Active Route Choice Algorithm for Sheltered Data Transmission

B.Muneendra Nayak[1], E.Madhusudhana Reddy[2]

[1]Lecturer, Dept of Computer Engineering, SVPPC (TTD), Tirupati.
[2]Professor, Dept of C.S.E, MITS, Madanapalle, Andhra Pradesh, india

***Abstract*: Security has become one of the major issues for data communication over wired and wireless networks and also providing a two way shortest path routing in wireless networks. Different from the past work on the designs of cryptography algorithms and system infrastructures, we will propose a dynamic routing algorithm that could randomize delivery paths for data transmission. 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**

***Keywords:* cryptography algorithms, constrained shortest paths, QoS routing**

I. INTRODUCTION

To improve the security of data transmission, we use a distance-vector-based algorithm for dynamic routing. In many distance-vector-based implementations, e.g., those based on RIP, each node Ni maintains a routing table in which each entry is associated with a tuple and Next hop denote some unique destination node, an estimated minimal cost to send a packet to t, and the next node along the minimal-cost path to the destination node, respectively.

A major obstacle against implementing distributed multimedia applications (e.g., web broadcasting, video teleconferencing, and remote diagnosis) is the difficulty of ensuring quality of service (QoS) over the Internet. A fundamental problem that underlies many important network function such as QoS routing, MPLS path selection, and traffic engineering, is to find the constrained shortest path—the cheapest path that satisfies a set of constraints. It is the cheapest path whose end-to-end delay is bounded by the delay requirement of a time- sensitive data flow. The additional bandwidth requirement, if there is one, can be easily handled by a pre- processing step that prunes the links without the required bandwidth from the graph. 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. 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 study of even very expensive algorithms is not a completely theoretical pursuit as they can yield valuable insights. A classic example is the initial PTAS for Euclidean TSP which had prohibitive running time, yet 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 [12] and Juttner *et al.* 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 . It was proved that the path that minimizes the target function satisfies one constraint and the other constraints multiplied by , where is a predefined constant and is the number of constraints. However, no known algorithm can find such a path in polynomial time. Ref. 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. For this task, it is slower than the algorithms proposed in this paper by two orders of magnitude based on our simulations.

II. QoS

In the field of computer networking and other packet-switched telecommunication networks, the traffic engineering term *quality of service* (QoS) refers to resource reservation control mechanisms rather than the achieved service quality. Quality of service is the ability to provide different priority to different applications, users, or data flows, or to guarantee a certain level of performance to a data flow. For example, a required bit rate, delay, jitter, packet dropping probability and/or bit error rate may be guaranteed. Quality of service guarantees are important if the network capacity is insufficient, especially for real-time streaming multimedia applications such as voice over IP, online games and IP-TV, since these often require fixed bit rate and are delay sensitive, and in networks where the capacity is a limited resource, for example in cellular data communication. In the absence of network congestion, QoS mechanisms are not required. A network or protocol that supports QoS may agree on a traffic contract with the application software and reserve capacity in the network nodes, for example during a session establishment phase. During the session it may monitor the achieved level of performance, for example the data rate and delay, and dynamically control scheduling priorities in the network nodes. It may release the reserved capacity during a tear down phase. A best-effort network or service does not support quality of service. An alternative to complex QoS control mechanisms is to provide high quality communication over a best-effort network by over-provisioning the capacity so that it is sufficient for the expected peak traffic load.
In the field of telephony, **quality of service** was defined in the ITU standard X.902 as "A set of quality requirements on the collective behavior of one or more objects". Quality of Service comprises requirements on all the aspects of a connection, such as service response time, loss, signal-to-noise ratio, cross-talk, echo, interrupts, frequency response, loudness levels, and so on. A subset of telephony QoS is Grade of Service (GOS) requirements, which comprises aspects of a connection relating to capacity and coverage of a network, for example guaranteed maximum blocking probability and outage probability. QoS is sometimes used as a quality measure, with many alternative definitions, rather than referring to the ability to reserve resources. Quality of service sometimes refers to the level of quality of service, i.e. the guaranteed service quality. High QoS is often confused with a high level of performance or achieved service quality, for example high bit rate, low latency and low bit error probability. See also Relation to subjective quality measures below.

