multipath. Instead it floods the sampled traffic load information.

In [18] a multipath routing scheme the proposed is based on the open Shortest Path First Algorithm. The proposed algorithm finds the subset of the path that satisfies cyclic conditions. The router chooses the path to the next-hop considering the cost of the path between the next-hop the destination must be less than the cost of path from the router to destination. An efficient data structure is maintained in the router to calculate the alternate next-hops. All the packets are sent to the destination through the loop-free paths when the router routes the packet to the next-hop. Each router R maintains the shortest path d(R, X) from the router to the destination X. The advantage of maintaining the next-hop in the router helps in the speedy recovery in the case of link failure.

Yen [19] proposes an algorithm to find the acyclic path that have the shortest distance between the two nodes in a network. The advantage of this algorithm is its computational upper bound which increases linearly with the K value. The algorithm considers every node except the destination node in the shortest path and computes the alternate shortest paths from the each node to the destination node. The rule is alternate node should not choose the node on the root path for transferring packets and also it should not choose the node that exists in the already chosen shortest path. When a new path is found, it remains as the second best shortest path. In this manner all the alternative paths are calculated from the each node and stored until the required number of K paths are achieved.

Hershberger *et al.* proposed [20] an algorithm to find the k-shortest acyclic paths based on the replacement paths. For some directed paths, replacement paths subroutine may fail, however it may be detected easily. The replacement path problem is to find the shortest path between the source and destination, in the each graph G with the edge removed for i=1, 2,…p. Problem arises when nodes of the path p are removed.

K-shortest path of QoS routing scheme is proposed [21] [22] for connection oriented networks. The algorithm determines one of all the k-best loop less paths. Dijkstras algorithm is used in the construction of paths. Algorithm for both the hop-based and bandwidth based classes are proposed. Hop-based algorithm is implemented to the networks where frequent updation of link state information is not possible.

3. METHODOLOGY

The following two resource-oriented objectives for load balancing are defined.

Objective 1: The objective of the proposed routing method is to find out the multiple path selection using the Fuzzy logic operation. Multiple path selection is based on the less latency, loss rate and less traffic in their updated bandwidth.

Objective 2: Minimize the number of hops. The objective is needed because large number of hops in certain link disturbs the other path in the same link. A feasible path with minimal number of hop with maximum number of reservable bandwidth is selected using the widest shortest path algorithm. The objective 2 is used in the each single path in the obtained multiple paths.

