Buffer Space Tradeoffs for VoIP QoS in Deflection Networks

André Muezerie*
Physics Institute of São Carlos
University of São Paulo
Caixa Postal 369, São Carlos-SP, 13560-970 Brazil
andremuz@if.sc.usp.br

Ioanis Nikolaidis
Department of Computing Science
University of Alberta
Edmonton, Alberta, Canada T6G 2E8
yannis@cs.ualberta.ca

Pawel Gburzyński
Department of Computing Science
University of Alberta
Edmonton, Alberta, Canada T6G 2E8
pawel@cs.ualberta.ca

March 15, 2005

Abstract

We consider the issue of QoS (Quality of Service) in a network catering to isochronous VoIP (Voice over IP) sessions. Our simulation experiments strongly suggest that the single–path paradigm of implementing network–layer virtual circuits neither yields the best utilization of network resources (buffer space) nor is it able to provide the best end-to-end service in terms of loss rate and delay. Instead, it turns out that the dynamic exploration of alternative paths at the packet level, i.e., asynchronous deflection routing assisted with small buffer space at the routers, results in a more efficient, and also more economical, delivery. By the latter we mean the lower overall buffer space requirements, aggregated over the routers as well as the end–points of the isochronous sessions.

Keywords: routing, multiple path routing, deflection, VoIP, QoS

*André Muezerie is grateful to CAPES for the generous support of his study leave.

1 Introduction

Voice-over-IP (VoIP) is quickly becoming a popular application that demonstrates how packet–switched networks can compete in providing services historically associated with circuit–switched reservation–based networks. VoIP sessions pose a wide range of Quality of Service (QoS) requirements. First, they have to be capable of timely delivery (low delay requirement) and low delay variability (jitter), where a violation of the latter could result in perceptible discontinuities/interruptions due to buffer underflow/overflow at the receiver. Second, packet loss has to be kept reasonably low, at a level where loss/error concealment techniques can be applied without resulting in noticeable artifacts. Finally, while VoIP bandwidth requirements for a single session are trivially small by today's standards, it is expected that statistical multiplexing of a large number of VoIP calls will tend to become the norm (at a scale not unlike that of large telephone service carriers). In any event, the degree of statistical multiplexing is a key factor for a provider: the higher the degree of statistical multiplexing the more cost–effective (and profitable) the VoIP deployment.

The anticipated scale of VoIP clearly indicates that fine–grained admission control of the data stream (e.g., at the level of individual calls) would result in a substantial overhead and should be avoided. It is therefore not surprising to see solutions based on forms of "bulk" trunk reservation imposed onto the packet-switched infrastructure in order to cope with the QoS requirements of VoIP connections. For example, once trunks have been reserved for the VoIP traffic (providing a "pool" of bandwidth in which to multiplex VoIP calls), specialized forms of routing, e.g. [1], attempt to relieve congestion when it develops. To this end, alternative routing schemes (other than shortest path) over reserved Virtual Leased Lines (VLLs) are used. The ability to establish VLLs is related to the relative success of modern routers to isolate traffic groups via link scheduling schemes (notably forms of Weighted Fair Queuing). On the surface, the use of routing over VLLs (and any routing–based congestion control in general) to relieve VoIP congestion appears reasonable because VoIP sessions possess no congestion control of their own. We note that the routing–based congestion control can be performed in a particularly flexible manner if the routing and the forwarding tasks are separated, as advocated by MPLS [2].

All these solutions are in agreement with the common wisdom that the most QoS friendly implementation of an end-to-end session should involve a network-layer virtual circuit, whereby all packets of the session follow the same path from source to destination. Intuitively, the deterministic character of such a connection makes it easier to set aside the right amount of resources at every intermediate node and predict what is going to happen when several virtual circuits cross at the same router. Consequently, most work on QoS-driven resource allocation has focused on path selections algorithms [3, 4, 5, 6, 7], assuming that once selected the (single) path will be followed by all packets of the session. This approach essentially equates a transport-layer session with a network-layer virtual circuit, even if (as in the Internet) the network-layer virtual circuit is not explicit. In other words, its proponents are willing to dispense the freedom afforded by the network layer at the apparent gain of a better control over resource usage.

In addition to the requisite capability of routers to split a link bandwidth into a small set of classes, one more engineering concern is how to dimension the router buffers. According to recent results, the usefulness of buffering in the core routers for the sake of congestion–relief is highly debatable. In particular, [8] demonstrates that even TCP can live comfortably with a 99% reduction of the buffer space at the routers (compared to typical large buffers commonly employed today). These observations indirectly suggest that the intuitions behind the contemporary prevalent trends in QoS provisioning via core router buffer dimensioning are not necessarily correct.

In this paper, we demonstrate that routing across virtual circuits, i.e., source-destination paths pre-established for sessions, need not yield the best use of network resources. We argue that the sacrifice of the flexibility inherent in packet-level routing should be viewed as a tradeoff in the resource control game, rather than a prerequisite for its success. The lack of network-layer circuits is an important feature of the Internet, which makes it qualitatively different from the old telephone network. Thus, imposing onto it trunk-like structures may not be the best way of turning the network into a VoIP-friendly infrastructure.

Our results also indicate that real–time isochronous traffic sessions, such as VoIP,[1] *do not* benefit from buffering in the network core, if the goal of that buffering is to improve resource sharing in virtual circuits. They do benefit from the kind of buffering aimed at decoupling the sending/receiving rate from the momentary availability of links (as do all isochronous loads handled by asynchronous servers), but this problem seems to be better addressed by putting the buffers elsewhere.

The rest of our paper is organized as follows. Section 2 reviews deflection routing techniques as

[1]Our observations naturally extend to video and other kinds of data streams that are not amenable to congestion–throttling actions.

alternatives to the traditional routing paradigm of the Internet. Section 3 describes a model that allows us to study the tradeoff regarding buffer placement at the core (interior) versus periphery of the network. Section 4 illustrates how deflection outperforms the traditional approach to routing, including alternative path routing strategies. The results shown there clearly demonstrate that buffer space is better utilized at the periphery rather than at the core of a network. Conclusions and directions for further research are summarized in Section 5.

2 Deflection routing

Consider a router within the network core that is about to forward a packet belonging to a transport-layer session. Regardless of the assumed end-to-end routing paradigm, this is the complete list of options available to the router:

1. Queue the packet for transmission on the "best" output port, i.e., one offering the "most attractive" route to the destination.

2. Drop the packet, e.g., if no buffer space is available for its storage.

3. Queue the packet for transmission on an output port that is considered a secondary choice (by the assumed route preference scheme).