III. TREE MODEL

Nodes such as routers or switches, and L_{phys} defines the link between them. A source sender probe is called a root and is labeled as 0□V_{phys} . A set of receivers is denoted as R□V_{phys} . The tree model is defined as a binary tree, where there is no possibility for two diverged paths to intersect one more time. It is possible to move from a physical model to a logical model, which is simpler to manage and takes into consideration all the characteristics of the physical one. A logical source tree can be defined □=(V,L).

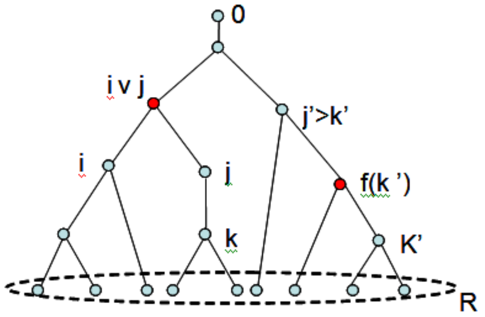

Figure 1: A logical tree. An ancestor j'>k' and the first common ancestor are defined. f(k') represents the parent of k'.

IV. DELAY MODEL

Delay model [10] is the method to define a delay over the links or a path in a tree model. The node probe is the root. Let <i,j> be a packet pair sent to destination i and j respectively. A packet pair can be described as a train consisting of two carriages, one of which goes behind the other. They cover a common path till a branch node and are directed to nodes i and j. The common path is the sequence of a set of links down to node i v j. Let us denote p(i,j) a sequence of links traversed by at least one of the two members of the packet pair. Let k□p(i,j) be,
and denote S(k)□□1,2□ as the set of packets which across k, where 1 and 2 are two members of packet pairs sent to i and j.

$X_k(l)$, with l □ $G(k)$ is observed, and it represents the cumulated delay along the root to node k.

A measurement represents the end-to-end delay from the root to the end receivers i and j. X_{ij} □ (X_i (1), X_j (2)) expresses this couple of one way delay. Where X_i (1) is the delay from the root to the destination i for the first member and X_j (2) is the delay from the root to the destination j for the second member. This is the only measurement which can be used and it takes into consideration the definition of the tomography with the active external measurement.

It is also possible to apply a delay model to the packet pairs. Each member of a packet pair traverses a common path to arrive to the respective destination. This common path is vital for the delay model. Let D_k and D_k□ be a pair of random variable for each node k. D_k and D_k□ represents the estimated value of delay over a link (f(k),k)□L for the first member and for the second one of the packet pair. They can have values in the real line R□ □□. R□ because a delay cannot be negative. The value □ can be assumed if the packet is dropped on a link and is not able to reach the address receiver. For the hypothesis $D_0 = D_0$□ equals to 0. Two kinds of independence are required for this model: for the delay between different pairs and for the delay within the same pair but over different links. For k□V, E_k □ D_k the difference between the delay experienced by □ D_k□

the first and the second member of pair crossing k. cumulated delay between two members in k.E_k is a quantity which measures how large is the Another hypothesis is to consider E_k =0. This is a rough approximation. The practical application shows a different delay and E_k □0. The state of the network can be not stationary. When a packet is sent, it can meet different states of the network, because it is time-dependent. The first and the second member test the network in different time because they are distanced even if the temporal gap between them is small. For example, a bottleneck can have a different impact on the first and the second member of a packet pair. E_k can never have a null value, even if there is a perfect back-to-back packets, for example, in case of a low traffic state of a network. The second member must wait the transmission of the first one. For this reason there is always a delay which is impossible to avoid.

V. FIXED BIN SIZE DISCRETE MODEL

Let Q be a set of finite delays. Delays are discretized to Q and D_k model defines the set Q as $Q \square \square 0,q,...,Bq,\square$ □takes a value in Q. Fixed bin size discree where q is a fixed bin size chosen a priori. The accuracy of the discretization depends on the choice of q. A smaller bin size provides more accuracy to the estimation of the probability distribution of D_k . The continuous model is a case of discrete model with infinite bins obtained with $\lim_{q \square 0}$.

Each value contained in i-th interval will have a unique value iq. This model introduces an error of quantization $q/2$. Fixed bin size discrete model is named (q,B) model for the structure of the set Q. The estimate of □ is the goal of inference problem. It can be obtained by using the maximum likelihood approach. The set of measurements X defines the likelihood function. It is discretized to the set Q, therefore, likelihood function is a discrete function. A measurement represents two one-way delays of the first and the second members of a packet pair sent. Above figure shows how this discretization is obtained from a continuous time.

