ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 2, Issue 2, February 2015

Finest Trail Collection using CSPF Schemes

R.Tino Merlin
Francis Xavier Engineering College

Abstract: In particular, finding the cheapest delay-constrained path is critical for real-time data flows such as voice/video calls. Because it is NP-complete, much research has been designing heuristic algorithms that solve the -approximation of the problem with an adjustable accuracy. A common approach is to discretize (i.e., scale and round) the link delay or link cost, which transforms the original problem to a simpler one solvable in polynomial time. The efficiency of the algorithms directly relates to the magnitude of the errors introduced during discretization. In this paper, we propose two techniques that reduce the discretization errors, which allow faster algorithms to be designed. Reducing the overhead of computing constrained shortest paths is practically important for the successful design of a high-throughput QoS router, which is limited at both processing power and memory space. Our simulations show that the new algorithms reduce the execution time by an order of magnitude on power-law topologies with 1000 nodes. The reduction in memory space is similar.

1. INTRODUCTION

The algorithms for computing the constrained shortest paths can be used in many different circumstances, for instance, laying out virtual circuits in ATM networks, establishing wavelength switching paths in fiber-optics networks, constructing label switching paths in MPLS based on the QoS requirements in the service contracts, or applying together with RSVP. There are two schemes of implementing the QoS routing algorithms on routers. The first scheme is to implement them as on-line algorithms that process the routing requests as they arrive. In practice, on-line algorithms are not always desired. When the request arrival rate is high (major gateways may receive thousands or tens of thousands of requests every second), even the time complexity of Dijkstra's algorithm will overwhelm the router if it is executed on a per-request basis.1 To solve this problem, the second scheme is to extend a link-state protocol (e.g., OSPF) and periodically pre-compute the cheapest delay-constrained paths for all destinations, for instance, for voice traffic with an end-to-end delay requirement of 100 ms. The computed paths are cached for the duration before the next computation. This approach provides support for both constrained unicast and constrained multicast. The computation load on a router is independent of the request arrival rate. Moreover, many algorithms, including those we will propose shortly, have the same time complexity for computing constrained shortest paths to all destinations or to a single destination. This paper studies the second scheme. A path that satisfies the delay requirement is called a feasible path. Finding the cheapest (least-cost) feasible path is NP-complete. There has been considerable work in designing heuristic solutions for this problem. Xue [1-2] and Juttner et al. [3] used the Lagrange relaxation method to approximate the delay-constrained least-cost routing problem. However, there is no theoretical bound on how large the cost of the found path can be Korkmaz and Krunz used a nonlinear target function to approximate the multi-constrained least-cost path problem [4]. It was proved that the path that minimizes the target function satisfies one constraint and the other constraints multiplied by $\sqrt[\lambda]{k}$, where λ is a predefined constant and K is the number of constraints. However, no known algorithm can find such a path in polynomial time. Ref. [5] proposed a heuristic algorithm, which has the same time complexity as Dijkstra's algorithm. It does not provide a theoretical bound on the property of the returned path, nor provide conditional guarantee in finding a feasible path when one exists. In addition, because the construction of the algorithm ties to a particular destination, it is not suitable for computing constrained paths from one source to all destinations.

Another thread of research in this area is to design polynomial time algorithms that solve the NP complete problem with an accuracy that is theoretically bounded. Let m and n be the number of links and the number of nodes in the network, respectively.

Given a small constant ε, Hassin's algorithm [6] has a time complexity of $0((mn/\varepsilon) \log \log (UB/LB))$, where UB and LB are the costs of the fastest path and the cheapest path from the source node to the destination node respectively. The algorithm finds a feasible path if there exists one. The cost of the path is within the cost of the cheapest feasible path multiplied $(1 + \varepsilon)$. Lorenz and Raz improved the time complexity to $0(mn (1/\varepsilon + \log n))$. [7-10]

One common technique of the above algorithms [11], [12], [13-15] is to discretize the link delay (or link cost). Due to the discretization, the possible number of different delay values (or cost values) for a path is reduced, which makes the problem solvable in polynomial time. The effectiveness

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 2, Issue 2, February 2015

of this technique depends on how much error is introduced during the discretization. The existing discretization approaches have either positive discretization error for every link or negative error for every link. Therefore, the discretization error on a path is statistically proportional to the path length as the errors on the links along the path add up. In order to bound the maximum error, the discretization has to be done at a fine level, which leads to high execution time of the algorithms.