With the conventional forwarding paradigm based on pre-established routes, the third possibility is excluded: the router has no choice regarding the selection of the output port. The optimization effort regarding the utilization of network resources is thus directed towards a precise description of what is meant by the "best" route to the destination, as well as determining the right packet scheduling policy at the router. The latter can be interpreted as part of the buffer management scheme, as it also prescribes the packet dropping rules.

The third option is only admissible in *deflection networks*, where the paths traveled by different packets of the same session can differ in an *a priori* unpredictable manner. Traditionally, deflection networks have been studied as synchronous systems, exemplified by Manhattan-street networks [9, 10, 11, 12, 13] and Shuffle-exchange networks [14, 15], where the routing decision affects a collection of packets arriving at the router in one cycle at all input ports. The routing problem in such a scenario [16] is defined as finding the best assignment (permutation) of these packets to the

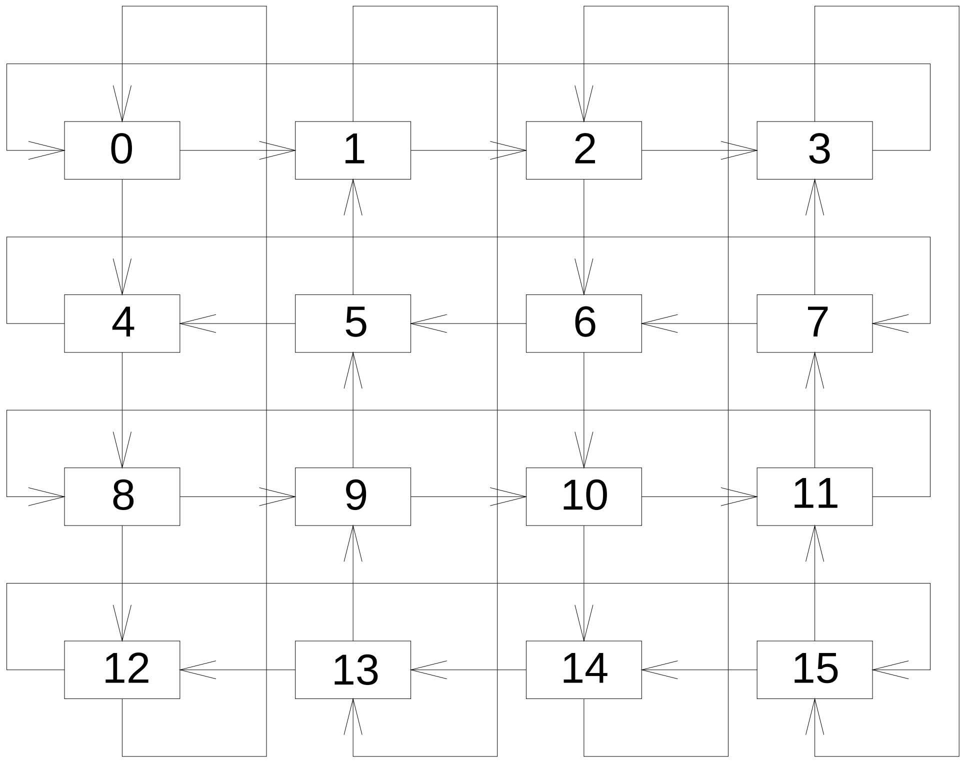


Figure 1: The topology of Manhattan-street network

output ports, usually one that minimizes the combined length of the next-hop shortest paths to the destinations. Owing to the regularity of the topologies considered as backbones for deflection networks, as well as their relatively small node order (the number of incoming and outgoing links—see Fig. 1), the optimization problem of routing is usually equivalent to a simple set of trivially computable rules [11].

The deflection approach to routing in wide-area networking has received little attention in the past. First, as we mentioned in the introduction, the prevailing intuition behind QoS provisioning in large irregular networks calls for deterministic resource allocation along a known path. Second, as most performance studies in deflection networks have dealt with synchronous routing in regularly shaped networks operating in a slotted manner, the results of those studies have not been considered interesting (or even applicable) from the viewpoint of realistic networks, like the Internet.

Some of our earlier studies [17, 18, 19, 20] were aimed at removing the rigid prerequisites of deflection routing and bringing this concept closer to a possible implementation in real-life irregular networks. As it turns out, deflection routing can be implemented in an unslotted manner in irregular networks, in a way that deals with one packet at a time and thus avoids the combinatorial complexity of the synchronous case for nodes with a non trivial degree [17]. The addition of small buffer space at the routers greatly improves the overall quality of routing decisions [21] and brings the maximum throughput achievable by the network to the level of its store and forward counterpart using infinite buffers. The scalability theorem for deflection networks [22] states that, as a matter of principle, deflection routing scales well with the increasing network size: asymptotically, the statistical increase of paths caused by deflection tends to disappear in networks of a reasonable shape and non-trivial size. It is possible to implement feedback and bandwidth allocation mechanisms that render deflection networks suitable for real-time applications [19]. There also exist deflection schemes imposing hard bounds on the path length while retaining the original spirit of deflection that postulates no packet loss in the network [18].

One apparent flaw of deflection routing, and the primary cause of the lack of interest in this concept as a prospective routing paradigm of serious networking, is the inherent unpredictability of paths. First, it is difficult to bound the delay and jitter: a packet can potentially stray arbitrarily far from the shortest path, possibly looping, i.e., visiting the same intermediate node more than once. Second, packets of the same session may arrive at the destination misordered. Thus, reassembly buffers are required to play them back in the original transmitted sequence.

Notably, as long as the network topology is not pathological, deflection networks (including asynchronous unslotted networks) appear quite stable, also under saturating load [17]. Although occasionally a packet may suffer a large delay, most of them tend to stay close to the shortest path. Formally, from the viewpoint of the QoS requirements of an isochronous session, a stray packet is lost. In a store-and-forward network, packets can be lost as well—because of the lack of buffer space at the routers or because of excessive delays caused by buffering. Thus, there is no fundamental difference between the two routing paradigms: deflecting a packet from the shortest path is equivalent to buffering it in the network. In both cases, excessive buffering in network core translates into loss. Conceptually, the choice is between storing in the explicit buffer space at the router or storing in the "virtual buffer" implied by the bandwidth–delay product of the links into which the packet has been diverted. In both cases, the delay incurred by storing the packet may contribute to its late arrival at the destination. Additionally, in the first case, the buffer may overflow and the packet may be immediately lost.

The lack of explicit network-layer virtual circuits in IP routing makes it natural to consider deflection as one of the options, if the preferred output link happens to be congested. When this option is admissible, one can expect that the amount of buffer space at the router can be reduced, as deflection will take over some of the traditional role of buffering. On the other hand, the need to play back misordered packets at the destinations may pose demands for reassembly buffer space. Thus, the adoption of deflection can also be viewed as trading off buffer space at the routers for buffer space at the end nodes.

