Injecting Inter-Autonomous system routes into Intra-Autonomous system routing. A Performance Analysis

Yakov Rekhter
T.J. Watson Research Center
IBM Corp.

Bilal Chinoy*
San Diego Supercomputer Center
La Jolla, CA 92186-9784

1 Introduction

The current TCP/IP Internet may be modeled as an arbitrary interconnection of autonomous systems [1] [Rekhter 91b]. An AS is defined to be a collection of internetwork routers managed and administered by a single authority or organization. Typically an AS uses a single routing protocol within it's boundaries to generate and propagate routing information, though is not uncommon to have a single AS use multiple routing protocols. Based on this approach, routing in the Internet is partitioned into intra-autonomous system routing (intra-AS routing) and inter-autonomous system routing (inter-AS routing). If both a source and a destination belong to the same AS, then routing between the source and the destination is completely realized by employing intra-AS routing procedures used by the AS. However, if a source and a destination belong to different ASs, then routing between the source and the destination cannot be realized either completely within the inter-AS routing or completely within the intra-AS routing, but involves both the intra and inter-AS routing. That, in turn, requires a certain degree of interaction between the intra and inter-AS routing procedures.

In [Rekhter 91a] we analyzed one aspect of such an interaction - the exchange of the reachability information between the inter and the intra-AS routing and presented several schemes that differ from each other with respect to the amount of information exchanged.

At one extreme, all the reachability information derived from the inter-AS routing is injected into the intra-AS routing. One advantage of such a scheme is the ability of any router within an AS to choose the locally optimal exit point from this AS [2]. The drawback of this scheme is that it requires all routers within an AS to have enough memory and processing power to handle all the reachability information derived from both the intra and the inter-AS routing. We will refer to such a scheme as the *CRI* scheme (*C*omplete *R*eachability *I*njection).

At the other end of the spectrum, none of the reachability information derived from the inter-AS routing is injected into the intra-AS routing. This scheme has clear advantages over the *CRI* scheme with respect to the memory and processing power required on the routers (since only routers that participate in inter-AS routing have to store and process inter-AS reachability information), but has disadvantages as well, such as the encapsulation of all inter-AS system traffic. Amongst the most serious problems with encapsulation is the potential fragmentation of packets at the entry router, and the additional processing that is necessary to perform this fragmentation and reassembly. Encapsulation also has problems when dealing with error reporting. For example, if an encapsulated packet causes an ICMP Source Quench message to be generated by the router that is not going to unwrap the packet, this message will be sent to the router that encapsulated the packet, rather than to the host that originated the packet. Other issues such as handling the packet's life time and processing IP options are potentially more involved and thereby inefficient to do when the encapsulation scheme is used. We will refer to such a scheme as the *ERI* scheme (*E*mpty *R*eachability *I*njection).

A compromise between these two schemes may be realized by a technique that injects only partial reachability

*previously at Merit, Inc.
[1] For the sake of brevity we will use "AS" instead of "autonomous system" for the rest of this document.
[2] The locally optimal exit point does not guarantee global path optimality.

information derived from the inter-AS routing into the intra-AS routing. This scheme allows for the configuration of both the amount of the inter-AS reachability information that has to be present in the intra-AS routing and the amount of required encapsulation. We will refer to such a scheme as the *PRI* scheme (*P*artial *R*eachability *I*njection) [3].

In this paper we attempt to evaluate the performance of the *PRI* scheme. This performance evaluation is based on traffic data obtained from the NSFNET backbone. The remainder of this paper is organized as follows; Section 2 describes the *PRI* scheme. Section 3 describes the performance metrics that are used to measure performance of the *PRI* scheme. Section 4 describes the measurement environment and sampling methodology. Section 5 analyses the performance of the *PRI* scheme based on collected data. Section 6 concludes the paper and discusses issues for future work.

2 *PRI* - injecting a subset of the inter-AS reachability information into the intra-AS routing.

If the inter-AS traffic is non-uniform with respect to different destinations, then we may talk about the **concentration** property of the traffic. The term concentration characterizes a property of the traffic where a relatively large fraction of traffic tends to be sent to a relatively small subset of all the destinations reachable via the inter-AS routing. The *PRI* scheme takes advantage of this concentration property.

Details of this scheme are described in [Rekhter 91a]. Functionally it consists of two components: a route placement component, and a route replacement component. The route placement component selects a particular route from the inter-AS routing database, and then injects reachability information associated with this route into the intra-AS routing. In order to maintain an upper bound on the total amount of the inter-AS reachability information that is present at any given moment in the intra-AS routing, this injection is preceded by the replacement of previously injected reachability information. Selection of candidate routes for replacement is done by the route replacement component.