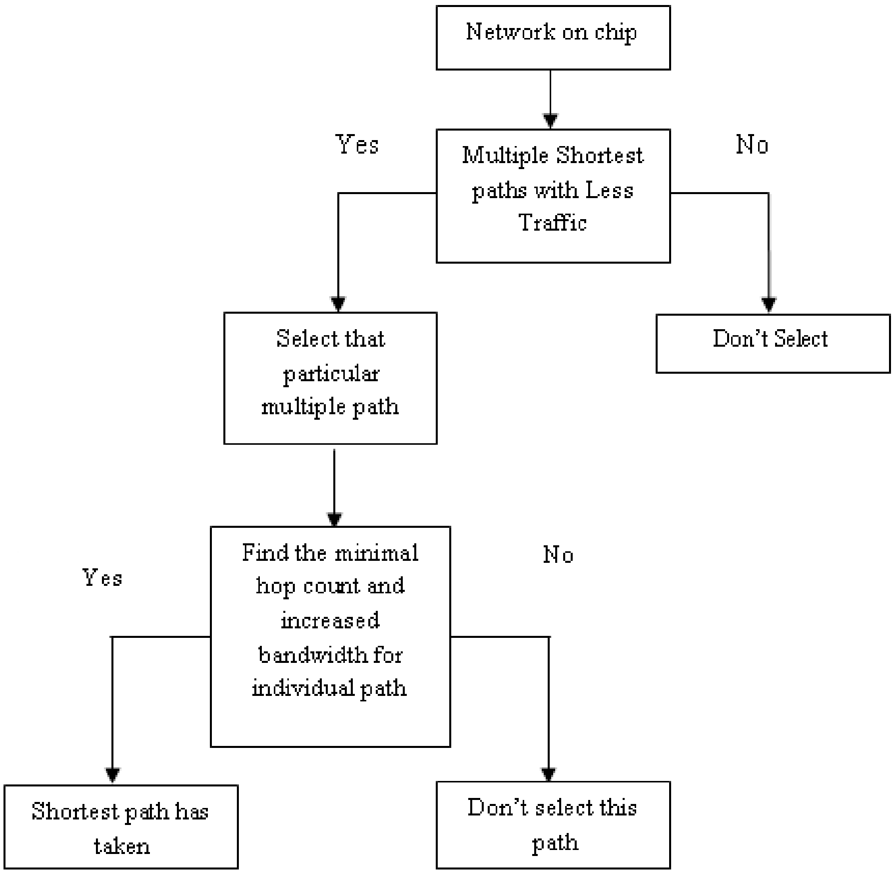

Figure-1: Flowchart for proposed shortest path.

3.1 Multiple-path selection algorithm:

Fuzzy logic operation is used to solve the multiple-path selection problem. First, link metrics of the respective network graph is updated after finding one path then move to the next path based on their graph. Multiple path selection is based on the less latency, less loss rate and less traffic. Taken this for consideration, correlation cost between links is introduced. Each link has a new cost with respect to their previous chosen path set S called as correlation cost

$$correlation \quad \cos t_S^L \sum_{S_{i\in S}} C_r\left(L, S_i\right) \tag{1}$$

$$C_r\left(A,B\right) = \frac{Cov\left(A,B\right)}{\sqrt{Var\left(A\right)}\sqrt{Var\left(B\right)}} \tag{2}$$

$$C_r\left(A,B\right) = \sum_{a\in A}\sum_{b\in B} Cov\left(a,b\right) \tag{3}$$

A and B are the two path in the defined network on the chip. The above equation (1) and (2) defines the correlated cost of the link $L = (i, j)$ with each chosen path S.

Then correlation cost is defined by below equation:

$$\cos t_S^L = \alpha . correlation \quad \cos t_S^L + \sum_{i=1}^{R} \alpha_i\left(i, j\right) \tag{4}$$

Where i and j defines the link between the correlation; R defines the traditional metrics; α and α_i are proper scaling factors to balance these metrics.