In [23], it is argued that deflection may be a fundamentally better approach than excessive buffering or dropping in the case of congestion. First, managing the private per-session playout buffer at a destination is considerably simpler (and better defined as a problem) than managing the shared buffer space at a router. This is because the destination applies the buffer to a single session at the exact point where its delivered QoS parameters can be monitored with ultimate fidelity and authority. Second, if the session can put up with packets arriving out of order (e.g., the packets can be processed as independent datagrams), the destination does not have to bother with reassembly buffers, while the router would still try to "fix what ain't broke" and buffer the packets in its effort to provide for the unneeded in-order delivery. With deflection, the buffer space at the routers can be viewed as a shared pool of global buffers, and the entire network will automatically tend to distribute the buffering problem over underutilized nodes. As the rationale of networks is to provide for resource sharing among their applications, it makes perfect sense to try to employ the same principle to their internal operation.

3 The model

In our model, a VoIP source employs Pulse Code Modulation (PCM) coding with silence suppression. The traffic consists of interleaved talkspurts and silence periods, whose durations are exponentially distributed with the means of $1.2\,s$ and $1.8\,s$, respectively. The session duration is exponentially distributed as well, with the mean of 1 minute. When a session completes, another one is immediately started with a new selection of the source-destination nodes. The total number of active sessions is thus constant and represents the offered load experienced by the network. We look at the average behavior of all VoIP sessions over a reasonably long amount of time (e.g., 1 hour).

We assume that the 64 Kbps encoded speech signal is collected over approximately 20 ms and packetized into 160-byte data blocks. Since most VoIP systems use the RTP/UDP/IP stack [24], we add 40 bytes of overhead to each packet to account for the RTP(12)/UDP(8)/IP(20) headers.

For voice streams using silence suppression, it is common to set the playout time only at the start of a new talkspurt, thereby minimizing the impact of session delay variations on the voice quality. Our model permits us to adjust the *playout threshold*, i.e., the delay elapsing between the reception of the first packet of a talkspurt and the actual commencement of the playout. Subsequent packets of the same talkspurt are played at the same fixed rate that is independent of their arrival and mimics the rate at which they were generated at the source.

Each node is capable of acting as a router and a host at the same time, i.e., it can be a source and/or destination of a VoIP session. To express the buffer space tradeoff, we represent the total amount of buffer space in the network as $B + b$, where B denotes the total amount of space equally partitioned among the destinations (to be used for reassembly buffers) and b stands for the amount of storage equally partitioned among the output links of all routers. While each node implements both types of functionality, its router and host operations are isolated and they use two separate buffer pools. In our simulation experiments, the total amount of buffer space in the network, $B + b$ remains fixed in a given experimental setup, while the varying ratio B/b determines the adjustable balance between the two categories of storage.

Two basic routing models are considered. In the traditional single path model, packets are always forwarded along a single shortest path connecting the source to the destination. It is formally assumed that the path is selected at the source, i.e., the routers have no flexibility to change it. As a variant of traditional routing (and a representative of alternative path routing schemes [25]), we consider a simple strategy whereby the source is aware of a second-best route (disjoint from the first route at least on the first hop) and utilizes it whenever the first route appears congested.

With deflection routing, each router along the source-destination path is allowed to forward any packet over any of its output links. For this operation, the output links are ordered according to the length of the shortest path to the destination offered by the next hop node. The packet is queued for forwarding on the best (i.e., shortest path) link whose buffer is not full.

Our experimental setups involve regular network topologies (traditionally considered for deflection networks), as well as irregular topologies more akin to typical fragments of the Internet. The latter were obtained with assistance of the Network Manipulator *nem* [26], which generates suitably sized Internet-like graphs by extracting them from an available router level map of the Internet. These graphs are believed to have the statistical properties found in measured Internet maps [27].

For simplicity all links have the same bandwidth of 1 Mbps. In regular topologies, the links have an equivalent bandwidth$\times$delay capacity for 2 packets (around 640 km) and in irregular topologies the link capacity is determined by the random positions of the nodes generated by *nem*.

4 Results

4.1 Rectangular topology

The rectangular topology shown in Fig. 2 is similar to the bi-directional (toroidal) Manhattan Street Network [28], except for the boundaries, i.e., the absence of the links closing the rectangle onto a torus. In the experiments illustrated in Fig. 3(a), the playout threshold is set to half the reassembly buffer size. The figure shows the behavior of packet loss in the regular network as the allotment of buffer space (B/b) shifts from the routers to the destinations. The curve $k = 1$ correspond to the classic best effort routing where only one best path is available to each destination. If there is no sufficient space in the output buffer, the packet is discarded. $k = 2$ represents the case where the source has two distinct shortest paths to the destinations, while each internal (routing) node acts in exactly the same way as for $k = 1$. The source selects the second path if the buffer associated with the first (preferred) path is full.

With deflection, we see a clear trend of consistently decreasing loss rate as the buffer space is taken away from the routers and assigned to the destinations. This does not happen with the traditional circuit-style routing because, in absence of deflection, the routers are more desperate about storing the packets that cannot be immediately forwarded on the single congested output link.

One obvious problem with large reassembly buffers at destinations is the large playout delay, which, above a certain value, renders the overall session delay too high to be acceptable by the application. If we insist on the playout threshold to be proportional to the buffer size (as the buffer space is shifted from the routers to the destinations), then the increasing ratio B/b will unavoidably