2.1 Route placement / replacement components.

To simplify the description of algorithms employed by the route placement and replacement components, we introduce the concept of route productivity. An inter-AS route's productivity is defined as the ratio of the number of packets that either originated within an AS or transited through the AS and that can be forwarded using this route, to the total number of packets that either originated within the AS and were destined outside the AS, or transited through the AS, for a fixed time interval [4]. We will refer to the time interval that is used to compute route productivity as the evaluation interval, and denote it by T_e [5].

If we consider the set of all inter-AS routes as a population, then the distribution of route productivity over this population can be characterized by either the route productivity relative frequency distribution function, or by the route productivity cumulative frequency distribution function. The route productivity relative frequency distribution f_p is defined by grouping routes according to their productivity. For a given productivity p, the value of $f_p(p)$ is defined as the ratio of the number of routes with productivity equal to p, to the total number of routes R_{total} that are present in the inter-AS routing. The route productivity relative cumulative frequency distribution F_p is defined as the ratio of the number of routes that have productivity less than or equal to p, to the total number of routes.

That is,

$$F_p(p) = \sum_{m \leq p} f_p(m) \tag{1}$$

Since productivity associated with an individual route may depend on the duration of the evaluation interval, definition of both f_p and F_p tacitly assumes that the same evaluation interval is used to compute productivity of the

[3] Note that this scheme has no implications on the dynamics (up/down status) of individual routes in the injected set.
[4] By definition, the productivity of an individual route is bounded between 0 and 1.
[5] Since, in general, traffic that can be routed via a particular route varies with time, the productivity of an individual route depends upon other factors and on the duration of the evaluation interval.

individual routes.

Selection of inter-AS reachability information for injection into the intra-AS is based on the productivity of the routes associated with that information. Route placement and route replacement components constantly evaluate the productivity of inter-AS routes, and periodically (at the end of each evaluation interval) adjust the set of the inter-AS reachability information present in the intra-AS routing tables to maintain only the most productive routes.

If one assumes that it should be possible to correctly predict the productivity of individual routes, then it should be equally possible to come up with an optimal algorithm (*OPT*) that works as follows: At the end of each evaluation interval the *OPT* algorithm injects reachability information associated with the routes that will have the highest productivity during the next evaluation interval. The *OPT* algorithm guarantees that only the routes that are the most productive within a current evaluation interval have their reachability information present in the intra-AS routing.

However, since the *OPT* algorithm requires a priori information that can be obtained only by looking into the future, it cannot be implemented in a real system. It is then clear that a route placement algorithm needs some heuristic that is based on the past, rather than on the future productivity of individual routes. In this paper we investigate the performance of one such heuristic that selects the most productive routes, as determined over the current evaluation interval, for injection at the end of the current evaluation interval. That is, the reachability information associated with the inter-AS routes that are the most productive during the current evaluation interval will be present in the intra-AS routing for the duration of the next evaluation interval. We denote the algorithm that employs this heuristic by *HEURISTIC* [6].

3 Performance criteria for the *PRI* scheme.

In order to evaluate the performance of the *PRI* scheme we select performance metrics. These metrics will reflect the two major objectives of the *PRI* scheme. The first objective is to constrain the amount of inter-AS reachability information that has to be present in the intra-AS routing. The second objective is to constrain the amount of encapsulated traffic (that may be caused by absence of certain inter-AS reachability information in the intra-AS routing).

The performance of the *PRI* scheme with respect to constraining the amount of the inter-AS reachability information is measured by the ratio R_e of the inter-autonomous system reachability information present in the intra-AS routing to the total amount of reachability information present in the inter-AS routing. Schemes with higher values of R_e would clearly require more router memory and processing power, and thus are less desirable. The *CRI* scheme has R_e of 1, while the *ERI* scheme has R_e of 0. The *PRI* scheme has R_e between 0 and 1.

The performance of the *PRI* scheme with respect to constraining the amount of encapsulated traffic is measured by the ratio P_e of the total number of packets that have to be encapsulated by any router within an AS within a given evaluation interval to the total number of packets that have emanated from the AS [7] over the same evaluation interval. Lower values of P_e are preferred. The *CRI* has P_e of 0, while the *ERI* scheme has P_e of 1. The *PRI* scheme has a P_e between 0 and 1.