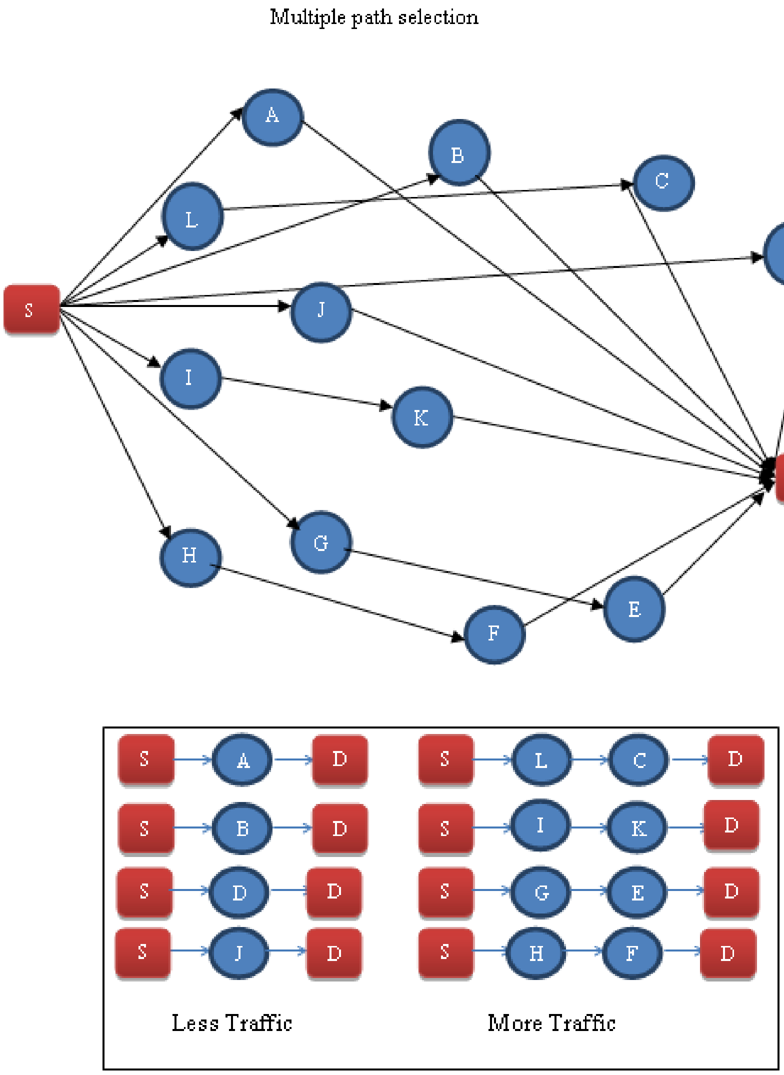

Figure-2: Example for Multiple shortest paths.

Multiple Shortest Path Algorithms

Step 1: Update the less bandwidth, less latency, less rate and less traffic of each link.

Step 2: Correlation cost for each link is calculated with respect to their previous chosen links.

Step 3: Calculate the cost function for each link using the combination of correlation cost and tradition metrics.

Step 4: Find the shortest path based on the new cost, less traffic rate using the fuzzy logic algorithm.

Step 5: Fuzzy logic algorithm computes all the possible path length based on the step 4 F_i from $i = 1, 2, \ldots n$ where $F_i = \left(a_1^{'}, b_1^{'}, c_1^{'}\right)$.

Step 6: Then initialize $F_{\min} = (a, b, c) = F_i = \left(a_1^{'}, b_1^{'}, c_1^{'}\right)$.

Step 7: Initialize i = 2.

Step 8: Compute

$$b = \begin{cases} b & \text{if} \quad b \le a_i^{'} \\ \dfrac{\left(b \times b_j^{'}\right) - \left(a \times a_i^{'}\right)}{\left(b \times b_j^{'}\right) - \left(a + a_j^{'}\right)} & \text{if} \quad b > a_i^{'} \end{cases}$$

$$a = \min\left(a, a_i^{'}\right)$$

Step 9: After computing the value of a and b. Set $F_{\min} = (a,b,c)$ as calculated in the previous steps.

Step 10: Then increment $i = i+1$ till I value is less than n+1

Step 11: *If* $i > n+1$ then it again computes a and b value till it becomes greater value.

Step 12: *if* i becomes greater than n, it stop the process and determines the routing path.

3.2 Widest shortest path algorithm:

Widest-shortest path is defined as a feasible path with the minimum hop count. If there is more than one path has the same minimum hop count value, the one with the maximum reservable bandwidth is selected.

