Best-Path vs. Multi-Path Overlay Routing

David G. Andersen, Alex C. Snoeren†, and Hari Balakrishnan

MIT Laboratory for Computer Science
{dga,hari}@lcs.mit.edu

†University of California, San Diego
snoeren@cs.ucsd.edu

Abstract

Time-varying congestion on Internet paths and failures due to software, hardware, and configuration errors often disrupt packet delivery on the Internet. Many aproaches to avoiding these problems use multiple paths between two network locations. These approaches rely on a path-independence assumption in order to work well; i.e., they work best when the problems on different paths between two locations are uncorrelated in time.

This paper examines the extent to which this assumption holds on the Internet by analyzing 14 days of data collected from 30 nodes in the RON testbed. We examine two problems that manifest themselves—congestion-triggered loss and path failures—and find that the chances of losing two packets between the same hosts is nearly as high when those packets are sent through an intermediate node (60%) as when they are sent back-to-back on the same path (70%). In so doing, we also compare two different ways of taking advantage of path redundancy proposed in the literature: mesh routing based on packet replication, and reactive routing based on adaptive path selection.

Categories and Subject Descriptors

C.2.5 [**Computer-Communication Networks**]: Local and Wide-Area Networks—*Internet*

General Terms

Measurement

Keywords

Networking, Measurement, Multi-Path Routing, Overlay Networks

1. Introduction

The routing infrastructure in the Internet does not attempt to provide loss-free packet delivery between end points. End-to-end transfers observe packet losses due to several reasons, including network congestion, path failures, and routing anomalies. As a result, applications and transport protocols have to cope with these packet losses. This is often done using packet retransmissions, coupled with a reduction in sending rate to react to congestion, resulting in degraded throughput and increased latency. Internet paths often experience outages lasting several minutes [1, 18], and end-to-end connections that are in the middle of data transfers usually end up aborting when such outages happen.

Over the past few years, both routing optimizations at the IP layer [24, 29, 32, 33] and overlay networks layered on top of the Internet routing substrate [1, 31, 23] have been proposed as ways to improve the resilience of packet delivery to these problematic conditions. These approaches either probe to find a single best path through the Internet, or send data redundantly along multiple paths.

To work well, these routing techniques require that a fundamental property hold, which is that *losses and failures on different network paths be uncorrelated with each other.* A failure or loss on one path from a source to a destination must not overlap in time with the failure of all other paths from the source to the destination.

Mesh routing is the simplest way to add redundant packets to the data stream by duplicating all of the packets along different paths [31]. In this scheme, the overhead is due to redundant packets, but the scheme does not require additional probing. When its paths are disjoint, mesh routing is resilient to the failure of any proper subset of its component paths. In this paper, we examine the behavior of mesh routing when its packets are sent over an overlay network and examine the degree to which its packets are actually lost independently.

In reactive routing implemented with overlay networks, the overlay nodes constantly probe the $O(N^2)$ paths between them and send packets either on the "direct" Internet path, or forward them via a sequence of other nodes in the overlay when the latter path provides better performance. The overhead in this approach comes from both probes and overlay routing traffic. The probes are required to ensure that when a problem occurs with the current path or when a better path presents itself, traffic is rerouted appropriately to reduce the observed loss rate. Inspired by the approach used in RON [1], we focus on a simple but effective overlay routing method that uses at most one intermediate node in the overlay network to forward packets.

We analyze fourteen days of probes between 30 geographically diverse nodes of the RON testbed. These probes include packets sent back to back via various mechanisms to help determine the degree to which failures and losses on the Internet are correlated. Using this data, we examine the performance of reactive routing and mesh routing, and compare their loss rate and latency reduction to the direct Internet path between pairs of nodes. The testbed grew over time with little selection of node location—results for other topologies will vary.

Our major findings are that:

This research was sponsored by the Defense Advanced Research Projects Agency (DARPA) and the Space and Naval Warfare Systems Center, San Diego, under contract N66001-00-1-8933.

IMC'03, October 27–29, 2003, Miami Beach, Florida, USA.

- The conditional loss probability of back-to-back packets (the probability of losing the second when the first was lost) is high both when sent on the same path (70%) and when sent via different paths (60%).
- The likelihood of multiple paths between a source and a destination simultaneously failing is high, and seems higher in 2003 than in our 2002 data.
- The overall packet loss rate between our hosts is a low 0.42%. Reactive routing reduces this to 0.33%, and mesh routing reduces it to 0.26%. These improvements come primarily from reducing the loss during higher-loss periods of time; during many hours of the day, the Internet is mostly quiescent and loss rates are low. During the worst one-hour period monitored, the average loss rate was over 13%.
- Mesh-based routing and reactive routing appear to exploit different network properties and can be used together for increased benefits.

We survey related work in Section 2. Section 3 presents the design of the simple probe-based overlay routing protocol and replication-based multi-path protocols that we study empirically in Section 4. In Section 5, we examine the implications of our results on the design of improved routing schemes, and we conclude in Section 6 with a summary of our findings.

2. Background and related work

The Internet is a best-effort medium, and its paths often exhibit packet loss. Congested routers and links combine to cause various levels of packet loss. Severe burst losses or outages may be exacerbated by link failures, routing problems, or both. Labovitz *et al.* show that routers may take tens of minutes to stabilize after a fault, and that packet loss is high during this period [18]. They also note that route availability is not perfect, causing sites to be unreachable some fraction of the time [19]. Paxson notes that packets are often subject to routing loops and other pathologies [25].

2.1 Reliable transmission

The traditional way to mask losses in packetized data transfer is to use packet diversity through retransmissions, forward error correction (FEC), or a combination of the two. Retransmissions are appropriate for end-to-end protocols, but adding this functionality at the network level can confound TCP's retransmission timers. Furthermore, not all applications require this functionality, and may not desire its cost in latency and bandwidth.Inspired by such applications, we examine loss-resilient routing strategies that do not dramatically increase end-to-end round-trip latencies.

Hop-by-hop ARQ schemes can reduce the delay for certain topologies [4], but require buffering and network support at intermediate nodes. Many ARQ schemes are tuned for certain loss characteristics, and function poorly over channels outside of their design space. While these schemes benefit links—such as wireless links—with high bit-error rates, they are not universally applicable in the general Internet context. In particular, these schemes do not apply in the case of congestive losses or link failures, the major causes of loss in the wired Internet.

