

SOURCE ROUTING IN NETWORKS WITH UNCERTAINTY: INFERENCE, SENSITIVITY AND PATH CACHING

XUN SU , GUSTAVO DE VECIANA

*Department of Electrical and Computer Engineering
University of Texas, Austin, TX 78712*

Abstract— In this paper we study source routing in an environment where imperfect state information is the norm. The uncertainty involved in several aspects of the routing process renders the route choices less than “optimal”. We start by conducting an experiment that compares the performance of an “inference”-based routing scheme to that of the traditional approach based on delayed link state broadcast. We then resort to a set of simple models to investigate to what extent the “crude” routing decisions based on limited statistical information conform to the ideal choices. In the conventional routing context, we identify a useful measure, the *gap*, which quantifies how successful a “crude” routing decision is likely to be. In the quality of service routing context we explore the possibility that a route choice based on limited statistical information is the “most likely” path to satisfy the user requirement. We also discuss the role of *critical points*, whose relative position affects the robustness of the routing decisions with respect to uncertain user requirement. Simulations establish the existence of *gap* and *critical point* in a realistic setup. The impacts of these observations on the effectiveness of a simple path caching scheme are then discussed.

I. INTRODUCTION

In order to provide end-to-end quality of service (QoS) guarantees to users, future networks are likely to require enhanced routing mechanisms. Indeed, current (unpinned) *hop-by-hop routing* is not particularly well suited to address *end-to-end* QoS requirements because they depend on the characteristics of the entire path. By contrast, *source routing* (or *explicit routing*) might be better suited to select paths satisfying user’s QoS requests, but it requires the availability and distribution of a large amount of network state information, possibly resulting in scalability problems. Moreover, since the state of the network is in constant flux, routers may be making decisions based on uncertain state information. In particular, future routing mechanisms may use hierarchical aggregation of state and/or topology information to deal with scalability issues, naturally resulting in a loss of accuracy. By the same token, routers may also have incomplete information concerning the characteristics of the user’s traffic. Indeed, since traffic is often best modeled as a stochastic process, there is a high degree of uncertainty in specifying traffic via crude source parameters and/or models and then translating these to an end-to-end QoS guarantee. In light of these new challenges, our goal is to investigate (QoS-sensitive) source routing based on uncertain network and source information.

To understand the characteristic of the routing problem we conduct the following idealized routing experiments. We assume that the user under consideration has no knowledge of the network state except for its current topology. Based on the accumulation of its past routing experience, a source routing engine infers the likelihood of success in using various routes, and uses

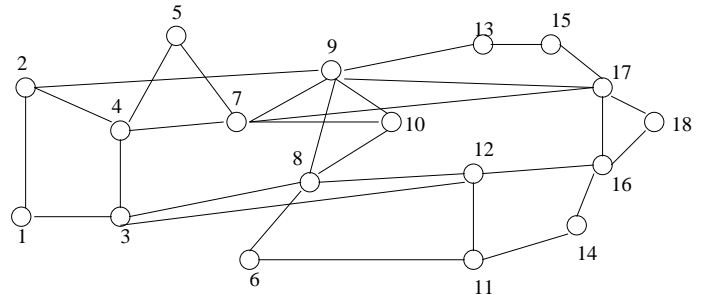


Fig. 1. Isp topology.

such inferences to select routes. We compare the performance of such a route selection scheme with mechanisms that use inaccurate state information, and argue that if the network operates in a “typical” regime, one can do relatively well using this “inference-based” approach. Furthermore, in order to identify the best inference strategy, we construct several alternative schemes that “adaptively” maintain link states via an exponentially weighted averaging mechanism. In particular, the performance advantages of an *end-to-end* strategy over its *link-by-link* counterpart are identified.