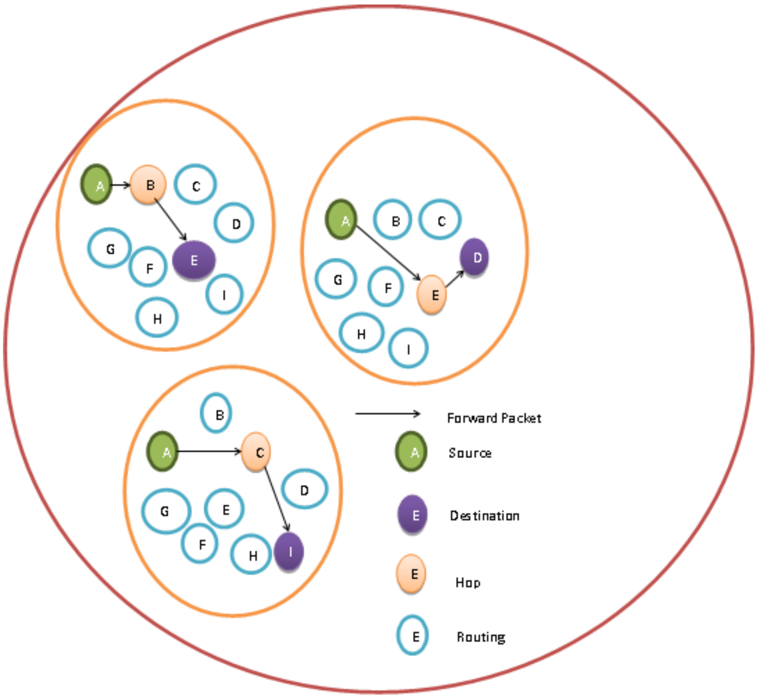

Figure 3: Individual path with their hop count.

The objective of the Widest-shortest path algorithms is to select the shortest path based on the bandwidth constraints; number of hops. In the first stage, the shortest path between the source and destination is in completed second stage, bandwidth is used to break the ties among the paths that have the same hop count, and the path that has the highest amount of available bandwidth is selected.

Widest Shortest Path algorithm

Widest_shortest_path$(G: Graph, s: Node, d: Node, d: real, j: real, b: Node \rightarrow int)$

Step 1: $r\_vec \leftarrow Sorting\ (L)$, sort it in decreasing order

Step 2: $E \leftarrow length \quad of \quad r\_vec$, it represents the number of different link residual bandwidth

Step 3: *for* $v \in V$ do

Step 4: $N_k \leftarrow \left| (b(v) - b) / L_{\max} \right|,$

Step 5: $\max\_hop = \max(\max\_hop, N_v)$

Step 6: $m \leftarrow \min(|v|, \max\_hop)$ where $|V|$ represents the number of nodes in G

Step 7: Initialize the single source (G, s, E, r_vec)

Step 8: *for* $h \leftarrow 1$ to $m-1$ do, it represents the iteration over hop count

Step 9: *for* $k \leftarrow 1$ to E do, represents the iteration over different values of link residual bandwidth

Step 10: *if* $(n_k \geq h)$ then, if h meets the hop count bound set by j

Step 11: *for* $(u, v) \in L$ do, for all links in G

Step 12: Relax (u, k, l_k, h, N_v)

Step 13: *if* $l_k(d) \leq d$ then

Step 14: return π_k , it represents that path which is found

Step 15: return FALSE

Initialize-single-source (G, s, E, r_vec)

Step 1: for $k \leftarrow 1$ to E , it represents iteration over different values of link residual bandwidth

Step 2: $n_k \leftarrow \left[r_{vec[k]} . j - b / L_{\max} \right],$ it represents bound on hop count determined by j

Step 3: for $v \in V$ do for all nodes in G

Step 4: $l_k[v] \leftarrow \infty,$ it represents for $r = r\_vec[k]$

Step 5: $\pi_k[v] \leftarrow NIL$

Step 6: return

Relax (u, k, l_k, h, N_v)

Step 1: *if* $N_v \leq h$ and $R_{(u,v)} \geq r_{vec[k]}$ then

Step 2: *if* $l_k[v] > l_k[u] + l_k(u, v)$ then

Step 3: $l_k[v] \leftarrow l_k[u] + l_k(u, v)$

Step 4: $\pi_k[v] \leftarrow u$

Step 5: return

4. EXPERIMENTAL RESULTS

The performance of the proposed shortest path methods is evaluated in accordance to the following metrics:

- Average Delay
- Packet loss
- Packet Delivery Ratio
- Throughput
- Overhead

Average delay

Average delay specifies how long a bit takes to reach another node or endpoint and it is measured by fraction of seconds. Formula for calculating Average delay is

$$Average\quad Delay = \frac{sum(each\quad data\quad packet\quad delay)}{sum(packet\quad received)}$$