FEC adds redundant information to a stream, allowing the stream to be reconstructed at the receiver even if some of the information is corrupted or missing [15]. FEC is commonly used in wireless systems to protect against bit corruption [22], and more recently in multicast and content distribution systems to protect against packet loss [10, 28]. The latter applications require packet—as opposed to bit—level FEC. We consider packet-level FEC in this paper.

Sending redundant data along the same Internet path is rarely completely effective due to high packet loss correlation. Bolot examined the behavior of packet trains on a single link between INRIA and the University of Maryland in 1992 [7]. He found that the conditional loss probability of back-to-back packets is high when the packets are closely spaced ($\sim$ 8 ms), but returns to the unconditional loss probability when the gap is $\sim$ 500 ms. Similarly, Paxson examined TCP bulk transfers between 35 sites in 1997 [26]. In this work, he found that the conditional loss probability of data packets that were queued together was 10–20 times higher than the base loss rate. We compare our loss probabilities with those of Paxson and Bolot in Section 4.

2.2 Improved routing

While ARQ and FEC schemes can reduce the perceived loss rate of a particular Internet path, there may exist alternative paths that provide lower loss rates. Early ARPANET routing attempted to optimize path selection for congestion [17], but this was removed for scalability and stability reasons. Today, a wide variety of traffic engineering approaches are employed to refine path selection in an attempt to decrease congestion, packet loss, and latency, and increase available bandwidth [3]. Unfortunately these techniques generally operate over long time-scales. As a result of current backbone routing's ignorance of short-term network conditions, the route taken by packets is frequently sub-optimal [1, 30]. Recent network path selection products [24, 29, 32, 33] attempt to provide more fine-grained, measurement-based path selection for single sites.

Recent research in overlay networks has attempted to improve path conditions through indirect routing. The RON project uses active measurements to take advantage of some of these alternate paths [1]. Various Content Delivery Networks (CDNs) use overlay techniques and caching to improve the performance of content delivery for specific applications such as HTTP and streaming video. Overcast and other application level multicast projects attempt to optimize routes for bandwidth or latency [16].

2.3 Multi-path routing

The success of traffic engineering and overlay routing indicates the presence of redundant routes between many pairs of Internet hosts. A variety of approaches have been developed to leverage the existence of multiple, simultaneous paths through multi-path routing. Dispersity routing [21] and IDA [27] split the transfer of information over multiple network paths to provide enhanced reliability and performance. Simulation results and analytic studies have shown the benefits of this approach [5, 6]. Chen evaluated the use of parallel TCP flows to improve performance, but did not examine failures, or real Internet paths [11]. In addition, researchers have suggested combining redundant coding with dispersity routing to improve the reliability and performance of both parallel downloads [9] and multicast communication [31]. Akamai is reported to use erasure codes to take advantage of multiple paths between sites [20], and the designers of the Opus overlay system have proposed the future use of redundant transmission in an overlay, but, to our knowledge, have not yet evaluated this technique [8].

2.4 Sources of shared failures

Multi-path and alternate-path routing schemes make generous assumptions about path independence that may not hold when considering typical Internet paths, as we show in Section 4. Single-homed hosts share the same last-mile link to their provider, creating an obvious shared bottleneck and non-independent failure point. Even multi-homed hosts may have unexpected sources of shared failures. Many providers have some degree of shared physical infrastructure. In 2001, a single train derailment in the Howard Street tunnel in Baltimore, MD, impacted Internet service for at least 4 major US backbone carriers, all of whom used fiber running through the same physical location [13]. We also recently observed that many failures manifest themselves near the network edge, where routing protocols are less likely to be able to route around them [14]. Seemingly unrelated network prefixes often exhibit similar patterns of unreachability because of their shared infrastructure [2].

Network failures are not only caused by external factors, but may be the result of network traffic or other failures. These cascading logical failures can cause widespread outages that affect multiple paths or providers. Finally, denial of service attacks or other global Internet problems such as worms and viruses can cause correlated, concurrent failures. For instance, the "Code Red" worm, as a side-effect of its scanning, could crash certain Cisco DSL routers and other products, causing correlated failures based solely upon network access technology [12]. We provide an empirical evaluation of the independence of losses on a particular set of Internet paths in Section 4.

3. Design

For much of the paper, we study two mechanisms for enhanced packet routing: probe-based reactive overlay routing, and multipath redundant routing. These techniques would usually not be used independently. For instance, it's necessary to choose which intermediate nodes to use in redundant routing. The logical way to choose these nodes is via network measurements. The difference is the degree to which resources are allocated to measurements vs. redundant data, a trade-off that we consider further in Section 5. We note, however, that both ends of the spectrum are useful: reactive routing alone avoids numerous failures, and redundant routing using a randomly chosen intermediate avoids as many (or more) failures than reactive routing.

3.1 Probe-based reactive overlay routing

RON-like systems periodically send probes to determine the availability, latency, and loss rate of the paths connecting the nodes in the overlay. A RON must choose a probing rate R, and a network size N. A generalized scheme would also need to choose the sets of nodes that probe each other. Higher probing rates permit quicker reaction to network change, with more overhead. Larger networks have more paths to explore, but create scaling problems. In the system we evaluate, every node probes every other node once every 15 seconds. When a probe is lost, the node sends an additional string of up to four probes spaced one second apart, to determine if the remote host is down. The paths are selected based upon the average loss rate over the last 100 probes. These are similar to the parameters used in an earlier evaluation of RON [1], but the interval between probes is five seconds longer.

3.2 Redundant multi-path routing

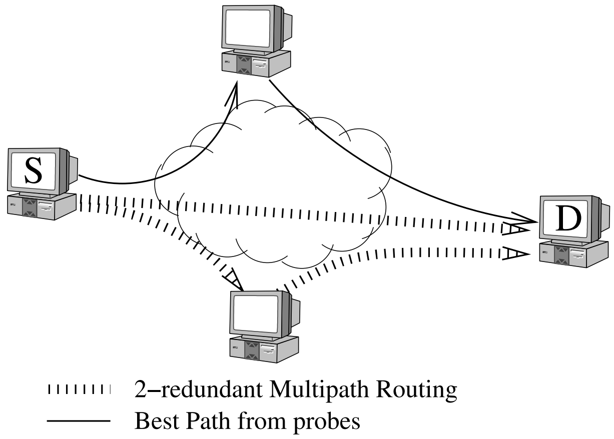


Figure 1: 2-Redundant multipath routing and best path routing. In this diagram, probing has determined that the best path is to travel indirectly via the top node. 2-Redundant multipath routing sends a packet down the direct path and via a random alternate hop, in this case, the bottom node.