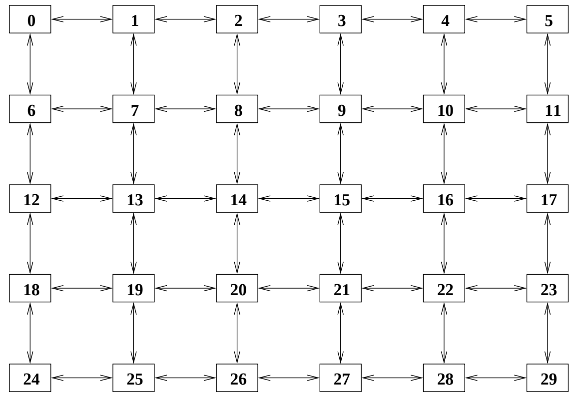


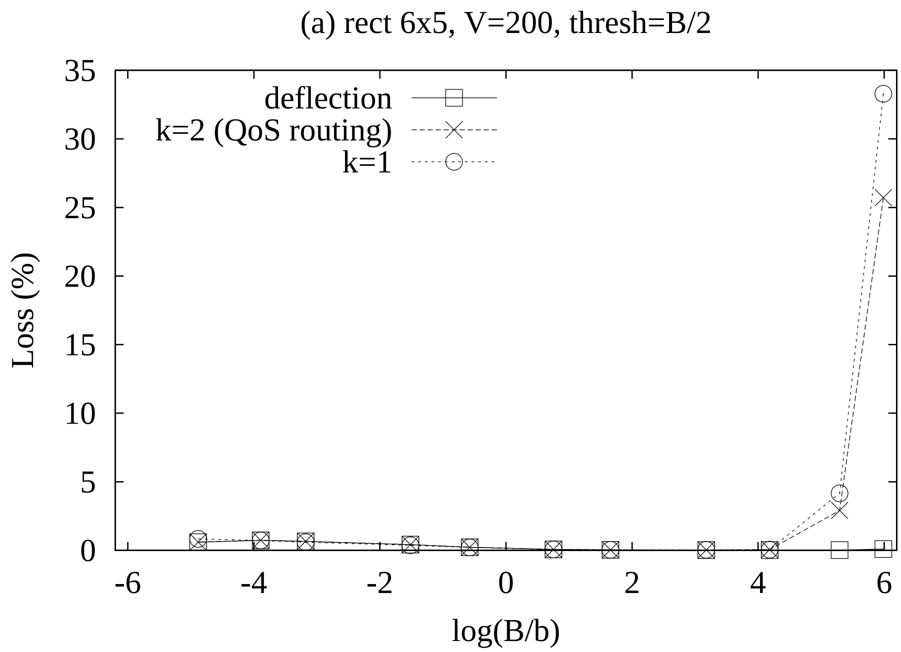
Figure 2: Rectangular topology

translate into proportionally increasing playout delays. As shown in Fig. 3(b), practically the entire delay perceived by the session comes from the playout buffers. Maintaining a reasonably bounded delay is an important QoS criterion, and it thus may make better sense to set the playout delay to a fixed value—large enough to keep the loss rate acceptable, and small enough to avoid excessive delays.

Somewhat surprisingly, it turns out that extending the playout delay at the destinations in proportion to the reduction of the buffer space at the routers is a gratuitous over-compensation. Consider Fig. 4(a), showing the percentage of sessions with acceptable QoS under the fixed playout delay of 4 packets, regardless of the actual size of the reassembly buffer at the destination. By "acceptable" in this case we mean a loss rate below 5%, which is tolerated by most audio decoders with no appreciable degradation in voice quality.

Figure 4 in fact demonstrates that the benefits (flexibility) of deflection more than compensate for the reduced buffer space at the routers. The apparent unpredictability of reception incurred by diversifying the routes of individual packets does not call for a significant increase in reassembly buffers, certainly not in proportion to the reduction of the buffer space at the routers. Note that in the area where the traditional routing completely breaks down, deflection is still able to provide reasonable service to a large population of sessions. Figure 4(b) shows the respective delays, which, for deflection, stay in an acceptable range for a typical VoIP application, with little buffer space at the routers. One can speculate that even lower end-to-end delays can be achieved with some form of packet delay prediction, e.g., as proposed in [29, 30].

Figure 5 illustrates what happens under a higher load (350 concurrent sessions compared to 200



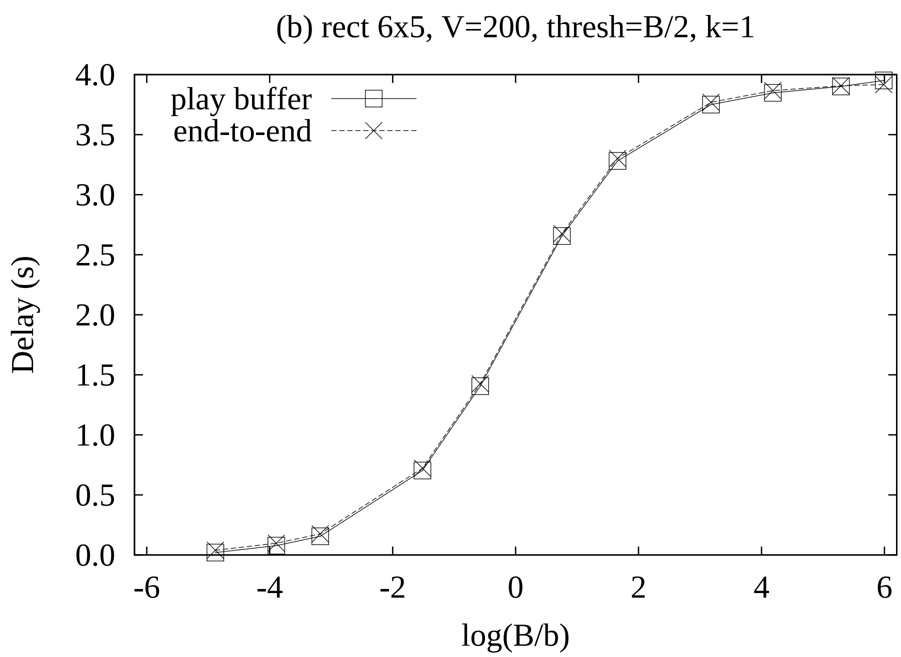
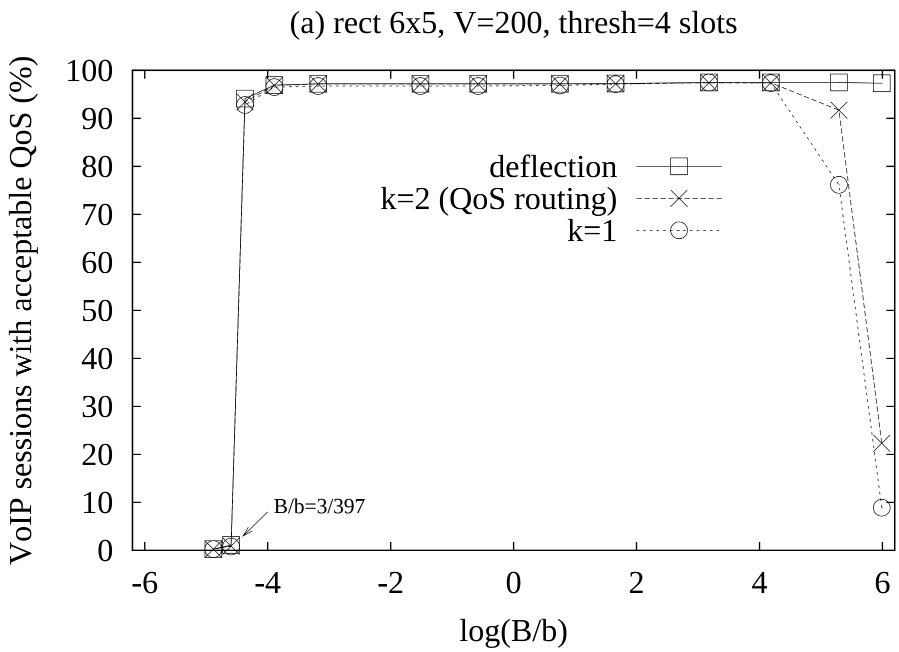