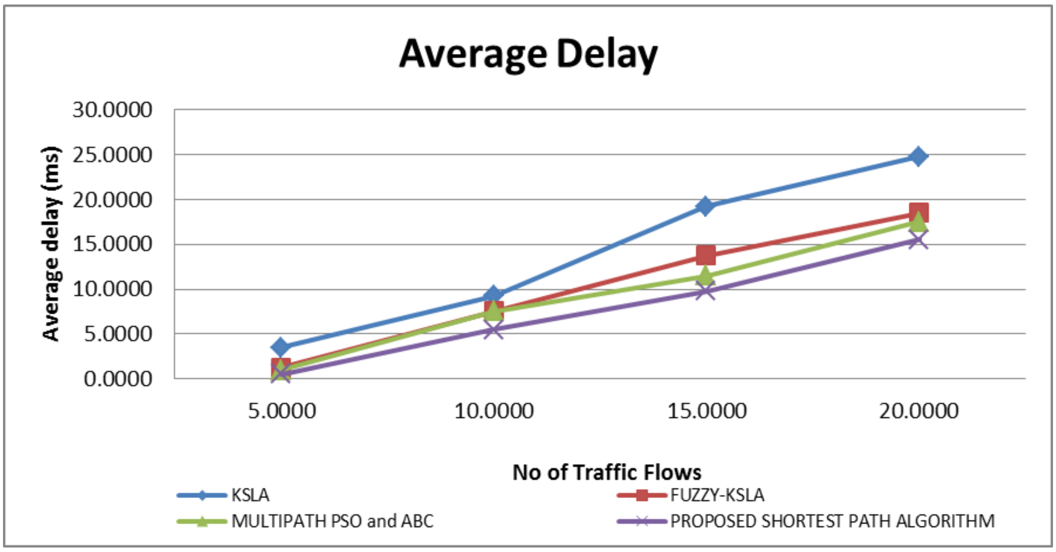

Figure-4: Average delay comparison.

The proposed method obtains less delay. The average delay report is showed in the figure 4.

Packet loss

It is the total number of packets that are transmitted by the sender and not received by the receiver. In other words, it is the difference between the total number of packets sent and total number of packets received.

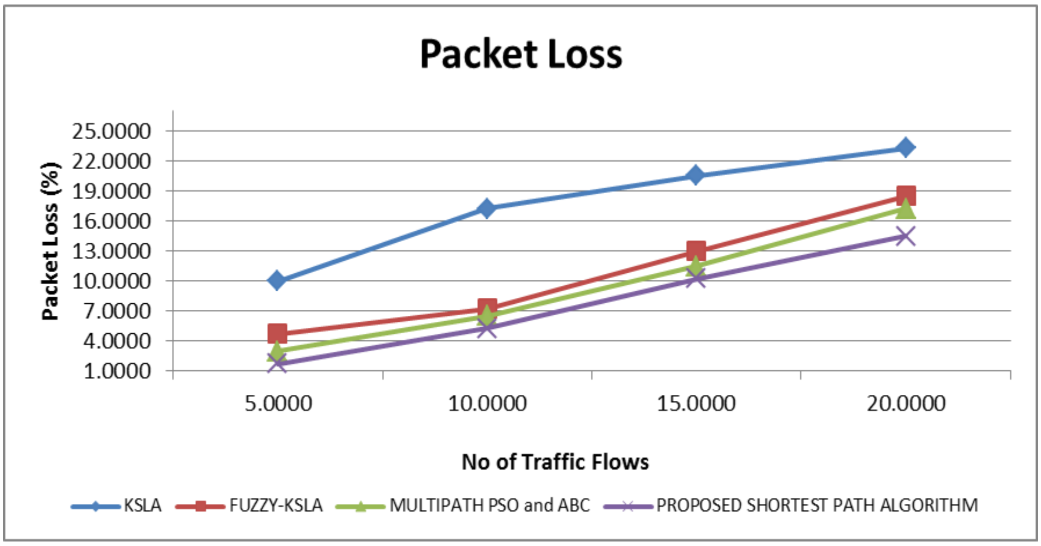

Figure-5: Packet Loss comparison.

