Learning Minimum Delay Paths in Service Overlay Networks

Hong Li, Lorne Mason and Michael Rabbat
Electrical and Computer Engineering Department
McGill University, Montreal, Canada
hong.li2@mail.mcgill.ca, {lorne.mason,michael.rabbat}@mcgill.ca

Abstract

We propose a novel approach using active probing and learning techniques to track minimum delay paths for real-time applications in service overlay networks. Stochastic automata are used to probe paths in a decentralized, scalable manner. We propose four variations on active probing and learning strategies. It can be proved that our approach converges to the user equilibrium for minimum delay routing. The performance of these strategies is studied via fluid simulations of a model of AT&Ts backbone network. The simulation results show that the proposed strategies converge to the minimum delay paths rapidly. We also observe, via simulation, that our approach scales well in the size of the service overlay network.

1. Introduction

With the trend of service convergence in next generation networks, real-time applications such as *voice over IP* (VoIP), video streaming, and IPTV are drawing significant interest from industry and consumers. However, the current best-effort Internet routing cannot satisfy the strict *quality of service* (QoS) requirements for real-time applications, such as low delay.

Service overlay networks (SONs) have been employed as a cost effective way of improving quality of service over the current best-effort Internet. Service overlay gateways in a SON are connected via virtual overlay links above the transportation layer. These service overlay gateways can be used to provide alternative routes with better quality of service characteristics than the direct path determined by underlying internet routing protocols. This paper studies the problem of identifying minimum delay paths in a service overlay network.

Low end-to-end delay is one of the most important quality of service requirements for real-time applications. For example, VoIP requires the mouth-to-ear delay to be less than 150ms. In many cases, the minimum delay path provides the best quality for VoIP and other real time streaming applications. However, the minimum hop path, as determined by the underlying network routing protocols, does not guarantee minimum delay. In [11], minimum delay paths are learned based on end-to-end delay estimation. In our work, no delay estimation is involved. The papers [8, 9] are very close to our work. The paper [9] studied the LRI algorithm where environment feedback is either 0 or 1, while in this paper the network delays are continuous values [7] where cross-correlation algorithm is applied. This paper is an extension of the paper [8], i.e. learning automata control only the probing process rather than the routing for all flows as in [8].

Our main contribution is to propose, analyze and simulate an active probing and learning algorithm using distributed learning automata to find the minimum mean delay paths in service overlay networks. In a service overlay network, where the performance of the underlying network is random and unknown to the overlay nodes, it is beneficial to actively probe the network performance to find the minimum delay overlay paths. In order to determine minimum delay routes we must probe the current network state. However, we desire a probing scheme that 1) does not inject an excessive amount of traffic just for probing, 2) that scales well to large overlay networks, and 3) that can adapt to changing network conditions. In order to address these three goals we adopt a reinforcement learning technique: the cross-correlation learning algorithm [7]. The basic idea is that, at each probing epoch, the path probed is chosen randomly according to a distribution over a set of possible paths between a given source and destination. The observed delay for this probe is then used to reinforce this path by increasing the probability of probing it again, and simultaneously decreasing the probability of probing the other paths. Precise details of the probing algorithm are given below. Because paths with poor performance receive little reinforcement, they are probed less frequently, and consequently, excessive probing resources are not wasted on paths that will likely yield low quality of service. Moreover, because each service overlay node locally runs independent learning automata, the algorithm scales gracefully to larger overlay networks.

The remainder of the paper is organized as follows. Section 2 presents the proposed active probing and optimal path learning strategies. The experiment set up and simulation results are presented in Section 3, and we conclude in Section 4.

2. Optimal routing with active probing and learning

In an autonomous system, minimum hop routing [3] cannot guarantee the optimal performance in terms of the mean network delays. This section describes an active probing and learning framework to determine the paths with minimum mean delay in a stationary dynamic network environment. The optimal paths between all sources and destinations are learned using distributed learning automata. Specifically, we adapt the cross-correlation learning algorithm, described in [7], to probe paths through a service overlay network.