Redundant multi-path routing sends redundant data down multiple paths, such that a certain fraction of lost packets can be recovered. In this study, we consider 2-redundant mesh routing [31], in which each packet is sent to the receiver twice, one on each distinct paths. In the most basic scheme, the first packet is sent directly over the Internet, and the second is sent through a randomly chosen intermediate node. We discuss the implications of our results on more complex FEC schemes in Section 5.2. When packet losses are independent, redundant data transmissions can effectively mask even high packet loss rates, but when losses are correlated, FEC schemes lose their effectiveness. We turn to an empirical study of this correlation to understand how common FEC schemes might fare in practice.

4. Evaluation

We evaluate the correlation of losses and failures on a deployed Internet testbed. Table 1 lists the 30 hosts used in our experiments. This testbed grew opportunistically as sites volunteered to host the nodes; no effort was made to explicitly engineer path redundancy. As Table 2 shows, the hosts are concentrated in the US, but span five countries on three continents. More importantly, the testbed hosts have a variety of access link technologies, from OC3s to cable modems and DSL links. We do not claim that this testbed is representative of the Internet as a whole. However, the nearly nine hundred distinct one-way paths between the hosts do provide a diverse testbed in which to evaluate routing tactics and packet loss relationships.

Table 3 lists the three datasets we examine. The first two, taken in 2002, were measured between 17 hosts on the RON testbed. The third was measured in 2003 between 30 hosts. RON_{wide} measured all combinations of mesh routing and probe-based routing to identify which combinations were most effective at reducing the probability of simultaneous losses. RON_{narrow} measures the three most promising methods with frequent one-way probes, sampling each path (for each method) every 45 seconds on average. RON_{2003}

Name	Location	Description
Aros	Salt Lake City, UT	ISP
AT&T	Florham Park, NJ	ISP
CA-DSL	Foster City, CA	1Mbps DSL
CCI	Salt Lake City, UT	.com
*** CMU**	Pittsburgh, PA	.edu
Coloco	Laurel, MD	ISP
*** Cornell**	Ithaca, NY	.edu
Cybermesa	Santa Fe, NM	ISP
Digitalwest	San Luis Obispo, CA	ISP
GBLX-AMS	Amsterdam, Netherlands	ISP
GBLX-ANA	Anaheim, CA	ISP
GBLX-CHI	Chicago, IL	ISP
GBLX-JFK	New York City, NY	ISP
GBLX-LON	London, England	ISP
Intel	Palo Alto, CA	.com
Korea	KAIST in Korea	.edu
Lulea	Lulea, Sweden	.edu
MA-Cable	Cambridge, MA	AT&T
Mazu	Boston, MA	.com
*** MIT**	Cambridge, MA	.edu in lab
MIT-main	Cambridge, MA	.edu data center
NC-Cable	Durham, NC	RoadRunner
Nortel	Toronto, Canada	ISP
*** NYU**	New York, NY	.edu
PDI	Palo Alto, CA	.com
PSG	Bainbridge Island, WA	Small ISP
* UCSD	San Diego, CA	.edu
*** Utah**	Salt Lake City, UT	.edu
Vineyard	Cambridge, MA	ISP
VU-NL	Amsterdam, Netherlands	Vrije Univ.

Table 1: The hosts between which we measured network connectivity. Asterisks indicate U.S. universities on the Internet2 backbone. Hosts in bold were used in the 2002 data.

measures a few additional routing types between more nodes, and over a longer period of time.[1] Table 4 lists the routing tactics for individual packets; probes consist of one or two packets sent via various routing methods.

We focus primarily on the RON_{2003} dataset, but highlight interesting differences from the prior datasets. This data set focuses on eight combinations of routing methods, collected from six sets of probes:

- *Direct*: A single packet using the direct Internet path.
- *Loss*: Probe-based reactive routing that attempts to minimize loss. Requires only probing overhead.

[1] Data is available at `http://nms.lcs.mit.edu/ron/`

Category	#
US Universities	7
US Large ISP	4
US small/med ISP	5
US Private Company	5
US Cable/DSL	3
Canada Private Company	1
Int'l Universities	3
Int'l ISP	2

Table 2: Distribution of the 30 testbed nodes.

Dataset	Samples	Dates
RON_{narrow}	4,763,082	8 Jul 2002 – 11 Jul 2002
RON_{wide}	2,875,431	3 Jul 2002 – 8 Jul 2002
RON_{2003}	32,602,776	30 Apr 2003 – 14 May 2003

Table 3: The three datasets used in our experiments. The RON_{narrow} dataset contains one-way samples for three routing methods. The RON_{wide} dataset has round-trip samples for eleven methods. The RON_{2003} dataset uses a larger number of probing hosts to measure six routing methods.

loss	loss optimized path (via probing)
lat	latency optimized path (via probing)
direct	direct Internet path
rand	indirectly through a random node

Table 4: The types of routes between measurement nodes. Probes consisted of one or more packets of these types, such as *direct rand* (one packet directly, one via a random intermediate node).

- *Lat*: Probe-based reactive routing that minimizes latency and avoids completely failed links.
- *Direct rand*: 2-redundant mesh routing, with no probing overhead. One copy of each packet is transmitted on the direct Internet path; the second over a random indirect overlay path. There is no delay between the packet transmissions. We use the first packet to predict the behavior of *direct* packets.
- *Lat loss*: Probe-based 2-redundant multi-path routing. In theory, this combination should be able to achieve the best of both worlds. It sends the first copy of each packet over a path selected to minimize loss, and the second over a path selected to minimize latency. We also use this to infer the *lat* packet.
- *Direct direct*: 2-redundant routing with back-to-back packets on the same path.
- *DD 10 ms*: 2-redundant routing with a 10ms gap between packets on the same path.
- *DD 20 ms*: As above, with a 20ms gap.