The above figure shows the comparison of proposed algorithm's packet loss with other algorithms. The proposed algorithm has the minimum packet loss when compared with other three algorithms.

Overhead

Control overhead defines as the number of control packets propagates throughout the networks by each node. It is measured by the following formula.

$$ControlOverhead = \frac{Number \quad of \quad Control \quad Message \quad Sent}{Number \quad of \quad Data \quad Receivedt}$$

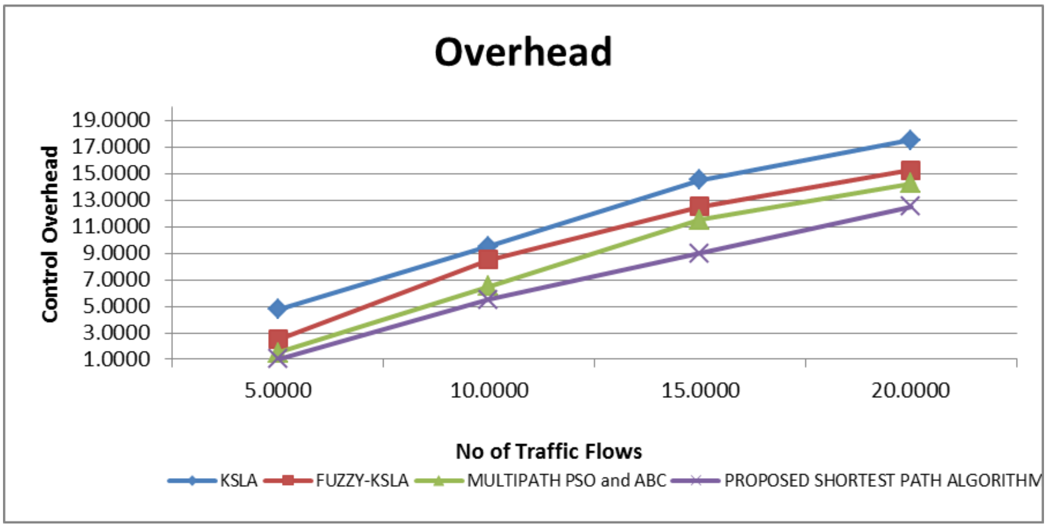

Figure 6: Overhead comparison.

The Figure 6 shows the comparison of the overhead of various algorithms. The proposed algorithm has the minimum overhead when compared with other algorithms.

Packet Delivery Ratio

PDR is the ratio of actual number of packets delivered in the destination. The PDR is measured using following formula.

$$PDR = \frac{total \quad number \quad of \quad packets \quad received}{total \quad number \quad of \quad packets \quad sent} \times 100$$

The comparison result is shown in the figure and proposed algorithm has the highest packet delivery ratio.

Throughput

Throughput is defined as the number of packets delivered successfully in a given amount of time.

$$Throughput = \frac{total \quad number \quad of \quad packets \quad successfull \quad delivered}{total \quad time \quad taken}$$

From the figure it is shown that the proposed method achieves high throughput value.

5. CONCLUSION

The shortest path from the source to destination in a network is needed. Combination of the Fuzzy shortest path and widest shortest path algorithm is proposed for the shortest path identification. Multiple shortest paths are more efficient than the single shortest path in case of the fault occurrences. Fuzzy shortest path algorithm finds multiple shortest paths and widest shortest path algorithms calculate minimum hop count and bandwidth for each path. Ranking is done with respect to the effectiveness of the path. The proposed algorithmic evaluated for the sample network and its reliability is proved in the experimental results. The proposed shortest path algorithm is compared with the KSLA, Fuzzy KSLA and Multipath PSO and ABC algorithms using the performance parameters such as average delay, packet loss, control overhead, packet delivery ratio and throughput.