Figure 3: Rectangular topology with threshold equal to half the reassembly buffer size: (a) loss rate and (b) delay



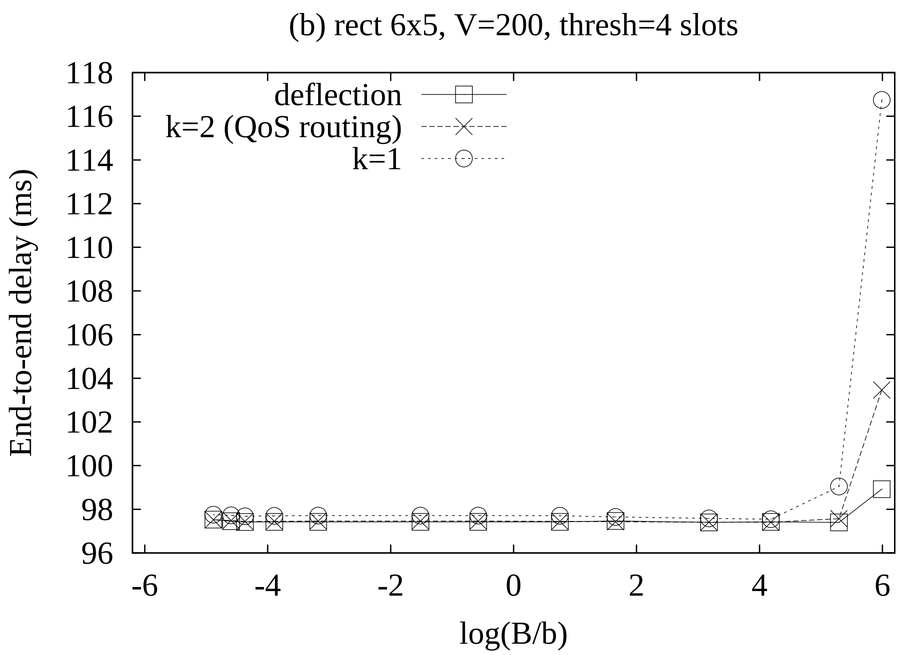
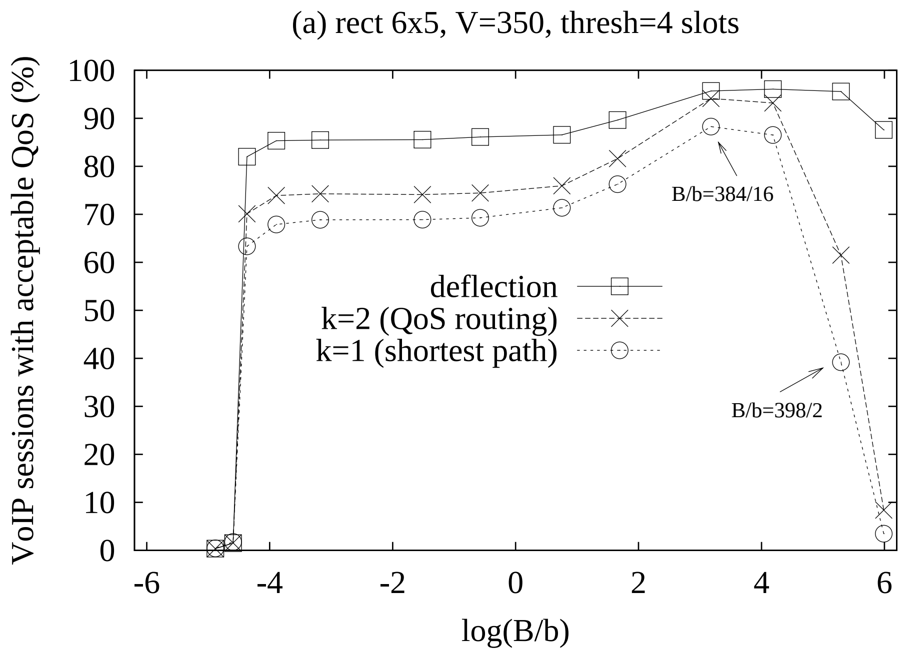


Figure 4: Rectangular topology with 200 simultaneous VoIP sessions and threshold set to 4 slots: (a) fraction of VoIP sessions with acceptable QoS and (b) end-to-end delay

in Fig. 4). In terms of the percentage of acceptable sessions, deflection turns out to consistently outperform the traditional routing. Even though the end-to-end delay with deflection tends to be slightly higher than for $k = 2$ (with short buffers at the routers), the total number of sessions with acceptable loss rate is consistently higher in the deflection case. We can also see that the best buffer allotment for classic routing is around $B/b = 384/16$, while it is $398/2$ for deflection. In agreement with the observations made in [21, 17, 18], deflection can take advantage of some rudimentary buffer space at the routers. In comparison to the completely buffer-less (synchronous) case, small buffers bring about a considerable improvement, while large buffers yield no further appreciable gains.

Using the best B/b ratios found in the earlier experiments, Fig. 6 demonstrates the impact of offered load on the QoS of VoIP sessions. Once the number of VoIP sessions exceeds 250, the classic shortest path routing clearly offers worse QoS than deflection. Deflection, on the other hand, provides sufficient QoS even at higher loads, reaching approximately 400 sessions before the QoS quickly deteriorates. We note that for $k = 1$ most packets are lost in the network due to buffer overflows, before they reach the destination. In the case of deflection all packets arrive at the destination, but a large fraction arrives too late. Note that this fraction of late packets is in essence under the application's control. Indeed, as can be seen in Fig. 7, if the late threshold is increased to 20 slots, deflection routing offers better QoS than classic routing also under high loads. The proper way to interpret this is that in classic routing, the nature of most losses (due to buffer overflows) leaves little control to the application to adjust to the congestion conditions, while deflection makes this issue more flexible by transforming congestion into delays instead of losses.