We present four major findings. First, losses on alternate paths are often not independent. If a packet sent directly from a source to a destination is lost, the chances are over 60% that a packet sent from that source to that destination via a random intermediate will also be lost. Second, the average Internet loss rate is quite low (0.42%) but individual paths at specific times have quite high loss rates. Third, mesh routing improves latency in two ways. Mesh routing is able to reduce the loss rate from 0.42% to 0.26%, which reduces retransmission and timeout latency. The overall packet latency is reduced by an average of 3 ms, but on 2% of the paths, mesh routing provides an average latency reduction of over 20 ms. Finally, path selection improves mesh routing. Paths with a greater degree of independence than the random paths used by mesh routing exist. Using probe based routing reduces the conditional loss probability to 55% for the second packet, suggesting that better path selection methods can improve the performance of mesh routing.

4.1 Method

Each node periodically initiates probes to other nodes. A probe consists of one or two request packets from the initiator to the target. The nodes cycle through the different probe types, and for each probe, they pick a random destination node. After sending the probe, the host waits for a random amount of time between 0.6 and 1.2 seconds, and then repeats the process.

Each probe has a random 64-bit identifier, which the hosts log along with the time at which packets were both sent and received. This allows us to compute the one-way reachability between the hosts. Most, but not all, hosts have GPS-synchronized clocks. We average one-way latency summaries and differences with those on the reverse path to average out timekeeping errors. Each probing host periodically pushes its logs to a central monitoring machine, where this data is aggregated. Our post-processing finds all probes that were received within 1 hour of when they were sent. We consider a host to have failed if it stops sending probes for more than 90 seconds, and we disregard probes lost due to host failure; our numbers only reflect failures that affected the network, while leaving hosts running. It is possible that we slightly under count outages caused by power failures or other events that affect both host and network.

4.2 Base network statistics

In contrast to earlier studies, the paths we measured had relatively low loss rates. Our paths' average loss rates ranged from 0% on many Internet2 or otherwise very fast connections, to about 6% between Korea and a DSL line in the United States. Figure 2 shows the distribution of average loss rates (over several days) on a per-path basis. The overall loss rate we observed on directly-sent single packets in 2003 was 0.42%, reduced from an earlier 0.74%. These changes may reflect topological changes to our testbed (it grew from 17 to 30 nodes, and some original nodes left or moved) as well as changes in the underlying loss rates, but the reduction in loss is still noteworthy. All of these loss rates are substantially lower than those observed in 1997 by Paxson.

Most of the time, the 20-minute average loss rates were close to zero; the `direct` line in Figure 3 shows the distribution of 20-minute loss samples. Over 95% of the samples had a 0% loss rate. The sampling granularity for the CDF is relatively coarse, so it groups low loss-rate conditions into the zero percent bin. During many hours of the day, the Internet is mostly quiescent and loss rates are low. During the worst one-hour period we monitored, the average loss rate on our testbed was over 13%.

4.3 Effects on loss rate

Table 5 examines the overall loss and latency results. Comparing the overall loss percentage (**totlp**) columns, we see that using probe data to pick better paths can reduce loss from 0.42% to 0.33%, with an almost insignificant effect on latency. Using random mesh routing can reduce the loss rate by almost 40%, and reduce the latency by a few milliseconds.

Sending two packets back to back on the same path results in loss improvements nearly as good as random mesh routing, especially if the two packets are delayed by 10 milliseconds (the **totlp** column in Table 5). Using path selection in conjunction with mesh routing results in a further improvement. These results suggest that the two methods are taking advantage of different situations: Mesh routing's packet redundancy is effective at masking transient congestion-triggered loss, possibly by de-correlating the losses through temporal shifting. In contrast, probe-based reactive routing avoids paths with longer-term pathologies. As would be expected, these two approaches can be combined—with high overhead—to get the best of both worlds, reducing losses by 45% and reducing latency by 13% (the *lat loss* row in Table 5).

Figure 3 shows the distribution of the 20-minute loss rates, and confirms the above conclusions. The loss avoidance routing is less effective at eliminating periods of small loss rates, but successfully avoids as many or more of the periods where the loss rate is high and sustained.

Our data suggests that most of the improvement in loss rates occurs during periods of elevated loss. Loss rates on the Internet are usually low (as is bandwidth utilization). 30% of the time during our monitoring, the overall loss rate between our nodes is less than 0.1%, and 68% of the time it is less than 0.2%. During the remaining periods, the average loss rate ranges up to 13%.

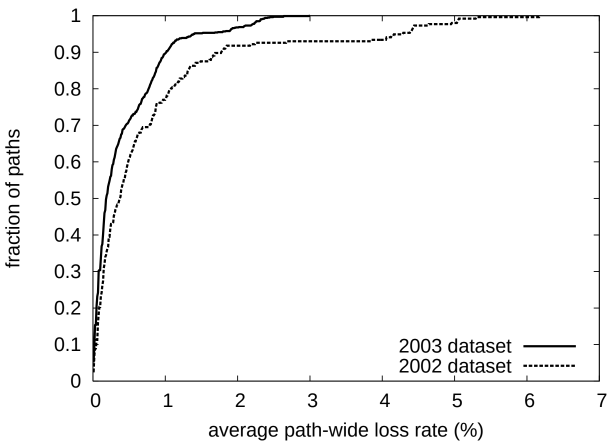


Figure 2: Cumulative distribution of long-term loss rates, on a per-path basis. 80% of the paths we measured have an average loss rate less than 1%.

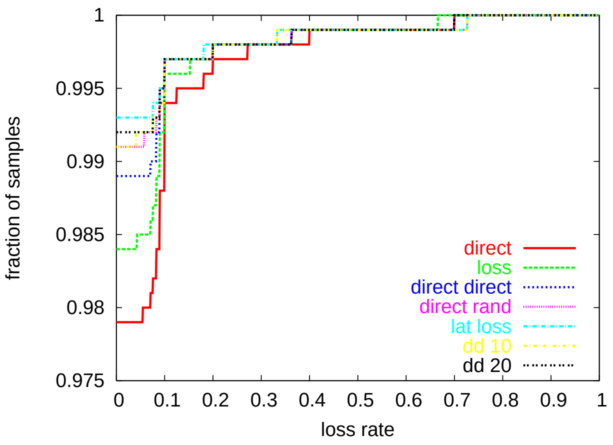


Figure 3: Cumulative distribution of 20-minute loss rates, on a per-path basis.