The bidimensional discretization allows defining the space of measurement□. Let □=QxQ be the space of the possible values taken by the measurements after the discretization of the set Q. For each pair of receivers i,j it is possible to define a m-th measurement $X^{i,j(m)} \overset{2}{=} x_{ij}$ □□. For m=1 to n packets pair <i, j> sent a collection of measurements $x_{i\,j}$ is made. It is important to observe that the measurements are made only when the end receivers have been chosen.

Let us denote the number of packet pairs, for which $X^{i,j(m)} = x_{ij}$, by $n(x_{i,j})$. It represents a bidimensional histogram on □ space. It depends, in fact, on the time from m=1 to n $X^{i,j(m)} = x_{ij}$.

Let us denote the maximum likelihood function of the measurement X by *lik*(X;□). Using the, let L(X;□) be the log-likelihood of the measurement X. To estimate □ by using MLE, it is necessary to maximize the If the set of measurements is obtained, a function can be maximized depending on □ only. The use of the equation does not provide a direct expression for $\hat{\square}$. A more complex approach [11, 12] can be used in the applying the Expectation Maximum algorithm. This algorithm allows to iteratively obtain □, step by step researching a local maximum of the likelihood function. Let us denote $\hat{\square}^{(l)}$ the value of □ at *l-th* step. The algorithm works until the estimated value $\hat{\square}^{(l)}$, reaches a stationary solution.

A stationary solution is a local point of maximum of the function where the algorithm reaches the steady state

$\hat{\square} \;\square\; \hat{\square}^{l} \;\square\; \hat{\square}^{L}$. L represents the necessary steps to get the stationary solution.

The problem is that $n_k(d)$ is an unknown value, which makes this approach infeasible. How can n_k (d) be computed if it is to infer d and its probability is to be estimated? It can be done by estimating the maximum of using the Expectation Maximum algorithm. ------

CONCLUSION

In this paper, we proposed two techniques, randomized discretization and path delay discretization, to design fast algorithms for computing constrained shortest paths. But 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 - approximation of DSDV.

REFERENCES

[1]. Shigang Chen, Meongchul song, Sartaj sahni: ―Two Techniques for Fast Computation of Constrained Shortes Paths‖, 2008

[2]. J. Breckling, Ed., The Analysis of Directional Time Series: Applications to Wind Speed and Direction, ser. Lecture Notes in Statistics. Berlin, Germany: Springer, 1989, vol. 61.

[3]. S. Zhang, C. Zhu, J. K. O. Sin, and P. K. T. Mok, ―A novel ultrathin elevated channel low-temperature poly-Si TFT,‖ IEEE Electron Device Lett., vol. 20, pp. 569–571, Nov. 1999.

[4]. M. Wegmuller, J. P. von der Weid, P. Oberson, and N. Gisin, ―High resolution fiber distributed measurements with coherent OFDR,‖ in Proc.ECOC'00, 2000, paper 11.3.4, p. 109.

[5]. R. E. Sorace, V. S. Reinhardt, and S. A. Vaughn, ―High-speed digitalto-RF converter,‖ U.S. Patent 5 668 842, Sep. 16, 1997. (2007) The IEEE website. [Online]. Available: http://www.ieee.org/

[6]. M. Shell. (2007) IEEEtran webpage on CTAN. [Online]. Available:http://www.ctan.org/tex-archive/macros/latex/contrib/IEEEtran/

[7]. FLEXChip Signal Processor (MC68175/D), Motorola, 1996.

[8]. PDCA12-70 data sheet,‖ Opto Speed SA, Mezzovico, Switzerland.

[9]. A. Karnik, ―Performance of TCP congestion control with rate feedback: TCP/ABR and rate adaptive

[10]. TCP/IP,‖ M. Eng. thesis, Indian Institute of Science, Bangalore, India, Jan. 1999.

[11]. Padhye, V. Firoiu, and D. Towsley, ―A stochastic model of TCP Reno congestion avoidance and control,‖ Univ. of Massachusetts, Amherst, MA, CMPSCI Tech. Rep. 99-02, 1999.

[12]. Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification, IEEE Std. 802.11, 1997.