To confront the perfromance of deflection with the best performance attainable by QoS routing, under mechanisms more sophisticated than those solely based on the path length minimization criteria, we also considered the Widest-K-Shortest (WKS) algorithm fully described in [25]. This algorithm selects the K-shortest paths from each source to destination and re-ranks them each time new link state information is received from neighbors. Link state (LS) exchange is the dominant cos factor of QoS routing [3]. Therefore, Probabilistic Frequency Reduction (PFR) [25] is used to reduce update costs without significantly impairing routing performance. Using PFR, the LS information is only exchanged with the neighbors with a certain probability (FC). Figure 8 shows that the performance of QoS routing can be somewhat improved by exchanging LS information among nodes; however, 1) the time intervals of the exchange must be significantly shorter than the duration of



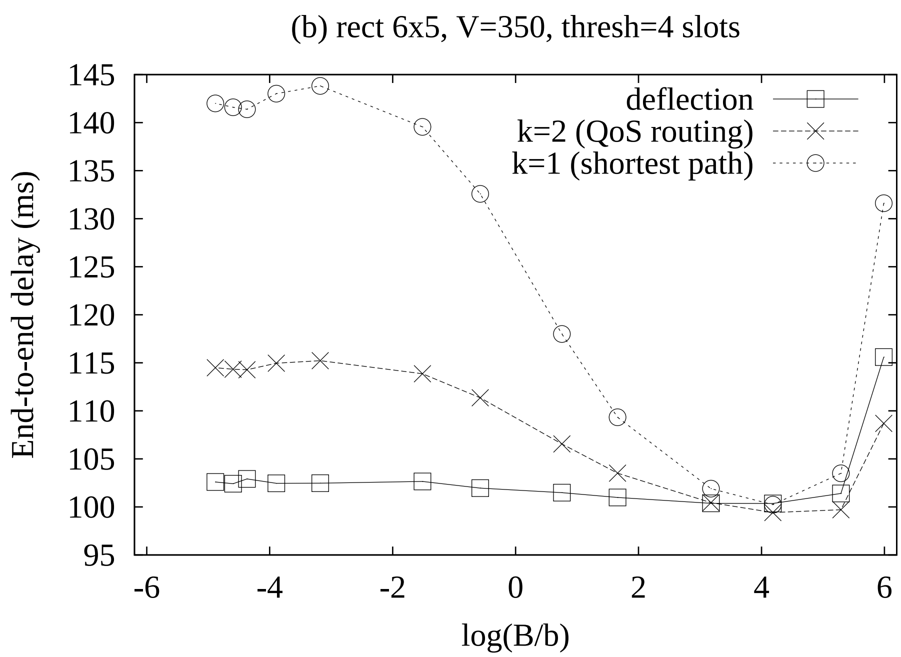


Figure 5: Rectangular topology with 350 simultaneous VoIP sessions and threshold set to 4 slots: (a) fraction of VoIP sessions with acceptable QoS and (b) end-to-end delay

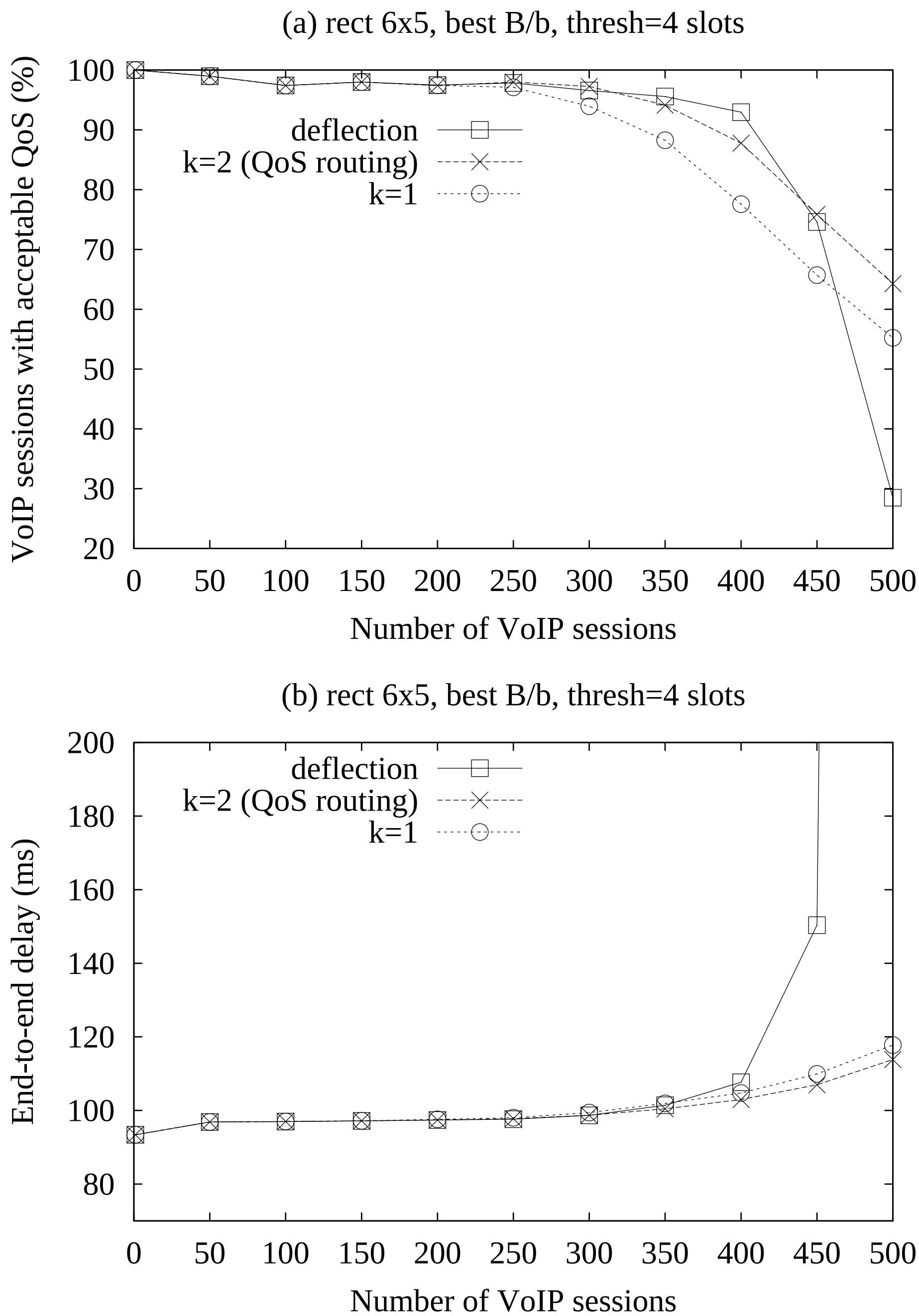


Figure 6: Rectangular topology with threshold set to 4 slots: (a) fraction of VoIP sessions with acceptable QoS and (b) end-to-end delay

