ISSN: 0971-1260
Vol-22-Issue-17-September-2019

A Framework For Pre-Computated Multi-Constrained Quickest Qos Path Algorithm

Ashutosh Sharma
School of Electronics and Electrical Engineering
Lovely Professional University
Phagwara, Punjab, India
ashutosh.25385@lpu.co.in

Abstract:-In this paper, a novel approach has been given on the most reliable quickest tuned data path. (MRTDP)., which gives the mobility or freedom get scaling down the delay capacity constrained to the path to be lie in the feasible reason. Let $N = (V, E, c, d, r)$network, which consists of V no of vertex and E no of edges. The QoS parameters used are capacity $c(u, v) > 0$, delay $d(u, v) > 0$ and reliability i.e. Probability of link failure $r(u, v) \in [0, 1]$ of the edge (u, v). We have to find the QoS path that is feasible and lies in the performance envelope. In this letter we present the MRTDP with $O(E + VlogV)$ time algorithm.

I. INTRODUCTION

A computer network can be given by undirected graph $G = (V, E)$ with loop free path from source to destination with assumption that nodes are perfect but with the probability of failure of links (u, v). The QoS parameters used are capacity $c(u, v) > 0$, delay $d(u, v) > 0$ and reliability i.e. Probability of link failure $r(u, v) \in [0, 1]$ of the edge (u, v)[1].

In this paper, we present the various application requirements of time tolerance with the path which gives the feasible path with reliability and transmission time[2]. The amount of data σ has to be transmitted from source node (s) to destination node (t) with consideration of edge $(u, v) \in E$, the probability of that path and transmission delay is[1, 3]:

$$T(u,v) = [d(u,v) + \sigma / c(u,v)]$$

$$R(u,v) = -\ln r(u,v)$$

Let P is the path chosen from (s) to (t) with three tuple consideration $P(v1, v2 \ldots vk)$. The minimum transmission delay for the σ unit data from node (s) to (t) for path P is [1, 4]:

$$T(P,\sigma,\alpha) = \sum_{i=1}^{k-1} d(v_i, v_{i+1}) + \left[\frac{\sigma}{\min_{i=1}^{k-1} \{c(v_i, v_{k+1})\}} \right]$$

Where first term is delay of path (P), denominator in second term is capacity of path (P), σ is the amount of data to be transmitted and α is the tuning factor or we can say the balancing term.

ISSN: 0971-1260
Vol-22-Issue-17-September-2019

Definition 1. A path $(s - t)$ is known as optimized path if path fulfills the condition of feasibility and best performance factors consideration of delay has to be least and reliability has to be most for the given amount of data (σ). The name for the path can be given most reliable quickest tuned data path (MRTDP)[5-8].

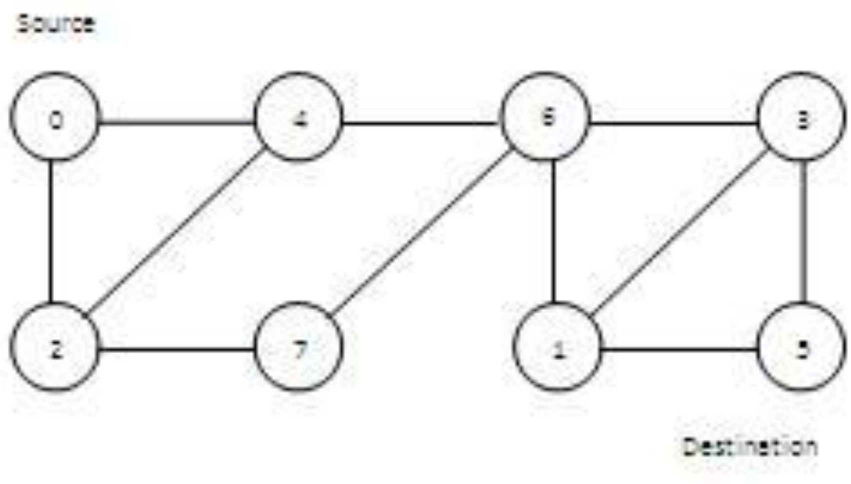

Fig 1 A network showing the 3-tuple (r, c, d)

To understand the concept of MRTDP, an illustration of a simple network is made in fig. 1 with the weight function of 3- tuple (r, c, d). To transmit 1 Mb data through path (s, u, t) from s to t the time required is and reliability is from equation of T withdifferent $\alpha$and the point should lies under area criteria theorem.

Definition 2. A point (x_1, y_1) is more optimal than (x_2, y_2) if $x_1 < x_2$ and $y_1 < y_2$such that x_1 and y_1 lies in the feasible region.