To further investigate issues related to uncertainty in routing we consider a “stripped-down” model-based framework, in which we explicitly model uncertainty in the network state by a crude link metric distribution. We identify a notion of “gap” among candidate routes between a particular source and destination pair, suggesting the existence of the dominant routes. These would in turn typically be the most likely routes to satisfy the user’s requirements. This conforms to the results in the Internet routing study by Paxson[4] where end-to-end routing measurement indicates the “persistence” of the routes followed by the data packets. Similarly we study the robustness of the routing decisions to variations in user requests or traffic characteristics. Based on our model we explore the notion of “critical point”, which suggests the fashion in which the variations in user requests might lead to different routing decisions. We deem these notions useful in that they provide a quantitative measurement of how robust the routing process is, and, as discussed in the sequel, provide good insights to design routing and/or route caching mechanisms for future networks.

II. TWO EXPERIMENTS: INFERENCE *vs.* DELAYED STATES

To understand the role played by the link states in the routing process, especially when they involve a certain degree of uncertainty, we conduct the following experiment. The setup of the ex-

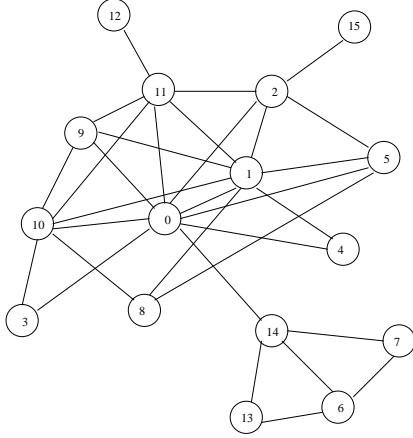


Fig. 2. Net-2 topology.

perimental network is shown in Fig. 1. The connections arrive at a source node according to a Poisson process, and the destination is uniformly chosen from the set of nodes except the source node. The holding time of each connection is modeled by a Pareto distribution. The parameters of the simulation were set as follows: link capacity = 25, connection arrival rate = 12 connections per second, average connection holding time = 2 second, and the bandwidth request of each connection is uniformly distributed between 1 and 3 units. We refer to this as the base case. We increase the traffic load by multiplying the arrival rates of the base case by a sequence of numbers, shown on the x-axis of our plots.

We compare the performance of two routing strategies in terms of the ratio of erroneous decisions, including admitted but blocked connections and falsely rejected connections. The first routing mechanism is an “inference-based” strategy, where no knowledge of the link states is assumed available, except the topology of the network. A group of routes is preselected as the set of candidate routes between a source node and a destination node. In particular, we select all the routes that are either shortest or next to shortest in hop count. During a “warm-up” phase, connections come in and are set up randomly among the group of candidate paths, establishing an initial assessment of the “success probability” of using these paths. The “success probability” of using a particular path is defined as the ratio between the number of successfully routed connections via a path and the number of attempts on using this path. After the “warm-up” phase, the routing mechanism simply selects the path with highest “success probability”, and the source routers dynamically maintain their estimates of the “success probability” via the accumulation of their routing experience, *i.e.*, by keeping track of the number of successes and attempts. The second strategy is the conventional routing approach *with delayed state information*, which makes its decision based on the availability of the resource. We plot in Fig. 3-5 the performance comparison of the two strategies as traffic intensity increases. From graph to graph, we progressively increase the amount of delay involved in link state broadcast.

Fig. 3, 4 and 5 summarize the results of the first experiment. We make several observations as follows:

1) When the delay in link state broadcast is small, *i.e.*, 1 second, (see Fig. 3), both approaches have very similar performance. Taking into account the cost of advertisement to obtain

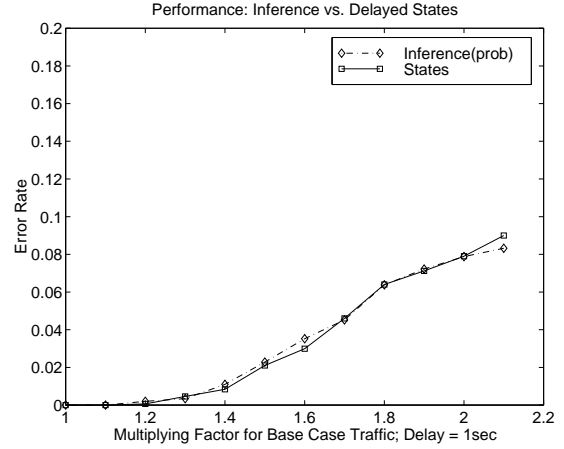


Fig. 3. Inference vs. Delayed state, delay = 1 sec.