Clearly P_e depends on R_e: P_e tends to 0 as R_e tends to 1. Injecting all inter-AS routes with productivity greater than p into the intra-AS routing results in $R_e(p) = (1 - F_p(p))$ and

$$P_e(p) = R_{total} \sum_{m \leq p} m f_p(m) \tag{2}$$

(where R_{total} is the total number of inter-AS routes present in the inter-AS routing during the evaluation interval).

Since the content of the inter-AS reachability information that is present in the intra-AS routing may change over time, changes to that information may consume both bandwidth and processing power on the routers. One way to quantify resources consumed as a result of these changes is to measure the total number of the inter-AS routes

[6] The replacement algorithm employed by the HEURISTIC is quite similar to the Least Frequently Used (LFU) algorithm for virtual memory page replacement. With LFU each page keeps a count of the total number of references to the page. The page with the smallest count is replaced.

[7] These packets may either originate within an AS or may just transit the AS.

whose reachability information has to be injected/withdrawn at the end of each evaluation interval, normalized to the length of the evaluation interval. We will denote this metric by B_e. If the productivity of inter-AS routes is time invariant, then B_e is bounded from below by the volatility of the inter-AS routing information. Since inter-AS reachability information cannot be injected/withdrawn from the intra-AS routing more often than the convergence time of the intra-AS routing, B_e is bounded from above by the convergence speed of the intra-AS routing protocol employed by a particular AS.

Clearly B_e depends on R_e: as R_e tends to 1, B_e tends to its lower bound; and conversely as R_e tends to 0, B_e tends to its upper bound [8].

The performance of the *PRI* scheme is affected by several parameters. One of these parameters is the offered workload, which is defined as the total amount of inter-AS traffic emanated from an AS [9]. Another parameter that can be varied to control the performance of the scheme is the duration of the evaluation interval.

4 Measurement environment and sampling methodology.

In order to evaluate the performance of the *PRI* scheme, we recorded destination address references of traffic coming into the NSFNET backbone from the attached regional networks. The recorded references were used to compute the performance metrics for the *PRI* scheme.

4.1 NSFNET Backbone.

The NSFNET backbone is a national wide area IP network interconnecting regional networks and national supercomputing centers. This network is managed by Merit Inc., a consortium of universities in the state of Michigan. The National Science Foundation awarded the joint project to Merit in November 1987 and the network was operational by July of 1988. The NSFNET backbone serves as the primary transit network for most of the IP internet traffic and is one of the largest component networks in the Internet.

4.1.1 Architecture.

The NSFNET backbone consists of 14 packet switching nodes, called NSSs (Nodal Switching System), interconnected by 21 T1 links. Each NSS internally consists of several IBM RT/PC machines, interconnected by the Intra-NSS dual 4 Mb/s token rings. Some of these RT/PCs are dedicated to packet switching (designated as PSPs, or packet switching processors) and one RT/PC which acts as the routing control processor (RCP).

At each NSS, mid-level networks attach to the backbone typically via a shared medium, such as an Ethernet. One or more RT/PCs in each NSS are the exterior PSP (or E-PSP) and are the interface between the attached mid-level networks and the NSFNET backbone. All inbound traffic from the attached mid-level networks and outbound traffic to the attached mid-level networks is switched through these E-PSP(s).

From the routing point of view the NSFNET backbone acts as a single AS. It uses either EGP-2 [Mills 84] or BGP [Lougheed 91] to exchange routing information with mid-level networks. Each mid-level network consists of one or more ASs.

4.1.2 Trace collection.

Each NSS has a RT/PC dedicated to gathering packet statistics. This machine listens in promiscuous mode on the intra-NSS token rings, and characterizes each packet that traverses the intra-NSS token rings based on a programmable set of statistical definitions (see Fig. 1). Packets with certain characteristics (with respect to the set of statistical definitions) are then logged to a disk. The trace gathering processor in each NSS is used to record packets

[8] In practice, B_e is bounded from above by the maximum update frequency allowed by a particular intra-AS routing protocol. This bound is usually tighter than the one imposed by the convergence speed of intra-AS routing.

[9] This traffic is a combination of traffic that originated within an autonomous system, but is headed outside the AS, and traffic that just transits the AS.

coming into the backbone from the attached mid-level networks [10]. The recorded packets are then grouped together based on the network part of their destination addresses. From this trace, destination network reference strings are generated. This reference string data is the starting point of all subsequent analysis.

4.2 Methodology.

For the purposes of this study, traffic into the backbone from all ingress points was collected and destination network reference strings were generated. During the period of trace collection there were approximately 2000 networks (outside of the NSFNET backbone itself) reachable via the backbone. Since at the level of inter-AS routing each network has associated with it a unique inter-AS route, there were approximately 2000 inter-AS routes in the NSFNET backbone routing tables.