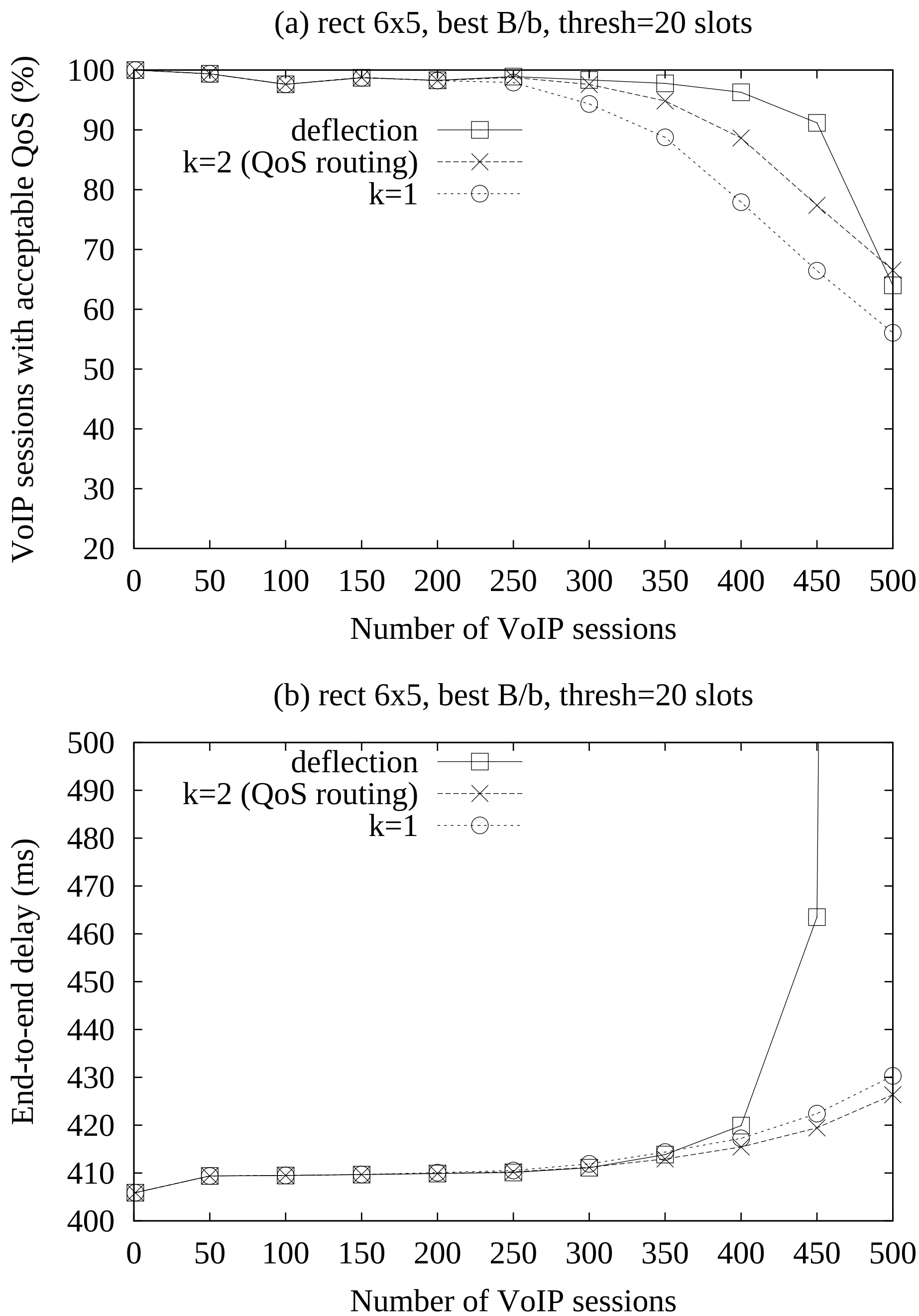


Figure 7: Rectangular topology with threshold set to 20 slots: (a) fraction of VoIP sessions with acceptable QoS and (b) end-to-end delay

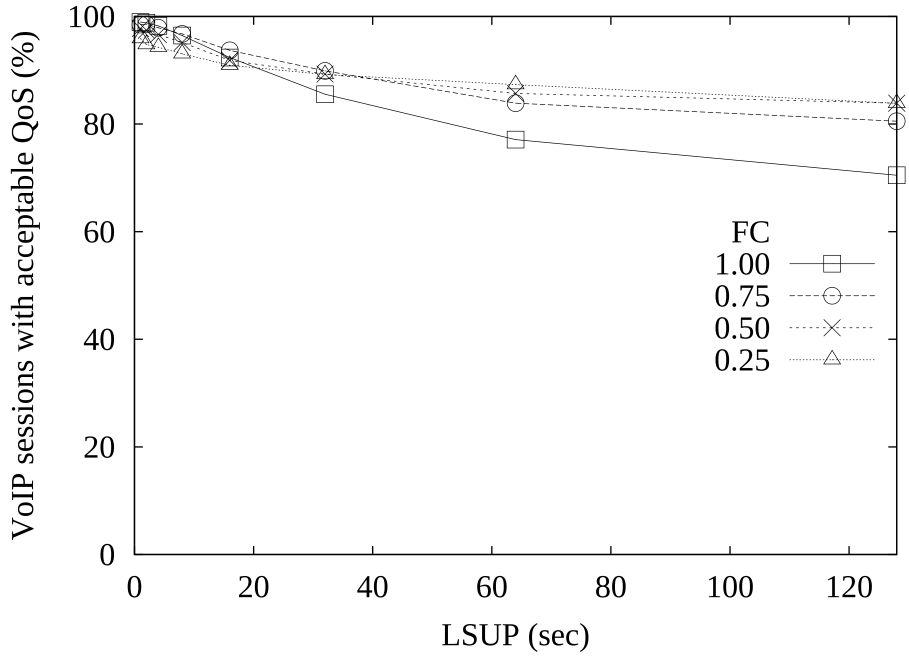


Figure 8: Fraction of VoIP sessions with acceptable QoS versus LSUP for different Frequency Coefficients (FC) on the rectangular topology

an average VoIP session (60 sec), 2) deflection is capable of achieving better improvements without having to pay the price of the LS update overhead.