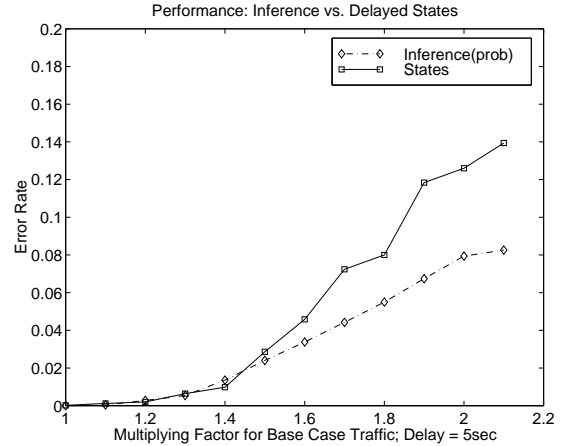


Fig. 4. Inference vs. Delayed state, delay = 5 sec.

fresh state information in terms of CPU usage and extra traffic load, one might argue that inference-based approach is advantageous.

2) When the delay is big, *i.e.*, ≥ 5 second (see Fig. 4), the inference approach outperforms the approach where inaccurate state information is used. Moreover, the difference in performance increases as the traffic intensity increases. Considering the fact that during congestion periods we would like to decrease the overhead associated with state information exchange, an inference-based approach appears to be a superior strategy.

3) From Fig. 4 and 5, we observe that there is a “saturation” point for the degree of degradation in performance when using link-state based strategy, *i.e.*, as the delay increases the performance degrades but it does not degrade indefinitely.

This concludes the first experiment. Let us describe our second experiment. The purpose of this experiment is to demonstrate the performance improvement that can be achieved by inferring *end-to-end* metrics, as opposed to single link metrics. We compare four routing schemes. The first two are the same as the ones in the previous experiment, namely, a “dynamic” routing scheme based on the (delayed) state information and an inference strategy based on success probability. The third one (Inference-link)

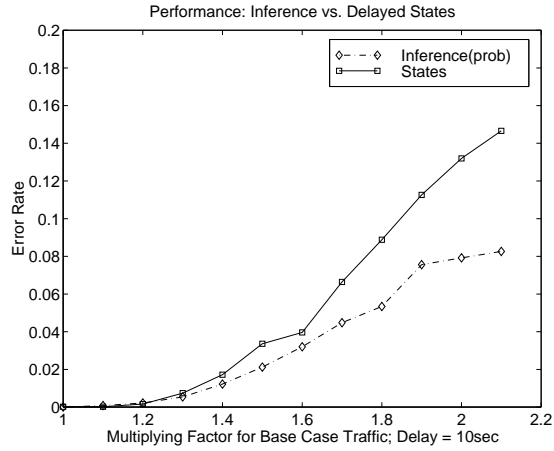


Fig. 5. Inference vs. Delayed state, delay = 10 sec.

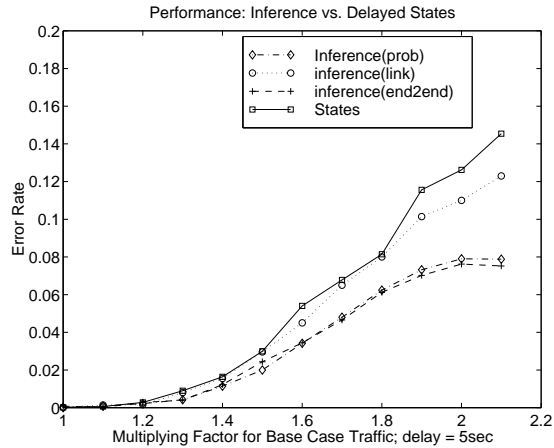


Fig. 6. An experiment(2): Inference vs. Delayed states.