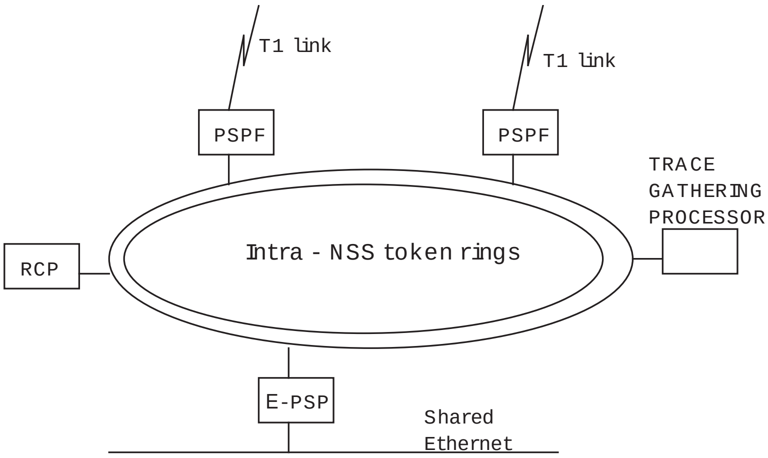

Figure 1: NSFNET NSS with trace gathering processor

Destination network reference strings were collected with a time granularity of 1 minute. The total trace period was 48 hours. The data collected over the total trace period was broken into six segments, each segment containing data for eight hours. The segments were further characterized by containing either peak or offpeak offered workload [11]. Each segment was further partitioned into consecutive fragments, where each fragment represented traffic for the duration of the evaluation interval. For the duration of the evaluation interval parameter we use several discrete values ranging from 1 to 120 minutes. The productivity of individual inter-AS routes were computed from the collected fragments for different values of the evaluation interval. Performance metrics (such as P_e, R_e and B_e) were computed over individual fragments, and then averaged over multiple fragments with similar parameters (i.e. offered workload, length of the evaluation interval).

5 Analysis of the measured data.

5.1 Concentration.

To verify the concentration property of the inter-AS traffic we plotted the route productivity relative frequency distribution function (as defined in section 2.1) of inter-AS routes for two different evaluation intervals (1 minute and 60 minutes) and both peak and offpeak offered workloads (see Fig. 2).

The curves show that distribution of productivity is highly non-uniform. A relatively large fraction of all inter-AS routes has relatively small productivity, while a relatively small fraction of all inter-AS routes has relatively high productivity. The non-uniformity is more noticeable for offpeak than for the peak offered workload.

[10] It should be noted that the trace gathering processor is not capable of keeping up with the volume of traffic on the intra-NSS token rings. Packet loss ranged from 0 to 5 percent of all packets seen on the token rings.

[11] The total offered workload during the off-peak period was approximately 65% of the peak offered workload.

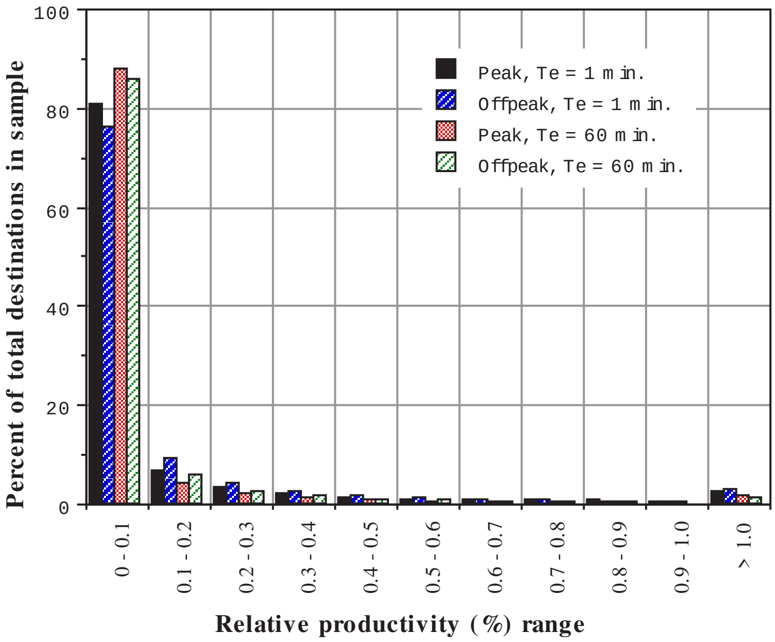

Figure 2: Histogram of route productivity

5.2 Encapsulation versus amount of the inter-AS reachability information injected into the intra-AS routing.

To study how encapsulation varies with the amount of inter-AS reachability information injected into the intra-AS routing, we plotted the percentage of encapsulated traffic P_e as a function of the percentage of inter-AS reachability information injected in the intra-AS routing R_e [12] (see Fig. 3). For this study we used the *OPT* algorithm with two different values of the evaluation interval (1 minute and 60 minutes), and both peak and offpeak offered workload.