		Simple Redundancy			Reactive		Mesh	Both
Loss %	*direct*	*direct direct*	*dd 10ms*	*dd 20 ms*	*lat*	*loss*	*direct rand*	*lat loss*
> 0	8817	5183	4024	3832	10695	7066	3846	3353
> 10	1999	1361	1291	1275	1716	1362	1236	1134
> 20	962	799	796	783	849	791	793	757
> 30	630	585	591	575	604	573	579	563
> 40	486	480	481	465	484	468	468	451
> 50	379	377	367	359	363	359	369	334
> 60	255	251	245	249	231	219	235	215
> 70	130	130	130	128	118	106	125	114
> 80	74	73	65	64	57	59	60	56
> 90	31	31	37	30	16	31	28	16

Table 6: Hour-long high loss periods, by routing method. Much of the benefit from reactive routing comes from avoiding longer periods of high loss, and mesh routing successfully improves losses when the overall loss rate is low. There were an equal number of total sampling periods for each method.

Type	1lp	2lp	totlp	clp	lat
			2003		
*direct**	0.42	-	0.42	-	54.13
*lat**	0.43	-	0.43	-	48.01
loss	0.33	-	0.33	-	55.62
direct rand	0.41	2.66	0.26	62.47	51.71
lat loss	0.43	1.95	0.23	55.08	46.77
direct direct	0.42	0.43	0.30	72.15	54.24
dd 10 ms	0.41	0.42	0.27	66.08	54.28
dd 20 ms	0.41	0.41	0.27	65.28	54.39
			2002		
*direct**	0.74	-	0.74	-	69.54
*lat**	0.75	-	0.75	-	69.43
loss	0.67	-	0.67	-	76.07
direct rand	0.74	1.85	0.38	51.17	68.33
lat loss	0.75	1.53	0.37	49.82	66.73
direct direct	-	-	-	72.70	-

Table 5: One-way loss percentages. Items marked with an asterisk were inferred from the first packet of a two-packet pair. 1lp and 2lp are the percent chances of losing the first and second packets. Totlp is the chance of losing both. Clp is the conditional loss probability percentage for the second packet. Lat is the average one-way latency in milliseconds. The 2002 *direct direct* data was extracted from the RON_{wide} dataset, and lacks comparable one-way loss and latency data.

For these periods of more serious loss, Table 6 examines the distribution of loss rates over one-hour windows.[2] Comparing the *loss* and *direct rand* rows in the table shows that mesh based routing is much more effective at reducing small amounts of loss, but as the loss rate grows more serious, probe-based methods begin to equal or outperform meshing.

4.4 Conditional loss probabilities

In the RON_{wide} and RON_{2003} datasets, we examined a wider number of probe types, including back-to-back *direct* packets. Bolot [7] examined packets separated by 8 ms, and found that their conditional loss probability was 60%. Paxson [26] examined TCP packets that queued together at a router, finding their conditional loss probability to be about 50%. In our experiments, back-to-back packets had a higher conditional loss probability—72.7% in 2002 and 72.15% in 2003—probably because we sent them with *no* intervening delay. The conditional loss probability of a packet sent through a random intermediate node was only 50% in 2002 and 62% in 2003. Taken relative to two direct packets, this indicates an appreciable difference in conditional loss probabilities when traversing an intermediate host, but these figures are more understandable when we consider the conditional loss probabilities of delayed packets. With a 10-ms delay, we observe a 66% conditional loss probability, similar to Bolot's 60%, which bridges half of the gap between back-to-back packets and those sent through an intermediate node.

Figure 4 shows the cumulative distribution of conditional loss probabilities across hosts, on the 115 paths on which we observed first-packet losses. With back-to-back packets, half of the hosts had a 100% conditional loss probability. This data suggests that redundant routing on the same path is likely to fall prey to burst losses in a way that multi-path avoids.[3]

The conditional loss probabilities of packets sent indirectly changed considerably from our 2002 to 2003 datasets, but the CLP for back-to-back packets on the same path was virtually identical. This suggests that the back-to-back loss probabilities may be more a function of router behavior and queuing dynamics, and that the indirect

[2] We used one hour windows to ensure that we had sufficient samples to detect the loss rate with fine granularity.

[3] These numbers are derived from relatively few losses, so there are likely excessive samples at 100% that should be in the 90s.

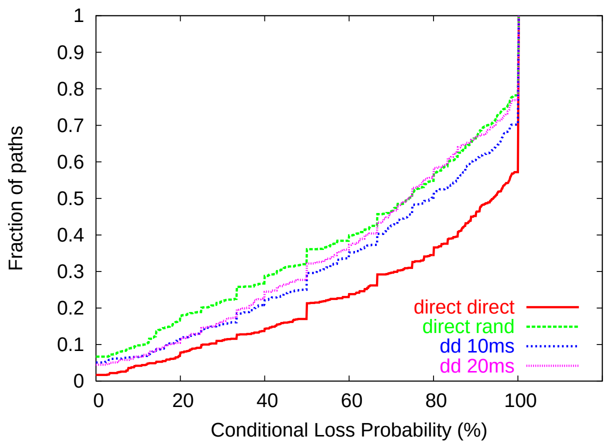


Figure 4: Cumulative distribution of conditional loss probabilities for the second packet in a back-to-back packet transmission. Two back-to-back direct packets have a higher CLP than two back-to-back packets where one is sent through a random intermediate. The highest average loss rate for a direct path was 4–6%; the conditional loss probabilities are much higher.

probabilities are more subject to variance based on changes in network topology.

4.5 Effects on latency

Figure 5 shows the cumulative distribution of one-way latencies in RON_{2003} for paths whose latency is over 50 ms (30% of the paths we measure). For paths with shorter latencies, the differences between routing methods are mostly insignificant. Overall, the average direct Internet path latency is 54.13 ms. Latency optimized routing reduces this by 11%, primarily by improving exceptionally bad paths. Many of the largest latency improvements we observed came from a period around 6 May 2003 when many of the paths to the Cornell node experienced latencies of up to 1 second. This explains why the 2003 dataset shows more latency improvement than the 2002 dataset, which was too short to observe many pathologies.

Interestingly, the improvement from mesh routing (2–3 ms overall) is mostly the same, regardless if the technique is used with or without reactive routing. Like the loss optimization case, this suggests that these two techniques improve latency by avoiding different sources of delays. Overall, mesh routing also made improvements to the pathological cases (`Cornell` and `Korea`), but the benefits were spread more evenly across a wider selection of paths.