Given the limited resources and ever-increasing tasks routers, it is practically important to improve the efficiency of the network functions. While QoS routing is expensive due to its nonlinear nature, it has particular significance to reduce the router's overhead in computing the constrained shortest paths. In this paper, we propose two techniques, randomized discretization and path delay discretization, which reduce the discretization errors and allow faster algorithms to be designed.

The randomized discretization cancels out the link errors along a path. The larger the topology, the greater the error reduction. The path delay discretization works on the path delays instead of the individual link delays, which eliminates the problem of error accumulation. Based on these techniques, design fast algorithms to solve the -approximation of the constrained shortest-path problem. We prove the correctness and complexities of the algorithms.

2. PROBLEM DEFINITION

The insight is that if we can reduce the error introduced by discretization without using a larger λ, we can improve the performance of the algorithm. We develop two new techniques. The first one is called randomized discretization. It rounds to ceiling or to floor according to certain probabilities. The idea is for some links to have positive errors and some links to have negative errors. Positive errors and negative errors cancel out one another along a path in such a way that the accumulated error is minimized statistically. We will prove that, when the following discretization approach is used, the mean of the accumulated error on a path P is zero and the standard deviation is bounded by.

$$r\sqrt{l(P)}/2\lambda.$$

Round randomly (RR): For every link (u, v) the delay value is divided by r /λ. If the result is not an integer, it is rounded to the nearest smaller integer or to the nearest larger integer randomly such that the mean error is zero.

The discretized delay of a path P is

$$d^r(P) = \sum_{(u,v)\in P} d^r(u,v)$$

The discretization error of a link (u, v) is

$$\Delta^r(u,v) = d(u,v) - d^r(u,v)\frac{r}{\lambda}$$

The discretization error of a link is path is

$$\Delta^r(P) = \sum_{(u,v)\text{ on }P} \Delta^r(u,v) = d(P) - d^r(P)\frac{r}{\lambda}$$

We design the randomized discretization algorithm (RDA), which is based on Dijkstra's algorithm but considers two additive metrics, delay and cost. It uses RR to discretize the link delays. We will prove that it solves the ε -approximation of DCLC.

Path Delay Discretization:

Each unit of discretized delay represents the amount r/λ of real delay. Due to rounding, each time discretization is performed, a discretization error up to r/λ is introduced between the discretized delay and the real delay. The maximum discretization error of a path is determined by the number of times that discretization is performed on the path. RTF, RTC, and RR perform discretization at the link level. Because discretization is carried out on each link, the maximum error on the path is linear to the path length. In order to achieve ε-approximation, the accumulated error on a path cannot be too large. There are two ways to reduce the error. One is to use a larger, which increases the execution time of an algorithm whose complexity is linear to λ. The other way is to reduce the number of discretizations performed on the path.

Our second technique to control error is to perform discretization on the path level, using the interval partitioning method for combinatorial approximation. For a path P, ideally, discretization is performed once as follows.

$$d'(P) = \left\lfloor \frac{d(P)}{r}\lambda \right\rfloor$$

Because only one discretization is performed, the maximum discretization error on any path is bounded by r/λ, independent of the path length.

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 2, Issue 2, February 2015

We design the path discretization algorithm (PDA) based on the above intuition. The algorithm solves the ε–approximation with the same worst-case complexity as RDA. However, its average execution time is better than RDA according to our simulations.

3. PERFORMANCE ANALYSIS

5.1.5 Data Transferring

Blue color is used to identify whether the data is transfer or not from one node to another node is shown in fig 1.

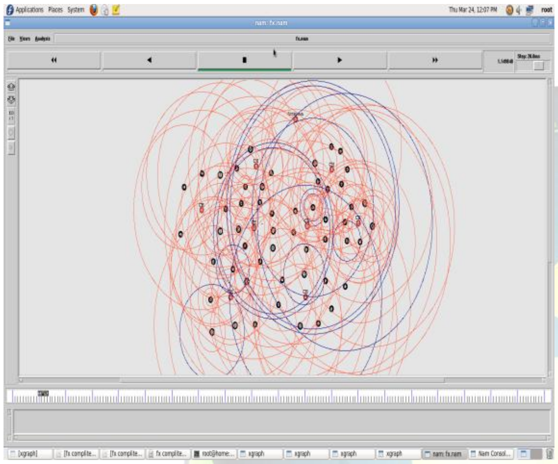

Fig-1 Data Transferring

5.1.6 Indicates the Watchdog

In this fig 1 is used to know about the watchdog .The watchdog will be placed in every group of node. In a group of node which has a higher energy is act as a watchdog in every group of nodes.