is an “adaptive” scheme based on the estimation of the link available bandwidth. The fourth one (Inference-end2end) is also an “adaptive” scheme. But rather than using the estimation of the link bandwidth, it maintains its estimation of the available bandwidth on the respective routes. In particular, an exponentially weighted averaging mechanism is used in the estimation procedures. When a connection arrival comes in, Inference-end2end selects a path with the greatest estimated path bandwidth (or reject the connection request), while Inference-link computes the path metrics from the estimated link bandwidths and makes a routing decision based on the resulting path metrics.

Fig. 6 demonstrates the result. At least two facts can be observed here:

1) The performance of the inference approach based on success probability is similar to that of the end-to-end available resource case. Intuitively, they are looking at the similar properties of the network thus they yield comparable results.

2) There is a noticeable degradation in performance if we rely on each link itself to maintain the estimated available bandwidth and furthermore use that information to make end-to-end routing decisions. We conjecture that the error in the link estimator will manifest itself in the incorrect end-to-end routing decisions.

This suggests that one should maintain estimation of the route performance rather than link states. However, it is well known that the number of routes grows exponentially as the size of the network increases. This calls for an efficient route state management scheme. In the next few sections we study the robustness of the routing decisions as the link states and/or user requests vary, and verify the obtained insights in a routing/caching setup via simulation.

III. SENSITIVITY ANALYSIS: CONVENTIONAL AND QOS SENSITIVE ROUTING

We consider a weighted directed graph $G(\mathcal{N}, \mathcal{L})$, where \mathcal{N} is a set of nodes and \mathcal{L} denotes the links among the nodes. Each link $l = (i, j) \in \mathcal{L}$ has a random metric D_l , representing uncertain delay/cost information. Let $s, d \in \mathcal{N}$ denote a pair of source and destination nodes, and \mathcal{P}_{sd} denote the set of possible paths between s and d , where a path $P = ((s, i)(i, j)(j, k) \cdots (l, d)) = (l_1, l_2, \cdots, l_n)$ is an appropriately ordered sequence of links. In the case of additive metrics (delay/cost), the goal is to determine a “shortest path” P^* between s and d , i.e., $P^* \in \operatorname{argmin}_{P \in \mathcal{P}_{sd}} D_P$.

Suppose that we have only average metrics when computing the optimal paths. Let $d_l = E[D_l]$ and $d_P = \sum_{l \in P} d_l$. Now one might ask what is the probability that the shortest paths obtained based on the true (random) metrics is among those determined using average metrics, i.e., $\operatorname{Prob}(P^* \in \operatorname{argmin}_{P \in \mathcal{P}_{sd}} d_P)$.

We model the delay/cost metrics as Gaussian random variables and assume that there are two paths between s and d , denoted by P_1 and P_2 . Assume $D_l \sim N(d_l, \sigma_l^2)$, then $D_{P_1} \sim N(d_{P_1}, \sigma_{P_1}^2)$, $D_{P_2} \sim N(d_{P_2}, \sigma_{P_2}^2)$, where $d_{P_i} = \sum_{l \in P_i} d_l$, $\sigma_{P_i}^2 = \sum_{l \in P_i} \sigma_l^2$, $i = 1, 2$.

Suppose $d_{P_1} < d_{P_2}$, then based on the mean values we choose P_1 as the shortest path. What is the probability that this path coincides with the optimal choice based on the realization of the random metrics? The answer is:

$$\begin{aligned} & \operatorname{Prob}(D_{P_1} \leq D_{P_2}) \\ &= \operatorname{Prob}\left(Z \leq \frac{d_{P_2} - d_{P_1}}{\sqrt{\sigma_{P_1}^2 + \sigma_{P_2}^2 - 2\operatorname{Cov}(D_{P_1}, D_{P_2})}}\right), \end{aligned}$$

where $Z \sim N(0, 1)$. Define the “gap” between P_1 and P_2 as

$$\begin{aligned} & \operatorname{gap}(P_1, P_2) \\ &= \frac{\sum_{l \in P_1} d_l - \sum_{k \in P_2} d_k}{\sqrt{\sum_{l \in P_1} \sigma_l^2 + \sum_{k \in P_2} \sigma_k^2 - 2\operatorname{Cov}(D_{P_1}, D_{P_2})}}. \end{aligned}$$

The greater the “gap” is, the more likely we make a good routing decision based on average link metric.