For optimized feasible path with high performance region we are interested with three tuple cost function to the Dijkastra algorithm[9, 10].

$$P(u,v) = -\ln r(u,v) e^{-\left[d(u,v) + \frac{\sigma}{c(u,v)}\right]}$$

$$P(u,v) = -\ln r(u,v) + [d(u,v) + \sigma / c(u,v)]$$

$$E(u,v) = K + \alpha L$$

where$r(u, v), d(u, v), c(u, v)$ and σ are reliability, delay, capacity of link between u and v and unit of data to be transmitted respectively. $\alpha$is the tuning factor depends on the application time requirement denoted by per second (sec^{-1}) or we can say that $\alpha = 1/T$ where T is the tuning time in seconds

Definition 3. Any request for the path is known as feasible path request if it comes in the feasible area ($A_{feasible}$) using pre-computation process to support the QoS.

Precomputation and performance envelope

Finding a QoS path is a NP-complete problem and also it increases extra overhead to solve the overhead problem and scalability purpose pre-computation process has been used. Also fault tolerance of network increased because if any link failure occurs it choose pre-computed path itself for restoration. A number of requests that are facilitate distribution of available resources for different requests in efficient manner leads to load balancing. As network has random traffic size and different time requirement of accomplish the job a pre-computation process in standard network goes undergoes two parts: first part comes when event has to be occurs i.e. compute all paths for different set of random values and find the high performance region and keep it in database and second part comes when event arrive[11, 12].

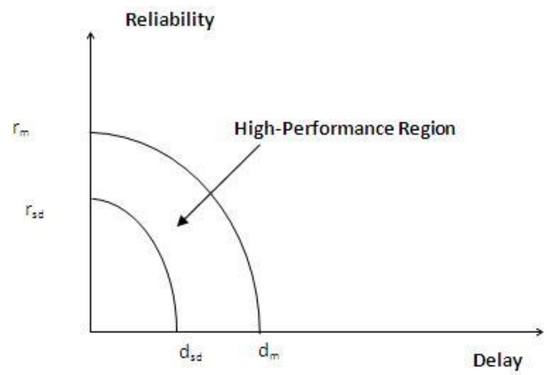

Fig. 2 Performance envelope for high performance

For approximation of the optimized high performance region we have used the statistical analysis to establish a relation between the mean- standard deviation problems and gave the bounded area between mean and standard deviation area. Precomputation part leads to gives these bounds by analysing the random set of values for the topology to analyse the ebhaviour of the network. Thus by analysisng the behaviour of network a tuning rule can be formed.

Number of algorithms has been proposed by authors on the QoS routing algorithms where path is computed in$O\ (rE\ +\ rVlogV)$, $r$is the different capacities[13]. The most reliable quickest path problem got very much attention due to integrity of number of application to the communication and wide area of research interest of researchers. In this paper, we show the most reliable tuned data path, which can be solved in $O\ (E\ +\ VlogV)$ time.

II. THE MOST RELIABLE TUNED DATA PATH

A new method of mixed metric has been shown here in this paper. The steps included in the algorithm shown below.

Algorithm

Algorithm helps to find the feasible path in the high performance region step by step and fig. 2 gives the overview of proposed algorithm of multi-constrained quickest path problem. There are four main basis steps which gives optimal QoS path which is feasible in High performance region are given below: I step compute shortest path using Dijkstra algorithm using given cost function and gives weight of the path. II step check path with weight if weight full fills the condition or go to step 0. The 2 & 3 step assign the path for traffic only and only if all these three steps become true unless they will go to step 0 again with next tuning factor.

The algorithm online real time routing with the proposed approach is given as below:

Algorithm MRTDP (W_i)

Input: MRTDP $G(V, E, c, d, r, \sigma, \alpha)$ with source node 1, the sink node $|V| = n, s \ and \ t.$

Output: A feasible optimized path for α_i

STEP0: Calculate Wedge(u, v) for each edge using cost formula and using Dijkastra algorithm to find shortest path with path weight $W_{path}(s, t)$ from s to t for tuning factor α_i.

STEP1:If $W_{path}(s, t)$ is $W_{sd}<W_{path}<W_{mean}$
then go to **STEP2**

else
go to **STEP0** with tuning factor αi+1

STEP2:If reliability of path s to t, is $r_{sd}<r_{path}<r_{mean}$
then go to **STEP4**

else
go to **STEP0** with tuning factor α_{i+1}

STEP3:If delay of path s to t, is $d_{sd}<d_{path}<d_{mean}$
then Path is feasible and optimized QoS.

else
go to **STEP0** with tuning factor α_{i+1}

In pre-computation part statistical analysis has been done to form the high performance region. The mean and standard deviation are the bounded region for the probability scale is inverse of natural log (-ln) to get the minimum cost values of weight for the usability of Dijkstra's algorithm.

Theorem 1. The MRTDP of $G(V, E, c, d, r, \sigma, \alpha)$ with respect to cost is unique having $O\ (E + VlogV)$ time complexity.