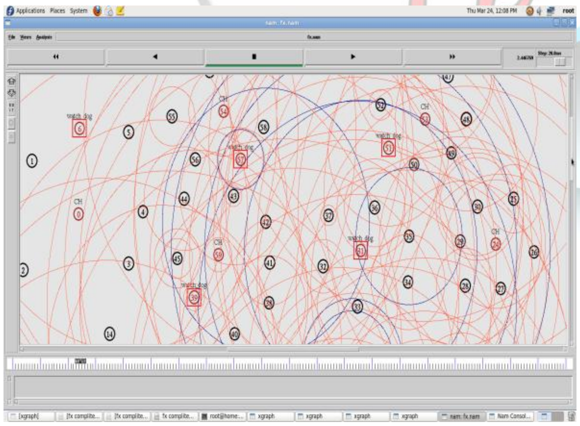

Fig-2 Indicates the Watchdog

5.1.7 Detecting the Malicious Node

The watchdog node will be finding about the malicious node and give alert signal to another node and cluster head. The data will not be transfer to the malicious node. Sometimes the data will be send to malicious node at that time the will be lost in fig-2.

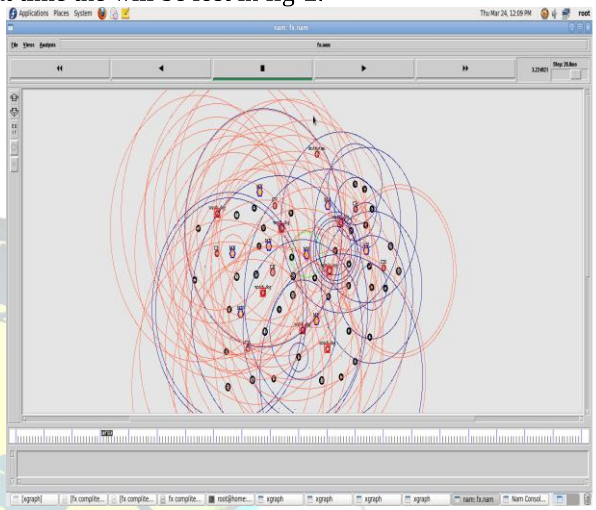

Fig-3 Detecting the Malicious Node

5.2.2 Energy

In this graph is shown 4 the difference between delay and time. The red color will be indicates the node energy and green color will be indicates the watchdog energy .

Table 1Energy

S.No	Node		Watchdog	
	Time	Delay	Time	Delay
1	0.0000	0.0000	0.0000	0.0000
2	10.0000	1.6500	10.0000	3.2000
3	20.0000	1.8000	20.0000	3.3000
4	30.0000	2.1800	30.0000	3.4000
5	40.0000	2.3800	40.0000	3.6500

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 2, Issue 2, February 2015

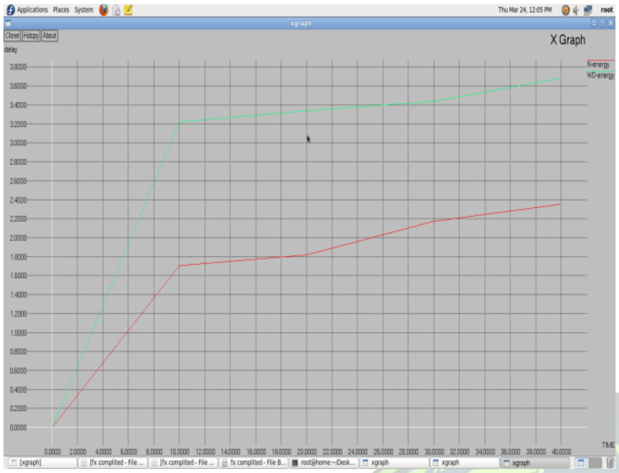

Fig-4 Energy Graph

4. CONCLUSION

While the previous approaches (RTF and RTC) build up the discretization error along a path, the new techniques either make the link errors to cancel out each other along the path or treat the path delay as a whole for discretization, which results in much smaller errors. The algorithms based on these techniques run much faster than the best existing algorithm that solves the ε-approximation of DCLC

REFERENCES

[1]. Raja Ratna, S, Ravi, R & Beulah Shekhar, ‘An intelligent approach based on neuro-fuzzy detachment scheme for preventing jamming attack in wireless networks’, Journal of Intelligent and Fuzzy Logic, Vol. 28, No. 2, pp. 801-820, Jan 2015, DOI:10.3233/IFS-141363. ISSN. 1064-1246, 2015.