Given a full mesh service overlay network modeled as a graph $G = (V, E)$, where $V = \{1, ..., m\}$ is the set of nodes in the graph, E is the set of directed links in the graph, we introduce one learning automaton for each source-destination pair $(i, k), i, k \in V, k \neq i$. Let $\pi_{ij}^k(t), i, j, k \in V$, denote the probability of sending a probe from node i to destination k via next hop j at time t. The automaton's state probability vector for source-destination pair (i, k) at time t is $\bar{\pi}_i^k(t) = [\pi_{i1}^k(t), ..., \pi_{ij}^k(t), ..., \pi_{im}^k(t)]$, and it satisfies $\pi_{ii}^k(t) = 0$ and $\sum_j \pi_{ij}^k(t) = 1$. Probes arriving at node i destined for $k \neq i$ at time t are forwarded to a next hop selected randomly according to the distribution $\bar{\pi}_i^k(t)$. When a probe arrives at the destination a reply is transmitted back to the originating node, and the round-trip delay is measured. Let u denote the next hop node probed. Then, the learning automata probabilities $\bar{\pi}_i^k(t)$ (maintained at the source node, i) are updated using the cross-correlation learning algorithm [7], for $j = 1, \ldots, m$,

$$\pi_{ij}^k(t+1) = \pi_{ij}^k(t) + G\, z(u,t)\, (\delta_{ju} - \pi_{ij}^k(t)) \quad (1)$$

where G is the learning gain of the algorithm, $z(u, t)$ is the reward for this probe, and $\delta_{ju} = 1$ if $j = u$ was the node probed and $\delta_{ju} = 0$ otherwise. Let $\mathrm{d}(u, t)$ denote the measured delay, and let $\mathrm{d}_{\max}$ be a pre-defined maximum delay. The normalized feedback function (or reward strength [7]) is given by $z(u, t) = 1 - \frac{\mathrm{d}(u,t)}{\mathrm{d}_{\max}}$.

Theorem 2.1 *When the learning gain G is sufficiently small, $\pi_{ij}^k(t)$ in the cross-correlation learning algorithm converges to the user equilibrium θ_{ij}^k with ϵ-optimality for the minimum delay routing problem, i.e. $\lim_{t\to\infty} \mathrm{P}\{|\pi_{ij}^k(t) - \theta_{ij}^k| > \epsilon\} = 0$, where $\theta_{ij}^k = 1$ if node k is the optimal next hop for node i to reach node j, otherwise, $\theta_{ij}^k = 0$.*

An outline of the proof is as follows. By setting G sufficiently small, the stochastic approximation specified by the cross-correlation learning algorithm satisfies the conditions for Kushner's weak convergence method [5] and converges to an Ordinary Differential Equation (ODE). The solution to the ODE is proved to be globally stable by finding a Lyapunov function for the end-to-end delays. Please refer to [6] for detailed proof.

Table 1. Active probing and learning

```
Input: $G = (V, E)$
Initialization: $t = 0, \forall k \neq i$, initialize $\bar{\pi}_i^k(0)$;
Iteration:
At time t:
(1) Node $i$ send a probe to the destination node $k$,
through a next hop node $u$ selected randomly
according to distribution $\bar{\pi}_i^k(t)$;
(2) Get feedback $\mathrm{d}(u, t)$ from the probe, compute
$z(u, t) = 1 - \frac{\mathrm{d}(u,t)}{\mathrm{d}_{\max}}$;
(3) update $\bar{\pi}_i^k(t)$, for $j = 1, ..., m$,
$\pi_{ij}^k(t+1) = \pi_{ij}^k(t) + G * z(u,t) * (\delta_{ju} - \pi_{ij}^k(t))$;
(4) t=t+1;
```

The active probing and learning algorithm is given in table 1. The probability of sending a probe from the source node i to the destination node k through a node j is determined by $\pi_{ij}^k(0)$. The initialization, $\pi_{ij}^k(0)$ can be thought of as representing our prior knowledge on the probability of node j being the optimal path next hop from i to k. Assuming no prior knowledge, we uniformly initialize the probability vector $\bar{\pi}_i^k(0)$ to ensure that all possible paths are explored, as given in equation (2).

$$\begin{cases} \pi_{ij}^k(0) = \frac{1}{m-1}, j \neq i, j \in \{1, ..., m\} \\ \pi_{ii}^k(0) = 0 \end{cases} \quad (2)$$

This method introduces random loops in the probing path. It can be shown that there is high probability of random loops on the probing path at the starting stage, and the probability increases with the network size. However, because routes with loops will always have higher delay than loop-free paths, the learning automata will automatically learn to avoid these loops [8].