Experimental data shows that as the percentage of the injected inter-AS reachability information increases, the percentage of encapsulated traffic monotonically decreases - P_e is inversely proportional to R_e. This behavior is a direct consequence of the observed high non-uniformity of productivity associated with individual inter-AS routes.

Observe that incremental gains in P_e falls off rather sharply after relatively small values of R_e. For example, injecting just 5% of the inter-autonomous system routes allows the forwarding of at least 25% of all inter-autonomous system traffic without resorting to encapsulation, whereas to keep encapsulation below 5%, at least 20% of the inter-AS routes have to be injected into the intra-AS routing.

The relationship between P_e and R_e exhibits similar behavior for both peak and offpeak offered workload, as well as for two different evaluation intervals. This is consistent with our previous observation that non-uniformity in productivity is present in both peak and offpeak offered workload.

The curves shown in Figure 3 indicate that if R_e is below the certain bound, then the *PRI* scheme with the *OPT* algorithm shows better performance with increase in offered workload. However, as R_e increases, the effect of the workload on performance becomes less noticeable. For example, with an evaluation interval of 1 minute and $R_e = 25\%$, the *OPT* algorithm shows the same performance for both peak and offpeak offered workload. We hypothesize that as R_e increases above the bound (i.e. 25%), better performance will be achieved with decrease in the offered workload.

[12] P_e was computed by applying Equation (2).

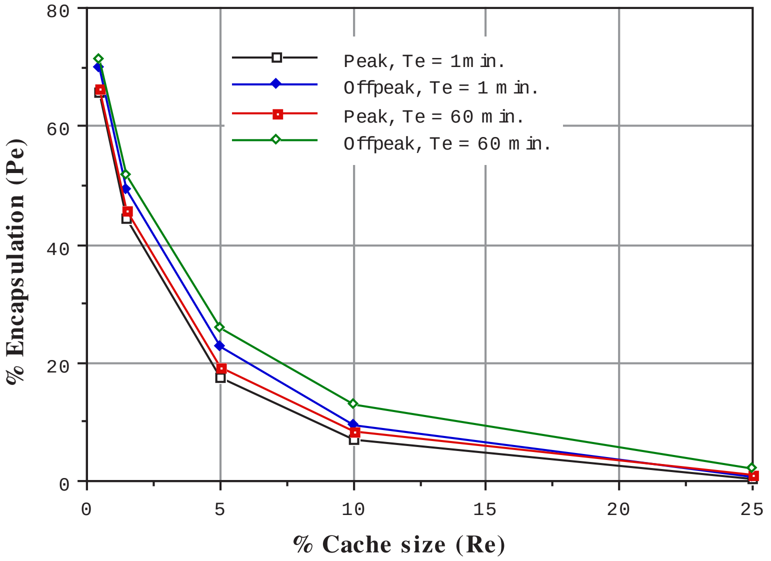

Fig. 3

Figure 3: Encapsulation v/s Cache size: Optimal Algorithm

5.3 Encapsulation versus Evaluation interval.

To study the effect of the evaluation interval (T_e) on the amount of encapsulation (P_e), we plotted P_e for discrete values of T_e ranging from 1 minute to 120 minutes. This was done for various values of R_e, for the peak and offpeak workloads, as well as for both the *OPT* and the *HEURISTIC* algorithms (see Fig. 4).

The plotted curves show that performance of the *PRI* scheme slowly degrades as the length of the evaluation interval increases. This can be explained by the slight decrease in concentration as the length of the evaluation interval increases. The degradation is more noticeable for the *HEURISTIC* algorithm than for the *OPT* algorithm. This can be attributed to the increase in inaccuracy in predicting productivity as the the length of the evaluation interval increases.

Observe that in absolute terms the degradation in performance with an increase of the evaluation interval is not significant. For instance, as the length of the evaluation interval increases from 1 minute to 120 minutes, the additional encapsulation due to this increase in the worst case does not exceed 7%. However, in relative terms, the degradation in performance is more noticeable when a large amount of the inter-AS reachability information is injected into the intra-AS routing.