In the case of QoS sensitive routing, the goal is to choose a path that satisfies the quality of service requirements by the users. Since uncertainty is inevitable in both routing metrics and user requests, one might want to search for a path that is most likely to meet the user’s QoS requirements. In [2] the authors tried to solve this problem by explicitly modeling the probability distribution on link metrics. In this section we investigate to what extent we can get away with only mean/variance of the link metrics.

Recall the Gaussian delay/cost model. A routing decision based on the mean metrics, e.g., $d_{P_1} < d_{P_2}$, would select path P_1 and make a “correct” decision with probability $\text{Prob}(D_{P_1} < D_{P_2})$. If the “gap” is relatively big this decision is likely to be correct. Now suppose a user arrives with an end-to-end delay/cost request d . If $d_1 < d_2$, does it necessarily hold that $\text{Prob}(D_{P_1} < d) > \text{Prob}(D_{P_2} < d)$? The answer is no. Consider the following two scenarios:

1) If $d_{P_1} < d < d_{P_2}$, the “average” routing will favor path 1. The “most likely” type strategy will also choose path 1 simply because $\text{Prob}(D_{P_1} < d) > 0.5 > \text{Prob}(D_{P_2} < d)$

2) if $d_{P_1} < d, d_{P_2} < d$, by using only the means we can not select the “most likely” path. Both P_1 and P_2 could be the path “most likely” to satisfy the delay/cost constraint d , depending on the tail of respective distributions. Observe that if path 1 is the one chosen by the “most likely” strategy, we must have $\sigma_{P_2}d_{P_1} + d\sigma_{P_1} < \sigma_{P_1}d_{P_2} + d\sigma_{P_2}$. Hence in this case the link metrics that incorporate the second order statistics might help in identifying the “most likely” path.

Just as link metrics, the user requests may also be considered uncertain. We seek the “critical” value of user request which makes equal the probabilities that the two paths satisfy the requirements, i.e.,

$$\text{Prob}(D_{P_1} < d) = \text{Prob}(D_{P_2} < d)$$

since the $D_{P_1} \sim N(d_1, \sigma_1^2)$, $D_{P_2} \sim N(d_2, \sigma_2^2)$, we have

$$d = \frac{d_1\sigma_2 - d_2\sigma_1}{\sigma_2 - \sigma_1}$$

for $\sigma_1 \neq \sigma_2$. Obviously when $\sigma_1 = \sigma_2$, the “critical” point d does not exist, in which case a routing decision is constant/robust to the variations in the user request. As such we have located a “critical” value of the user request, around which the routing decision is sensitive to the user request. Specifically, a shift in the user request from one side of the “critical point” to the other will change the corresponding routing decision. In other words, the sensitivity is low when the user requests do not go across the “critical point”. Observe that as load condition ($d_{P_1}, d_{P_2}, \sigma_{P_1}, \sigma_{P_2}$) varies, the routing decision regarding the same user request may also change, depending on (once again) the relative position of the user request and critical point d .

IV. PATH CACHING

In the previous section we use a set of crude models to illustrate the effect of “gap” and “critical points” on the routing process. The models are not to be taken as exactly reflecting the network loading condition, but rather intended to provide qualitative insight into the key elements that impact routing decisions. In this section we use simulation in a routing/caching[5] environment to solidify these insights in a realistic application.