In order to avoid these initial random loops, a geographical location aware initialization method can be used. Let $D(i,k)$ be the Euclidean distance (or the great circle distance on a sphere) between nodes i and k. Geographical location aware initialization, for $j = 1, ..., m$, as given by,

$$\pi_{ij}^k(0) = \frac{\mathrm{I}_{D(j,k)<D(i,k)}}{\sum_j \mathrm{I}_{D(j,k)<D(i,k)}} \qquad (3)$$

where $\mathrm{I}_{D(j,k)<D(i,k)}$ is an indicator function, guarantees only the nodes whose distances to the destination node k are less than that from the origin node i are probed.

We propose two approaches to actively probe and learn the minimum delay paths: hop-by-hop learning and end-to-end learning. In hop-by-hop learning, as illustrated in Fig. 1, each intermediate node on the forward path from a probe's source node S to its destination node D also receives feedback and performs learning updates. In end-to-end learning, only the source node S of the probe can learn from the end-to-end performance measured by the probe, which is easier to implement at the cost of slower learning speed. Please refer to the technical report [6] for more detail.

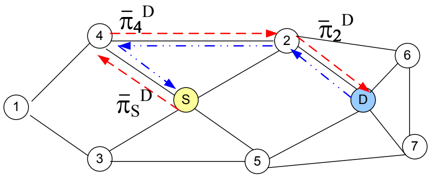

Figure 1. Hop-by-hop learning of the optimal path from node S to destination node D. $bar\pi_i^k$ **at each node is initialized uniformly.**

The hop-by-hop learning allows all the intermediate nodes on the forward path of a probe to get feedback from the probe. As shown in Fig. 1, the forward path of a probe is randomly chosen. On the forward path of the probe sent from S to D, each intermediate node i chooses its next hop node randomly according to its state probability vector $\bar{\pi}_i^D$. The probe is required to record all the intermediate nodes it goes through until it reaches the destination node D. When the probe reaches its destination node D, it has to follow the exact reverse path of the forward path to go back to S. For example, as shown in Fig. 1, the nodes on the forward path of the probe with source-destination (S,D) are S, 4, 2, D. The reverse path from D to S is then D, 2, 4, S. At each node i on the reverse path, the state probability vector $\bar{\pi}_i^D$ is updated according to the round trip time from node i to D.

3. Experiments

In the paper, a fractional Brownian motion (fBm) process is used to model the Internet traffic [10]. The gravity model is used to model the mean network traffic demand between ingress and egress points of a network [2, 13]. We also consider the time difference in different time zones so that the number of active users at each time zone varies with the GMT (Greenwich Mean Time) [2].

The network topology under study is derived from AT&T's backbone network [1, 12]. It includes 50 nodes located in the major cities in the United States.Our simulations are conducted in a fluid network [4] at timescale $\tau = 5$ms. We simulated the probing and learning process for a full mesh 10-node overlay network above the 50 PoP (Point of Presence) node model of AT&T's backbone network, as shown in Fig. **??**. The 10 overlay nodes are chosen randomly from the 50 PoP nodes. Each overlay node sends active probes periodically to all other overlay nodes every 5 ms. For a source-destination pair, the learned path is the path decided by the stochastic automata; the optimal path is the minimum mean delay path.

Let n denote the probe number, $\bar{d}(n)$ be the average delay on the learned paths. The mean delay on the learned paths for probe number n is computed as $\bar{d}(n) = \frac{1}{|V|(|V|-1)} \sum_{i,j \in V, i \neq j} d_n(i,j)$, where $d_n(i,j)$ is the delay measured by probe number n between node i and j. The mean delays $\bar{d}(n)$ on the learned paths with a learning gain of 0.001 and 0.01 are shown in Fig. 2. As can be seen, the learning speed increases proportionally with the learning gain for the simulated overlay network.

We also simulated the hop-by-hop learning method for larger overlay network. Fig. 3 shows the learning speed for hop-by-hop learning with uniform initialization in 10, 15, 20, 25 node overlay network. It can be seen that the convergence speed does not change much as the network size increases. Also note that larger overlay networks (20 or 25 nodes) converge at a slightly slower rate, but that in general, the size of the overlay network does not dramatically impact the number of probes required to learn minimum delay paths.

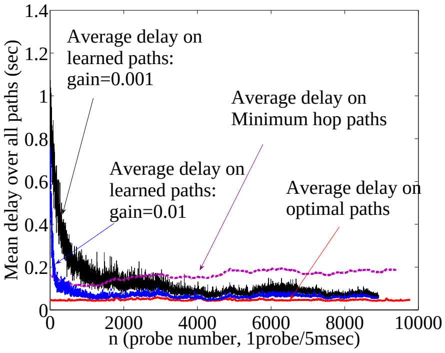

Figure 2. Mean delays on the learned paths (hop-by-hop learning) versus the mean delays on the optimal paths and those on the minimum hop paths.