REFERENCES

1. Shuchita Upadhyaya and Gaytri Devi, "Path Identification in Multipath Routing", *Journal of Global Research in Computer Science*, Volume 2, No. 9, September 2011.
2. D. Dubois, H. Prade, "Fuzzy Sets and Systems: Theory and Applications", *Academic Press, New York*, 1980.
3. E.Q.V. Martins, "On a multicriteria shortest path problem", Eur. J. Operat. Res. 16 (1984) 236–245.
4. R. Yager, "Paths of least resistance on possibilistic production systems", *Fuzzy Sets Syst*. 19 (1986) 121–132.
5. J. Broumbaugh-Smith, D. Shier, "An empirical investigation of some bicriterion shortest path algorithm", Eur. J. Operat. Res. 43 (1989) 216–224.
6. C.M. Klein, "Fuzzy shortest paths", Fuzzy Sets Syst. 39 (1991) 27–41.
7. K. Lin, M. Chen, "The fuzzy shortest path problem and its most vital arcs", Fuzzy Sets Syst. 58 (1994) 343–353.
8. S. Okada, T. Soper, "A shortest path problem on a network with fuzzy lengths", Fuzzy Sets Syst. 109 (2000) 129–140.
9. Tzung-Nan Chuang, Jung-Yuan Kung, "A new algorithm for the discrete fuzzy shortest path problem in a network", Applied Mathematics and Computation, Volume 174, Issue 1, 1 March 2006, Pages 660-668.
10. Esmaile Keshavarz, Esmaile Khorram, "A fuzzy shortest path with the highest reliability", Journal of Computational and Applied Mathematics 230 (2009) 204–212.
11. Yong Deng, Yuxin Chen, Yajuan Zhang, Sankaran Mahadevan, "Fuzzy Dijkstra algorithm for shortest path problem under uncertain environment", *Applied Soft Computing*, Volume 12, Issue 3, March 2012, Pages 1231-1237.
12. Reza Hassanzadeha, Iraj Mahdavia, Nezam Mahdavi-Amirib and Ali Tajdin, "A genetic algorithm for solving fuzzy shortest path problems with mixed fuzzy arc lengths", *Mathematical and Computer Modelling*, Volume 57, Issues 1–2, January 2013, Pages 84-99.
13. Yajuan Zhang, Zili Zhang, Yong Deng, Sankaran Mahadevan, "A biologically inspired solution for fuzzy shortest path problems", *Applied Soft Computing*, Volume 13, Issue 5, May 2013, Pages 2356-2363.
14. E.Q.V. Martins, M.M.B. Pascoal and J.L.E. Santos, "The K Shortest Paths Problem," Research Report, CISUC, June 1998.
15. D.Eppstein, "Finding the k Shortest Paths", March 31, 1997
16. J. Moy, "OSPF version 2," RFC2328, 1988.
17. C. Villamizar, "OSPF Optimized Multipath (OSPF OMP)", Internet Draft, 1999. vailable: http://www.ietf.org/ proceedings/99mar/I-D/draft-ietf-ospf-omp-02.txt
18. P. Narvaez, K. Y. Siu, "Efficient Algorithms for Multi-Path Link State Routing", ISCOM'99, Kaohsiung, Taiwan, 1999.
19. J.Y.Yen, "Finding the K-shortest loop less paths in a Network", Management Science, Vol. 17,No. 11, july, 1971
20. J.Hershberger, M.Maxel,S.Suri, "Finding the K shortest paths:A New algorithm and its implementation", ACM Transactions on Algorithms, Vol. 3, No. 4, Article 45, Publication date: November 2007.
21. R. Guerin and A. Orda, "QoS-based Routing in Networks with Inaccurate Information: Theory and Algorithms", IEEE/ACM Pansaction on Networking, Vol. 7, No. 3, June 1999, pp. 350-364
22. YanxiaJia, IoanisNikolaidis and P. Gburzynski, "Multiple Path QoS Routing", *Proceedings of ICC'01 Finland*, pages 2583-2587, June 2001.

Source of support: Nil, Conflict of interest: None Declared