[2]. Khongbantabam Susila Devi & Ravi,R 'Medical E-mail Spam Classification using a Score Based System and Immune System Embedded with Feature Selection Process', Journal of Pure andAppliedMicrobiology,Print-ISSN:0973-7510,vol.9,Spl.Edn.1,pp.673-680, 2015.

[3]. Raja Ratna, S, Ravi, R & Beulah Shekhar, ‘Mitigating Denial of Service attacks in Wireless networks’, International Journal of Advanced Research in Computer Engineering and Technology, vol. 2, no.5, pp. 1716-1719, 2013.

[4]. Raja Ratna, S & Ravi, R 2014, ‘Malicious Route Defending Reliable-Data Transmission Scheme for Multi Path Routing in Wireless Network’, World Academy of Science, Engineering and Technology, vol.8, no.12, pp. 1950-1955.

[5]. Raja Ratna, S & Ravi, R 2015, ‘Code Based Rescheduling: A Highly Reliable Inter Layer Scheme to Prevent Physical layer Threat in Wireless Network’, Global Journal of Pure and Applied Mathematics, vol.11, no.1, pp.75-90.

[6]. Muthukumaran, N & Ravi, R 2014, 'Quad Tree Decomposition based Analysis of Compressed Image Data Communication for Lossy and Lossless using WSN', World Academy of Science, Engineering and Technology, Print-ISSN:2010-376X, Volume. 8, No. 9, pp. 1543-1549.

[7]. Khongbantabam Susila Devi & Ravi, R 2015, 'A New Feature Selection Algorithm for Efficient Spam Filtering using Adaboost and Hashing Techniques', Indian Journal of Science and Technology, Print-ISSN:0974-6846, vol. 8.no.13,pp. 65753.

[8]. Khongbantabam Susila Devi & Dr. R. RAVI2015, ‘ A new algorithm for similarity preserving hashing based on Mvhash- Damerau Levenshtein for email filtering’, International Journal of research in Computer Engineering and Electronics, vol. 4, no.1. pp. 1-16.

[9]. Dr. R. Ravi, A. Jenefa, “Classifier A Real time detection System for detecting Suspicious URLs in twitter Stream”, International Journal of advance research in computer science and management studies Vol. 2, No 2, Feb 2014.

[10]. Dr. R. Ravi, M. Karthika, “An efficient and affordable user authentication protocol defiant to password pinching and reclaiming.”, International Journal of advance research in computer science and management studies Vol. 2, No 2, Feb 2014.

[11]. Dr. R. Ravi, M. Jenefa, “A Machine Doctor for Diagnosing Ophthalmology problems using Neural Network”, International Journal of Advance Research in Computer Engineering and Technology Vol. 2, No 2, Feb 2014.

[12]. Dr. R. Ravi, M. Jenefa, “Enhanced Secure Data transmission with Hash technique and level analysis for MANETs”, International Journal of computer science and management Research Vol. 3, No 2, March 2014.

[13]. Dr. R. Ravi, S. Vasanthi, “Estimating the Exposure of Network Mechanism to High-Tech DDoS Attacks”, International Journal of Advance Research in Computer Engineering and Technology Vol. 3, No 3, March 2014.

[14]. Muthukumaran, N & Ravi, R, 'Design and analysis of VLSI based FELICS Algorithm for lossless Image Compression', International Journal of Advanced Research in Technology (IJART), Print-ISSN: 6602-3127, Vol. 2, No. 3, pp. 115-119, March 2012.

[15]. Dr. R. Ravi, A. Deepika, “Image Decomposition and Restoration for Blurred Images using Filtering Techniques”, International Journal of Advance Research in Computer Engineering and Technology Vol. 3, No 3, March 2014.