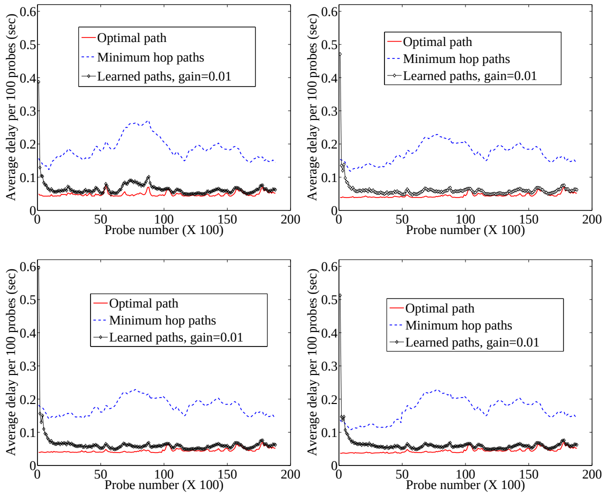

Figure 3. Hop-by-hop Learning result in a 10 (top left),15 (top right), 20 (bottom left) and 25 (bottom right) node overlay network, gain=0.01.

4. Conclusion

In the paper, we proposed a novel method to learn the minimum delay paths for each source-destination pair in service overlay networks. Based on the cross-correlation learning automata, we proposed four active probing and learning strategies to learn the optimal paths, which are uniformly initialized hop-by-hop learning, geographical location aware initialized hop-by-hop learning, uniformly initialized end-to-end learning, and the geographical location aware initialized end-to-end learning. The performance of the proposed active probing and learning strategies is simulated in service overlay networks over a model of the AT&T's network. The simulation results show that the learning method converges to the minimum mean delay paths very quickly (around 5 seconds for hop-by-hop learning in a 10-node overlay network), and the convergence speed scales well with the overlay network size. The cross-correlation learning algorithm can be proved to converge to the user equilibrium. Future work will focus on applying the proposed learning method for voice over IP packet routing in service overlay networks.

References

[1] G. R. Ash. *Traffic Engineering and QoS Optimization of Integrated Voice & Data Networks*. McGraw-Hill, 2006.

[2] A. Gunnar, M. Johansson, and T. Telkamp. Traffic matrix estimation on a large ip backbone: a comparison on real data. *Proc. IMC '04*, Oct. 2004.

[3] IETF. Rfc 2328: Ospf version 2. Apr. 1998.

[4] L. Jansen, I. Gojmerac, M. Menth, P. Reichl, and P. Tran-Gia. An algorithmic framework for discrete-time flow-level simulation of data networks. *20th ITC*, Jun. 2007.

[5] H. Kushner and F. Vázquez-Abad. Stochastic approximation methods for systems of interest over an infinite time horizon. *SIAM J. on Control and Optim*, (2):712–756, Oct. 1996.

[6] H. Li, L. Mason, and M. Rabbat. Learning minimum delay paths in service overlay networks. *Tech. Report, ECE, McGill University*, Mar. 2008.

[7] L. Mason. An optimal learning algorithm for s-model environments. *IEEE Trans. Auto. Control*, 18:493–6, Oct. 1973.

[8] L. Mason. Equilibrium flows, routing patterns and algorithms for store-and-forward networks. *Large Scale Systems*, 8:187–209, 1985.

[9] S. Misra and B. Oommen. Dynamic algorithms for the shortest path routing problem: Learning automata-based solutions. *IEEE trans. Sys. Man, and Cyb. part B: cyb.*, (6), Dec. 2005.

[10] I. Norros. On the use of fractional brownian motion in the theory of connectionless networks. *IEEE JSAC*, 13(6), Aug. 1995.

[11] V. Raghunathan and P. Kumar. On delay-adaptive routing in wireless networks. *43rd IEEE Conf. on Decision and Control*, Dec. 2004.

[12] N. Spring, R. Mahajan, D. Wetherall, and T. Anderson. Measuring isp topologies with rocketfuel. *IEEE/ACM Trans. Networking*, 12(1):2–16, Feb. 2004.

[13] M. Zhang, Y.and Roughan, C. Lund, and D. Donoho. Estimating point-to-point and point-to-multipoint traffic matrices: An information-theoretic approach. *ACM/IEEE Trans. Networking*, (5):947–960, Oct. 2005.