5.4 *OPT* versus *HEURISTIC* algorithm.

To evaluate the performance of the *HEURISTIC* algorithm versus the *OPT* algorithm, we plotted the difference between the percentage of encapsulated traffic that can be achieved by the *HEURISTIC* algorithm and by the *OPT* algorithm, normalized to the percentage of encapsulated traffic that can be achieved by the *HEURISTIC* algorithm.

That is,

$$\Delta = \frac{P_e^{HEU} - P_e^{OPT}}{P_e^{HEU}} \tag{3}$$

as a function of the percentage of inter-AS reachability system injected into the intra AS routing (R_e) (see Fig. 5). These curves were plotted for both peak and offpeak workload, as well as for several values of the evaluation interval.

The data shows that performance of the *HEURISTIC* algorithm as compared with the *OPT* algorithm monotoni-

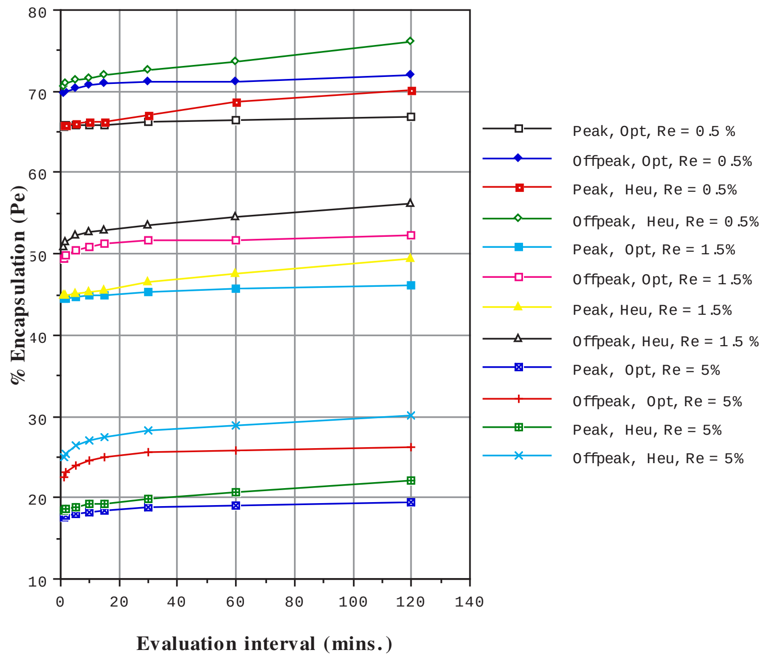

Figure 4: Encapsulation v/s Evaluation Interval

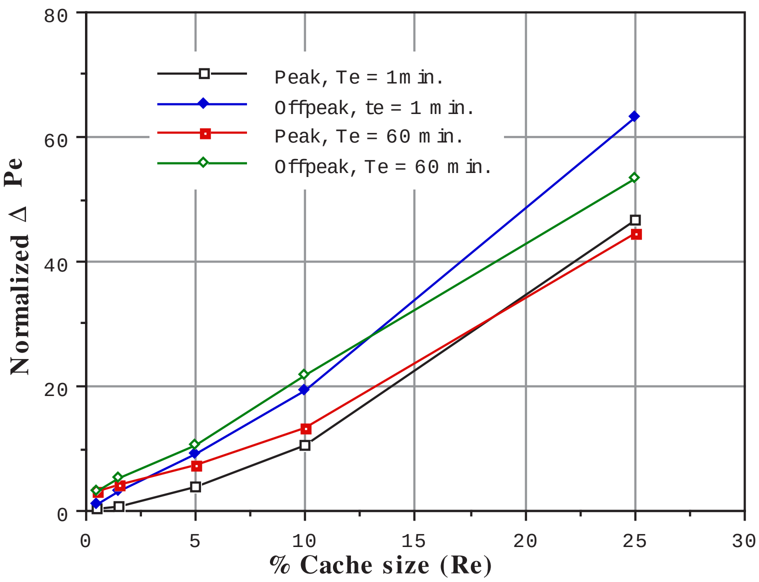

Figure 5: Normalized Δ P_e v/s cache size

cally degrades with increase in R_e [13]. To explain this phenomenon, observe that the ability to correlate the past productivity of the individual routes with their future productivity is crucial to the performance of the *HEURISTIC* algorithm. However, as R_e increases, a larger fraction of the inter-AS reachability information injected into the intra-AS routing corresponds to the inter-AS routes with small productivity. Since the productivity of these routes is more volatile and less correlated across samples, correlation between the past and the future productivity of these routes may be rather weak, thus resulting in the observed behavior.