A. Simulation setup

The network topology upon which we run the simulation is shown in Fig. 2. We consider QoS routing to meet end-to-end delay requirements in this simulation. The routing metrics are the available bandwidth on respective links and we use the Guaranteed Service model[6] to infer bandwidth requirements from

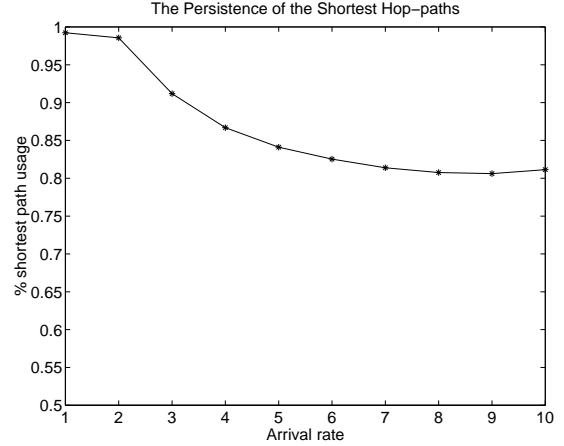


Fig. 7. The existence of gap.

user’s end-to-end delay requirement. The connection requests arrive according to a Poisson process and we assume their holding times are exponentially distributed. The traffic matrix is summarized in [1]. Two types of delay requests come in, which we term “stringent requests” and “relaxed requests”. The “stringent requests” refer to an end-to-end delay tolerance of 15 msec. The “relaxed requests” refer to an end-to-end delay tolerance of 30 msec. The routing algorithm iterates in the increasing order of the hop count and tries to find the first path capable of satisfying the delay requirements.

B. Key observations

1) “Gap”: An interesting observation is made regarding how long a route will remain effective before it is replaced by a better performing route. Here we measure the percentage of admitted connections from 11 to 3 that are routed by the shortest hop paths 11 – 10 – 3 and 11 – 0 – 3. As the traffic intensity changes, the use of the two shortest routes remain consistently above 80%. The point is that among the theoretically large number of routes between any source-destination pair just a few are likely to be used. See Fig. 7.

2) “Critical point”: In light of the previous observation, the shortest hop paths are predominantly used in the connection setup. We would like to understand the preference among the shortest paths as a function of the link state dynamics. In Fig. 8 and Fig. 9, we plot the percentage usage of the path 11 – 10 – 3 for the admitted “relaxed” and “stringent” connections. Observe that for both types of connections, the preference for path 11 – 10 – 3 increases as the traffic load intensifies. Moreover, notice that for the “stringent” connections, the preference for path 11 – 10 – 3 shifts from 42% (unfavorable) to 67% (favorable), while for the “relaxed” connections, the preference for the same path remains above 50%. This suggests the importance of managing separate preferred routes for different types of connection requests between same pair of source and destination nodes.

C. The Performance of the Caching Policy

The observations made previously regarding the impact of the “gap” and “critical point” suggest the feasibility of a good route caching scheme. To implement such a scheme, we maintain a

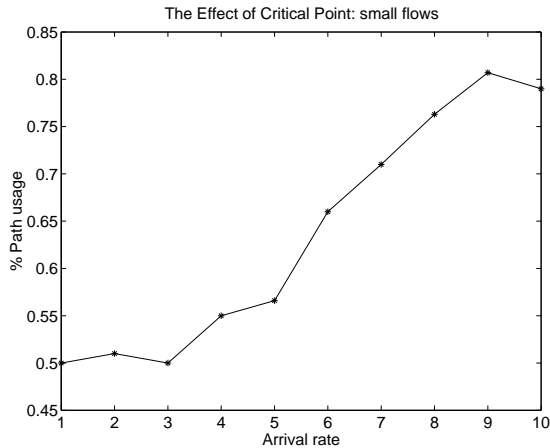


Fig. 8. Path usage: relaxed connections.