4.6 Other combinations of methods

RON_{wide}'s broader examination confirmed that the three routing methods upon which we focused—*loss, direct rand,* and *lat loss*—are the most interesting. Some other methods had a few noteworthy features, however. The loss probability for `rand rand` was as low as *direct rand*, though its latency was far worse. The latency of *direct lat* was better than any other method, by several milliseconds. Table 7 shows the results of this more broad examination.

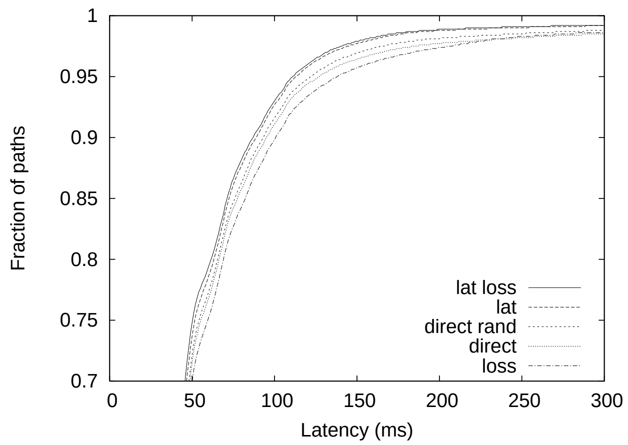


Figure 5: Cumulative distribution of packet one-way latencies for higher-latency paths. Note that the CDF begins at 70%; the remainder of the paths had latencies under 50 ms.

Type	1lp	2lp	totlp	clp	RTT
direct	0.27	-	0.27	-	133.5
rand	1.12	-	1.12	-	283.0
lat	0.34	-	0.34	-	137.0
loss	0.21	-	0.21	-	151.9
direct direct	0.29	0.49	0.21	72.7	134.3
rand rand	1.08	1.12	0.12	11.2	182.9
direct rand	0.29	1.20	0.12	39.2	130.1
direct lat	0.29	0.95	0.11	39.3	123.9
direct loss	0.27	1.06	0.11	40.0	130.5
rand lat	1.15	0.41	0.11	9.3	131.3
rand loss	1.11	0.28	0.11	9.9	140.4
lat loss	0.36	0.79	0.10	29.0	128.8

Table 7: One-way loss percentages for the expanded set of routing schemes from the 2002 RON_{wide} dataset. This table presents round-trip latency numbers, not one-way latency numbers.

5. Implications

What do our results say about the design of improved routing schemes? This section considers the cost, theoretical benefits, and realized benefits from these schemes to understand the trade-offs involved in their use.

Applications using improved routing schemes have certain requirements from the network, much like a service level agreement: a certain average loss rate, average latency, maximum latency, and a maximum amount of outage time. These applications run in the context of an underlying network with its own loss rates, patterns of loss (outages, burst losses, and congestive losses), and latency variation.

In our model, application designers have a certain "bandwidth budget" that they can spend to attempt to meet their goals. They can spend this bandwidth via probing, packet duplication, or a combination. For a given application, what is the best allocation of that budget between reactive routing and mesh routing?

5.1 Probe-based reactive overlay routing

Reactive routing assumes that some path through the overlay can provide good service; FEC and redundant routing can even construct a better path from independent bad paths. This creates clear differences between the failure scenarios that these methods can handle. If no individual paths are good, reactive routing does not help. On the other hand, if failures result in only a small subset of functional paths through the network, a probe-based reactive mechanism is better positioned to utilize these paths.

Benefit: Reactive routing circumvents path failures in time proportional to its probing rate. For the N possible one-hop paths from a source to the destination, where each has a loss probability p_i,

$$p_{reactive} = min_i \left(p_i\right)$$

Reactive routing is constrained to the latency of the best path, as well. In general, the path with the best loss rate may not have the best latency [1].

Cost: The cost of all-paths probing and route dissemination is fixed—each host must send and receive $O(N^2)$ data. The cost is not dependent upon the amount of traffic in the flow; hence, it can be large in comparison to a thin data stream, or negligible when used in conjunction with a high bandwidth stream.

Reality: We do not know the loss rate of the theoretical best path between the nodes, but reactive routing performs about as expected within the constraints of how quickly it can adapt. The major question is if the probing and routing overhead is worth the improvement it provides; from our data, it's clear that many paths experience no benefit from reactive routing, but in the cases where they do, the benefits are often large.

5.2 Redundant multi-path routing

Redundant encoding is generally accomplished through the use of a FEC technique that adds extra packets to the data stream, rather than increasing the size of individual packets, since increasing the packet size may bump into path MTU limitations. An efficient FEC sends the original packets first, to avoid adding latency in the no-loss case—the so called standard codes. Reed-Solomon erasure codes are a standard FEC method that provide a framework with which to apply variable amounts of redundancy to groups of packets [28]. As a simpler case, packets can simply be duplicated and sent along multiple paths, as is done in mesh routing [31]. We restrict our evaluation to using this simple encoding over two paths, so-called 2-redundant routing, since we believe the number of truly loss-independent paths between two points on the Internet is relatively low[4]. FEC without path diversity can avoid random losses, but cannot tolerate large burst losses or path failures.

Benefit: When paths are completely independent, redundant routing can handle the complete failure of up to $(R-1)$ paths per node. When packet losses are independent, redundant routing on N paths whose loss probability is p_i can improve the overall loss rate to the product of their individual loss rates

$$p_{redundant} = \prod_{i=1}^{N} p_i$$

[4]For example, if there are three such paths, any redundant encoding must be able to tolerate a loss of at least $\frac{1}{3}$ of the packets in a window, which would require at least 50% overhead anyway.

2-redundant routing on random paths achieves, in expectation, the square of the average loss rate:

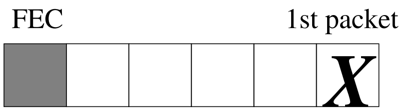
$$E\left[p_{2-redundant}\right] = \left(E\left[p_i\right]\right)^2$$

When used in conjunction with the direct Internet path, multi-path routing has good expected latency. Multi-path routing can provide a smaller expected latency even if the alternate paths have similar latency, while still providing reduced loss [31].

Cost: The cost of simplistic N-redundant routing is a factor of N. A 2-redundant routing scheme results in a doubling of the amount of traffic sent. The cost does not depend on the size of the network.