The performance degradation with increase in R_e is more noticeable for the offpeak offered workload, than for the peak offered workload. This may be explained by the less accurate correlation between the past and the future productivity of individual routes due to weaker concentration of the offpeak offered workload, as compared to the peak offered workload.

5.5 Volatility of the injected information.

We also studied the volatility of the inter-AS reachability information that should be injected into the intra-AS routing by the *PRI* scheme (see Fig. 6).

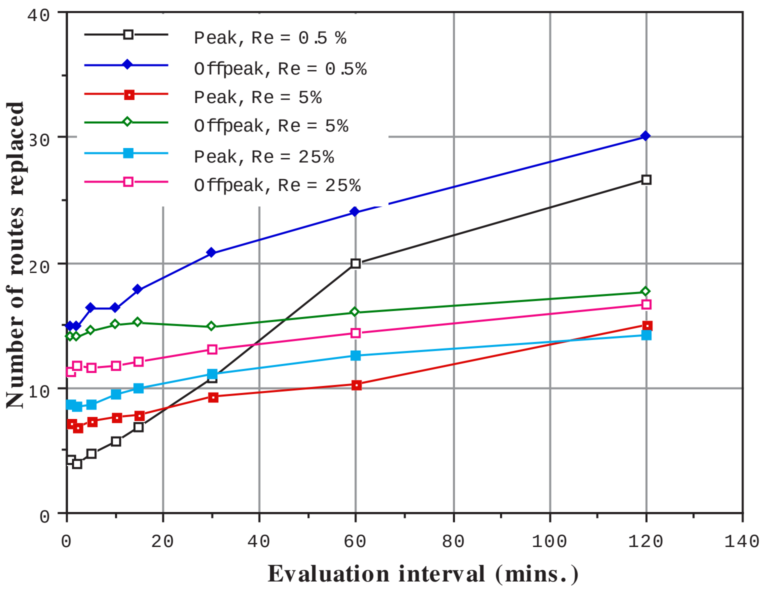

Figure 6: Number of routes replaces v/s Evaluation Interval: Heuristic Algorithm

The volatility was measured by fixing the fraction of the inter-AS reachability information to be injected into the intra AS routing to R_e, computing the set of R_e most productive routes for two consecutive evaluation intervals, and then determining the changes in the set between these two intervals. For this study we used only the *HEURISTIC* algorithm. This volatility was studied for different values of R_e, length of the evaluation interval and offered workload.

Figure 6 shows that in absolute terms the amount of change in the inter-AS reachability information increases as the length of the evaluation interval increases. That is not surprising, since the productivity of individual routes exhibits greater fluctuation with an increase in the evaluation interval.

While in absolute terms the amount of change in the inter-AS reachability information increases as the evaluation interval increases, the bandwidth B_e consumed by these changes actually decreases rather rapidly with the increase in the evaluation interval (see Fig. 7a and 7b).

That is, the number of routes that are injected into the intra-AS routing, normalized by the length of the evaluation

[13] It should be pointed out that while the relative performance of the *PRI* scheme with the *HEURISTIC* algorithm monotonically degrades with increase in R_e, the absolute performance of the *PRI* scheme with either the *OPT* or the *HEURISTIC* algorithm monotonically improves with R_e.

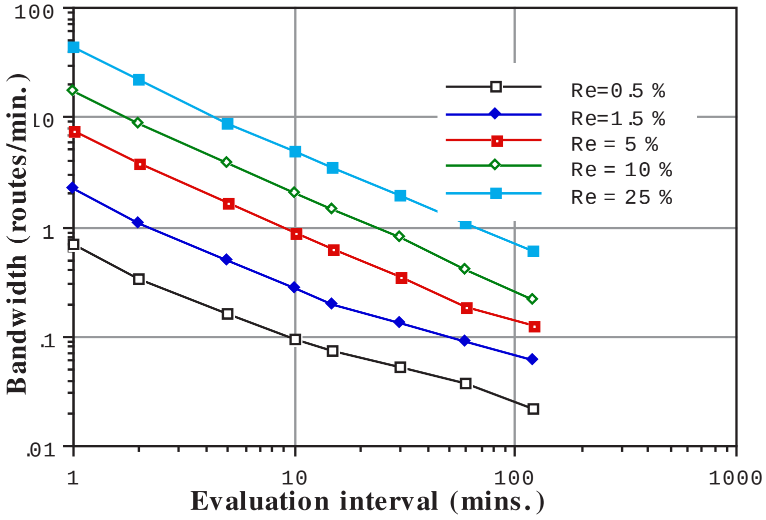

Figure 7: Bandwidth (routes/min.) consumed in route replacement: Peak Workload

interval, shows steady decline with increase in the evaluation interval. Also observe that the bandwidth consumed by the changes is not significantly affected by the choice of a particular algorithm (*OPT* versus *HEURISTIC*), or by the offered workload.