For the next part any values of reliability, capacity and delay for specific application requirement path weight is calculated using proposed algorithm and then check for the path whether it comes in feasible region or not.

III. EXPERIMENTAL SETUP AND RESULT DISCUSSION

In this section we will discuss about the experimental setup and result discussion. The experiment has been done on Ubuntu 15.04 and simulation part has been done using NS2 simulator. 1000 simulation for the given topology in figure 2 has been given to get the high performance region.

The scale used in the performance envelope for reliability is –ln, and for delay its seconds. The region boundaries are bounded in the statistical parameters

TABLE I. HIGH PERFORMANCE ENVELOPE FOR THE FIGURE 2 WITH TUNING FACTOR 1

Constraint	***Reliability (lnscale)***	***Delay***	***Weight***
Mean	1.201	14.9	32.39
S.D.	1.45826	6.419588	28.53873858

For the tuning factor 0.00625 and 0.025 the performance bound is given below is table 2 and table 3.

TABLE II. HIGH PERFORMANCE ENVELOPE FOR THE FIGURE 2 WITH TUNING FACTOR 0.025

Constraint	***Reliability(lnscale)***	***Delay***	***Weight***
Mean	3. 327	15.1	711.238
S.D.	3. 764035	6.454112	221.7036

TABLE III. HIGH PERFORMANCE ENVELOPE FOR THE FIGURE 2 WITH TUNING FACTOR 0.00625

Constraint	***Reliability(lnscale)***	***Delay***	***Weight***
Mean	3.327	15.1	2726.72
S.D.	3.764035	6.454112	860.1086

Now take any random set of values taken for different values of tuning factor the path given directly for the tuning factor 1, 0.025 and 0.00625.

The best paths for the different tuning factor are given below.

TABLE IV. HIGH PERFORMANCE PATH WITH DIFFERENT TUNING TIME

Tuning time (sec)	***160***	***40***	***1***
Pest path	0..4..6..1..5	0..2..7..6..3..1..5	0..2..7..6..3..1..5

IV. CONCLUSION

In this letter, we proposed a tuned routing algorithm for most reliable and quickest path. We also show that the worst time required to solve the MRTDP problem is same as the basic approach of dijkastra algorithm. Further the MRTDP problem can be extended to online real time routing

ISSN: 0971-1260
Vol-22-Issue-17-September-2019

using software defined network (SDN) with integrity of different application where different QoS parameters are required which will be extended in a forthcoming paper.

REFERENCES

[1] Y. Chen and Y. Chin, "The quickest path problem," *Computers & Operations Research,* vol. 17, pp. 153-161, 1990.

[2] G. Xue, "End-to-end data paths: quickest or most reliable?," *IEEE Communications Letters,* vol. 2, pp. 156-158, 1998.

[3] G.-H. Chen and Y.-C. Hung, "On the quickest path problem," *Information Processing Letters,* vol. 46, pp. 125-128, 1993.

[4] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network flows*: Elsevier, 2014.

[5] S. Ruzika and M. Thiemann, "Reliable and restricted quickest path problems," in *Network Optimization*, ed: Springer, 2011, pp. 309-314.

[6] H. I. Calvete, L. del-Pozo, and J. A. Iranzo, "Algorithms for the quickest path problem and the reliable quickest path problem," *Computational Management Science,* vol. 9, pp. 255-272, 2012.

[7] Y.-K. Lin, "Spare routing reliability for a stochastic flow network through two minimal paths under budget constraint," *IEEE transactions on reliability,* vol. 59, pp. 2-10, 2010.

[8] Y.-K. Lin, C.-F. Huang, L. C.-L. Yeng, and Y.-L. Cho, "Project Reliability Interval for a Stochastic Project Network Subject to Time and Budget Constraints," *IEEE Transactions on Reliability,* vol. 66, pp. 689-699, 2017.

[9] R. Kumar and P. Cholda, "A framework for continuity of mission-critical network services," in *Advanced Networks and Telecommuncations Systems (ANTS), 2015 IEEE International Conference on*, 2015, pp. 1-3.

[10] K. Gopal and R. Kumar, "An algorithm for computing the best-performing path in a computer network," *International Journal of Performability Engineering,* vol. 3, pp. 203-212, 2007.

[11] Z. Duan, Z.-L. Zhang, and Y. T. Hou, "Service overlay networks: SLAs, QoS, and bandwidth provisioning," *IEEE/ACM Transactions on Networking (TON),* vol. 11, pp. 870-883, 2003.

[12] E. L. Følstad and B. E. Helvik, "The cost for meeting SLA dependability requirements; implications for customers and providers," *Reliability Engineering & System Safety,* vol. 145, pp. 136-146, 2016.

[13] M. L. Fredman and R. E. Tarjan, "Fibonacci heaps and their uses in improved network optimization algorithms," *Journal of the ACM (JACM),* vol. 34, pp. 596-615, 1987.