Reality: Not surprisingly, 2-redundant mesh routing does not approach a p^2 improvement in loss rates, due to the high correlation of lost packets. It does, however, produce measurable improvements in both loss and latency, particularly on paths with poor transmission characteristics.

The high degree of loss correlation (over 50%) on the paths we measured suggests that FEC schemes must add considerable protection to packets or spread their redundancy over a significantly large window to avoid most losses. While this may work for bulk data flow, avoiding lower levels of loss is often more important in interactive applications, where this extra recovery delay may not be acceptable. Consider a FEC scheme to correct for 20% loss or less. This scheme must add 1 redundant packet for every 5 data packets:



If the first packet in a packet train is lost, the high conditional loss probability tells us that there is a 70% chance that the second packet will *also* be lost - so to avoid this, the FEC information must be spread out by nearly half a second if sending packets down the same path. For most terrestrial wired networks, this extra delay eliminates whatever latency savings would have been obtained by avoiding retransmission.

5.3 Design space and Internet limitations

There are some situations where redundant routing is not appropriate. Running an unmodified bulk-flow TCP directly over a redundantly-enhanced path would be problematic because the apparent low loss rate will trick TCP into taking far more than its fair share of the bandwidth. However, running low-rate TCPs (or any application where it's known that the application will not exceed its share of the channel) might be acceptable.

In contrast, reactive overlay routing is appropriate for most kinds of traffic, though its overhead may be prohibitive for low-bandwidth flows. For low-bandwidth flows, redundant approaches can offer similar benefits with lower overhead. For high-bandwidth flows, FEC approaches result in overhead proportional to the size of the flow, whereas alternate-path routing has constant overhead. The overhead of both schemes as a function of bandwidth and the number of nodes, N, is shown below.

Probe-based reactive routing imposes overhead that grows rapidly with the size of the network, but is independent of flow size. 2-Redundant mesh routing imposes a size-independent $2x$ overhead per packet:

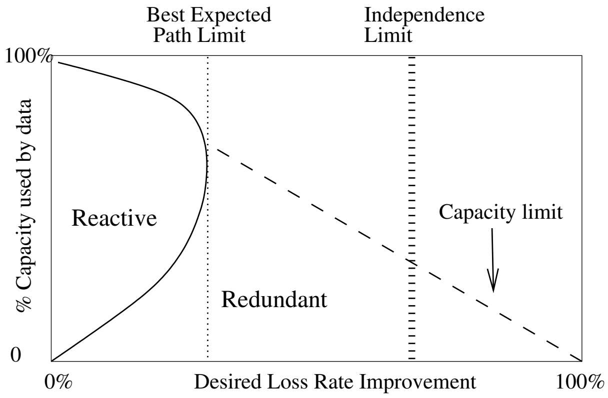


Figure 6: When to use reactive or redundant routing. Reactive routing asymptotically approaches the performance of the best expected path. Its probes require some network bandwidth. Redundant routing is limited by the capacity of the network relative to the flow bandwidth, and by the degree of independence of its paths. Within these bounds, the flow's bandwidth determines whether reactive or redundant routing provides the required improvement with smaller overhead.

Probe-based	2-Redundant Mesh
$1 + \frac{N^2}{Bandwidth}$	2

To understand the space in which each method is applicable, we compare two parameters, the desired improvement in loss rate and the amount of network bandwidth being used by the original data stream. We define "loss rate improvement" as

$$\frac{Loss_{Internet} - Loss_{method}}{Loss_{Internet}}$$

Figure 6 depicts this space graphically, and shows the bounds that limit the performance of each scheme. We consider three major bounds.

Best Expected Path Limit: Probing can only find the best network path at any given time. As the probing frequency increases, the achieved performance asymptotically approaches the performance of the best expected path.

Capacity Limit: Both schemes face a capacity limit. If the original data stream is using 100% of the available capacity, neither scheme can make an improvement: Probing cannot send probes, and redundant routing cannot duplicate packets. The bandwidth required by redundant routing is linear with the flow rate. The "constant" bandwidth required by reactive routing decreases slightly with a relaxation in loss rate demands, because when less improvement is required, the probing rate can be reduced.

The slope of the capacity limit is negative: achieving greater gains requires using more of the bandwidth. For redundant routing, achieving extremely high reliability requires sending multiple copies of each packet, so the capacity limit is quite low. For probing, maximizing reliability requires probing at an extremely high frequency, which also takes away capacity for data transmission.

Independence Limit: Redundant routing is ultimately limited by the loss and failure independence of the network. The actual values of this limit are unknown, but our evaluation suggests that without expressly designing a network for path independence, having 50% of failures and losses occur independently would be a reasonable upper limit for designers to consider.

6. Conclusions and future work

Overlay networks are emerging as an increasingly popular way to deploy new Internet services, and have the ability to overcome performance and resilience shortcomings in the underlying Internet substrate. In this paper, we examined two techniques that reduce end-to-end loss rates and latency by leveraging path independence in the underlying network. Probe-based reactive overlay routing takes advantage of path diversity by trying to find the best path among its nodes. Mesh routing send duplicate copies of packets along alternate paths in the hope that paths fail independently.

Our evaluation shows that there is a reasonable—but not large—degree of loss and failure independence in the underlying Internet links, such that about 40% of the losses we observed were avoidable. Many of the benefits of routing indirectly could also be achieved by sending duplicate copies of packets with a 10 or 20 ms delay between them along the same path.

We examined conditional losses for low-bandwidth probe traffic and did not consider the impact of additional traffic on the underlying loss rate. The interdependence of losses could increase as a function of network load (for instance, RED queues could fill up and begin exhibiting drop-tail behavior). An interesting question for future work is if over-provisioning network capacity to support probing and meshing overhead would itself reduce the losses to the same degree that loss avoidance tactics do, or if the benefits we observed in this study would also arise in a bandwidth-enhanced context.

Finally, our study showed that reactive routing and redundant routing can work in concert to provide further loss reduction and to further de-correlate the loss of back-to-back packets. It would be interesting to explore what combinations of these methods prove to be sweet spots for common patterns of Internet traffic.

Acknowledgments

We thank our anonymous reviewers for their accurate, helpful, and remarkably consistent feedback. Nick Feamster, Michel Goraczko, Kyle Jamieson, Jinyang Li, Allen Miu, Rodrigo Rodriguez, and Stan Rost all provided great comments during the development of this paper.