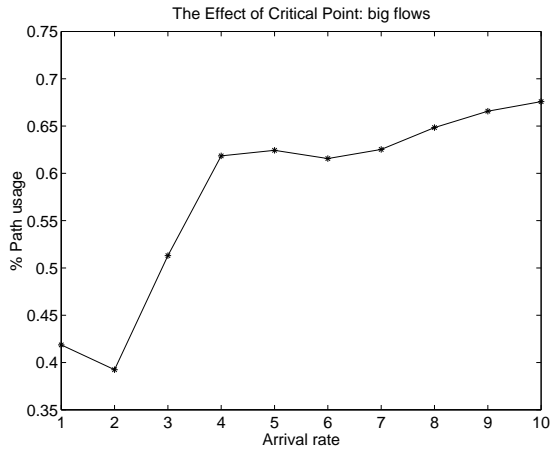


Fig. 9. Path usage: stringent connections.

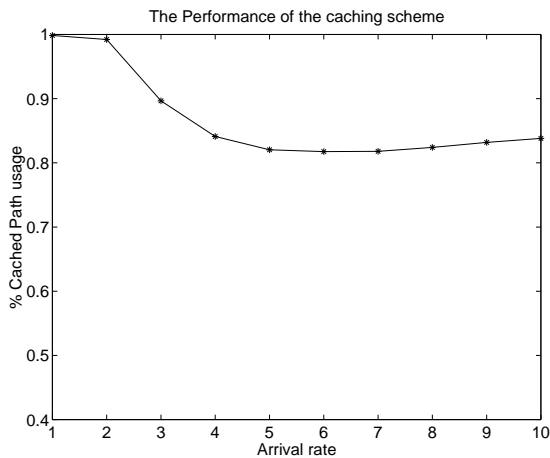


Fig. 10. Performance of Caching.

route cache, each connection arrival is first assigned a route taken from the cache, if an entry corresponding to the user request exists. The source router then attempts to establish the connection by checking the current link states along the chosen route. If the check passes, the connection is set up. If the cached route is not successfully established, we recompute the best path and try to set up the connection in a similar fashion. If the newly computed path is successfully set up, it is cached and replaces the old cached entry. We evaluate this caching scheme using the performance measure defined as the ratio between the number of the successful cached path establishments and the number of the total successful connection establishments. See Fig. 10. We conclude that the savings in path computation is very significant (more than 80%) and the simple rule we utilize is able to track the changing preference in path priority that is shown in Fig. 8 and 9. In short, this simple seek-and-replace cache/routing scheme is aware of “gap” and “critical point”, hence provides an effective way to steer towards optimal route reuse strategy.

V. CONCLUSION

In summary we believe this work sheds some new light on the characteristics and approaches that might be used to design routing mechanisms for broadband networks. We explored issues related to the delay in link-state-based routing algorithms and proposed an inference-based routing scheme. The benefit of the inference strategies is evidenced through experiments. Moreover, the merit of the end-to-end inference method is established. We analyzed a set of simple models of link metrics and identified the notion of “gap” and “critical point”. Their significance in evaluating the robustness of routing decisions in an uncertain environment is studied. We examined the significance of these notions in a routing/caching scenario and identified corresponding properties in a realistic configuration.

REFERENCES

- [1] A.G. Greenberg and R. Srikant. Computational techniques for accurate performance evaluation of multirate multihop communication networks. *IEEE/ACM Trans. Networking*, 5(2):266–277, 1997.
- [2] Roch Guérin and Ariel Orda. QoS-based routing in networks with inaccurate information: Theory and algorithms. In *Proc. IEEE Infocom*, volume 1, pages 75–83, 1997.
- [3] S. Nelakuditi, Z. Zhang, and R.P. Tsang. Adaptive proportional routing: A localized QoS routing approach. In *Proc. IEEE Infocom*, 2000.
- [4] V. Paxson. End to end routing behavior in the internet. *IEEE Trans. Networking*, 5(5):605–615, May 1996.
- [5] M. Peyravian and A. Kshemkalyani. Network path caching: Issues, algorithms and a simulation study. *Computer Communications*, 20:605–614, 1997.
- [6] S. Shenker, C. Partridge, and R. Guerin. Specification of guaranteed quality of service. *IETF Internet Draft, draft-ietf-intserv-guaranteed-svc-07.txt*, Feb. 1997.