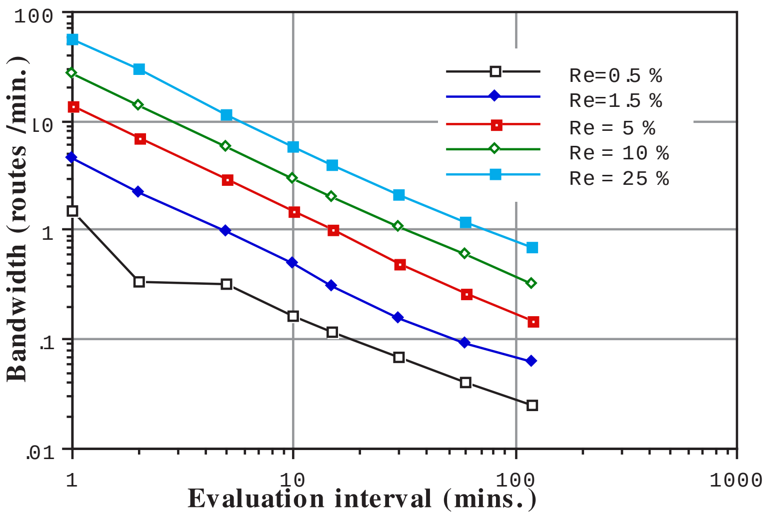

Figure 8: Bandwidth (routes/min.) consumed in route replacement: Offpeak Workload

6 Conclusions.

The study clearly demonstrates the feasibility of the *PRI* scheme. For instance, injecting just 10% of the total inter-AS reachability information (about 200 entries) into the intra-AS routing permits the forwarding of at least 85% of the transit traffic without resorting to encapsulation.

Since the performance of the *PRI* scheme is not very sensitive to the length of the evaluation interval, longer intervals may be used. Hence, it should be possible to use quasi- static mechanisms to control the injection of the inter-AS reachability information into the intra-AS routing. This is especially true if the set of injected routes is composed of only the high productivity routes, since in this case the contents of this set remain relatively unchanged over long time intervals. An example of such a mechanism may be as simple as a configuration file that lists the set of the reachability information that, if present in the inter-AS routing, should be injected into the intra-AS routing.

It should be pointed out, that in order to get noticeable benefits, the *PRI* scheme should not try to arbitrarily limit the required encapsulation. Based on the observed data we may conclude that the most beneficial use of the *PRI* scheme for a large transit AS is when the encapsulated traffic constitutes about 10% of the total traffic.

While this paper focuses on the exploitation of the non-uniformity in the intra-AS traffic by the *PRI* scheme, there may be other applications that can take advantage of the observed non-uniformity. Also, this paper does not attempt to cover all of the implementation details that may be encountered.

7 Acknowledgements.

We would like to express our deep acknowledgements to Hans-Werner Braun (San Diego Supercomputing Center) , Elise Gerich (Merit / NSFNET), Jessica Yu (Merit / NSFNET), Tony Li (cisco systems), Dave Oran (DEC) and Radia Perlman (DEC) for their constructive comments and suggestions.

References

[Braun 89] Braun, H-W., "The NSFNET Routing Architecture", RFC 1093, February 1989.

[Denning 68] Denning, P.,J., "The Measurement of Locality and the Behavior of Programs", The Computer Journal, Vol. 27, No.3, 1984, pp. 238-245.

[Lougheed 91] Lougheed, K., Rekhter, Y., "A Border Gateway Protocol 3 (BGP-3)", RFC 1267, October 1991.

[Mills 84] Mills, D., "Exterior Gateway Protocol Formal Specification", RFC 904, April 1984.

[Rekhter 91a] Rekhter, Y., "Constructing intra-AS path segments for an inter-AS path", Computer Communications Review, Vol 21, No. 1, January 1991.

[Rekhter 91b] Rekhter, Y., Gross, P., "Application of the Border Gateway Protocol in the Internet", RFC 1268, October 1991.