7. References

[1] Andersen, D. G., Balakrishnan, H., Kaashoek, M. F., and Morris, R. Resilient Overlay Networks. In *Proc. 18th ACM SOSP* (Banff, Canada, Oct. 2001), pp. 131–145.

[2] Andersen, D. G., Feamster, N., Bauer, S., and Balakrishnan, H. Topology Inference from BGP Routing Dynamics. In *Proc. Internet Measurement Workshop* (Marseille, France, 2002).

[3] Awduche, D. O., Chiu, A., Elwalid, A., Widjaja, I., and Xiao, X. *Overview and Principles of Interent Traffic Engineering*. Internet Engineering Task Force, May 2002. RFC 3272.

[4] Balakrishnan, H., Seshan, S., and Katz, R. Improving Reliable Transport and Handoff Performance in Cellular Wireless Networks. *ACM Wireless Networks 1*, 4 (Dec. 1995).

[5] Banerjea, A. Simulation study of the capacity effects of dispersity routing for fault tolerant realtime channels. In *Proc. ACM SIGCOMM* (Aug. 1996), pp. 194–205.

[6] Bestavros, A. An adaptive information dispersal algorithm for time-critical reliable communication. In *Network Management and Control, Volume II*, I. Frish, M. Malek, and S. Panwar, Eds. Plenum Publishing Co., New York, New York, 1994, pp. 423–438.

[7] Bolot, J. End-to-End Packet Delay and Loss Behavior in the Internet. In *Proc. ACM SIGCOMM* (San Francisco, CA, Sept. 1993).

[8] Braynard, R., Kostic, D., Rodriguez, A., Chase, J., and Vahdat, A. Opus: an overlay peer utility service. In *Proc. 5th International Conference on Open Architectures and Network Programming (OPENARCH)* (June 2002).

[9] Byers, J. W., Luby, M., and Mitzenmacher, M. Accessing multiple mirror sites in parallel: Using tornado codes to speed up downloads. In *Proc. IEEE Infocom* (Mar. 1999), pp. 275–283.

[10] Byers, J. W., Luby, M., Mitzenmacher, M., and Rege, A. A digital fountain approach to reliable distribution of bulk data. In *Proc. ACM SIGCOMM* (Aug. 1998), pp. 56–67.

[11] Chen, J. *New Approaches to Routing for Large-Scale Data Networks*. PhD thesis, Rice University, 1999.

[12] Cisco Security Advisory: Code Red Worm - Customer Impact. `http://www.cisco.com/warp/public/707/cisco-code-red-worm-pub.shtml`, 2001.

[13] Donelan, S. Update: CSX train derailment. `http://www.merit.edu/mail.archives/nanog/2001-07/msg00351.html`.

[14] Feamster, N., Andersen, D., Balakrishnan, H., and Kaashoek, M. F. Measuring the effects of Internet path faults on reactive routing. In *Proc. Sigmetrics* (San Diego, CA, June 2003).

[15] Gallager, R. G. *Low-Density Parity-Check Codes*. PhD thesis, Massachusetts Institute of Technology, 1963.

[16] Jannotti, J., Gifford, D. K., Johnson, K. L., Kaashoek, M. F., and O'Toole Jr., J. W. Overcast: Reliable multicasting with an overlay network. In *Proc. 4th USENIX OSDI* (San Diego, California, October 2000), pp. 197–212.

[17] Khanna, A., and Zinky, J. The Revised ARPANET Routing Metric. In *Proc. ACM SIGCOMM* (Austin, TX, Sept. 1989), pp. 45–56.

[18] Labovitz, C., Ahuja, A., Bose, A., and Jahanian, F. Delayed Internet Routing Convergence. In *Proc. ACM SIGCOMM* (Stockholm, Sweden, September 2000), pp. 175–187.

[19] Labovitz, C., Malan, R., and Jahanian, F. Internet Routing Instability. *IEEE/ACM Transactions on Networking 6*, 5 (1998), 515–526.

[20] Lewin, D. Systems issues in global Internet content delivery, 2000. Keynote Address at 4th USENIX OSDI Conference.

[21] Maxemchuk, N. F. *Dispersity Routing in Store and Forward Networks*. PhD thesis, University of Pennsylvania, May 1975.

[22] McAuley, A. J. Error Control for Messaging Applications in a Wireless Environment. In *Proc. INFOCOM Conf.* (Apr. 1995).

[23] Miller, G. Overlay routing networks (Akarouting). `http://www-math.mit.edu/~steng/18.996/lecture9.ps`, Apr. 2002.

[24] Opnix. Orbit: Routing Intelligence System. `http://www.opnix.com/newsroom/OrbitWhitePaper_July_2001.pdf`, 2001.

[25] Paxson, V. End-to-End Routing Behavior in the Internet. In *Proc. ACM SIGCOMM '96* (Stanford, CA, Aug. 1996), pp. 25–38.

[26] Paxson, V. End-to-End Internet Packet Dynamics. In *Proc. ACM SIGCOMM* (Cannes, France, Sept. 1997), pp. 139–152.

[27] Rabin, M. O. Efficient dispersal of information for security, load balancing and fault tolerance. *J. ACM 36*, 2 (Apr. 1989), 335–348.

[28] Rizzo, L., and Vicisano, L. RMDP: An FEC-based reliable multicast protocol for wireless environments. *Mobile Computing and Communications Review 2*, 2 (1998).

[29] RouteScience. Whitepaper available from `http://www.routescience.com/technology/tec_whitepaper.html`.

[30] Savage, S., Collins, A., Hoffman, E., Snell, J., and Anderson, T. The End-to-End Effects of Internet Path Selection. In *Proc. ACM SIGCOMM* (Boston, MA, 1999), pp. 289–299.

[31] Snoeren, A. C., Conley, K., and Gifford, D. K. Mesh-based content routing using XML. In *Proc. 18th ACM SOSP* (Banff, Canada, Oct. 2001), pp. 160–173.

[32] Sockeye. `http://www.sockeye.com/`.

[33] Stonesoft. Multi-link technology white paper. `http://www.stonesoft.com/files/products/StoneGate/SG_Multi-Link_Technol%ogy_Whitepaper.pdf`, Oct. 2001.