4.2 Triangular topology

For the sake of exposition we review the performance of deflection vs. classic routing in the "triangle" topology of Fig. 9. The topology, while far from being arbitrary or "random," illustrates the performance in a network where we encounter both "hotspots" (the "bottleneck" nodes connecting two triangles, of degree four) as well as alternative paths (the triangles). As such, it helps illustrate how even a small network, e.g., one falling under the administrative control of a network provider, can benefit from deflection. Under light traffic, all three routing algorithms have similar loss rates (not shown). When the load is higher, deflection routing shows lower loss rates with small router buffers, as can be seen in Fig. 10, for an offered load of 100 VoIP sessions. Indeed, in the particular topology the best (shortest) routes for all nodes pass through the even-numbered nodes, and when the links between these nodes become congested, they discard packets under the classic routing regime. With deflection, the links between even and odd-numbered nodes are automatically used

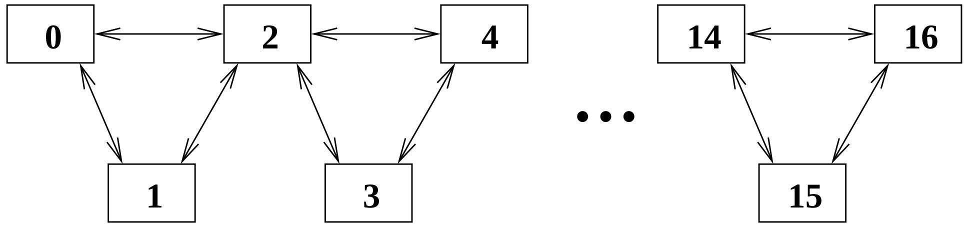


Figure 9: Triangular topology

to distribute the load, which results in a higher throughput and, therefore, lower losses. At higher loads the effect is even more pronounced.

Figure 11 shows that the performance of QoS routing in the triangular network improves under LS exchange, as the scheme is able to exploit the benefits of the available alternative paths.

4.3 Irregular topologies

Finally, the "proof of the pudding" is whether similar patterns of performance behavior tend to appear in random irregular topologies. We used *nem* to generate several irregular topologies with sizes ranging from 35 to 750 nodes, with all link lengths calculated from the (random) node positions generated by *nem*. The results plotted in Fig. 12(a) show the fraction of VoIP channels with acceptable QoS for a 35 node network with 120 simultaneous VoIP sessions using the 4-slot playout threshold. The delay for the same topology is shown in Fig. 12(b). The observed QoS was very similar to what was seen with regular topologies. Namely, it is still the case that moving buffers from core to periphery improves the number of sessions that can be carried out, but only under the deflection regime. The delay is also improved in the case of deflection. The reason behind the observed results can be traced to the fact that in the irregular topologies, not many alternative paths exist between source and destination. Indeed although the irregular topologies generated by *nem* have an average degree of 2.6, this value is manifested through a few nodes with a very high degree and many nodes with a much smaller degree, including numerous nodes with the degree of one. Clearly, shortest path routing quickly saturates the best paths but "second–best" paths rarely exist as the $k = 2$ curves clearly indicate. It is the deflection's capability of utilizing (even if just transiently) an underutilized link on an as–needed basis which provides performance dividends even when true path alternatives do not exist.

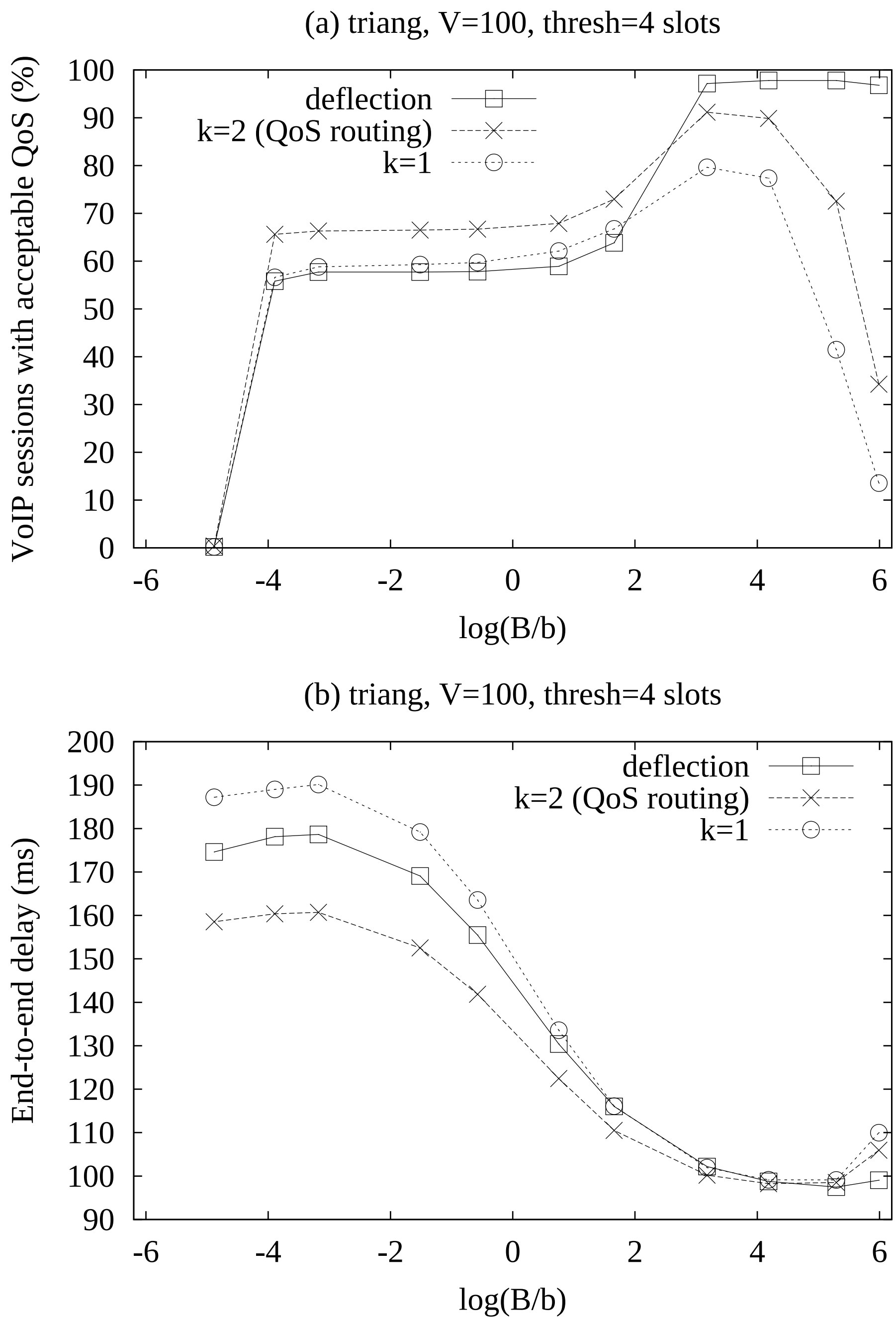


Figure 10: Results for triangular topology: (a) fraction of VoIP channels with acceptable QoS and (b) end-to-end delay

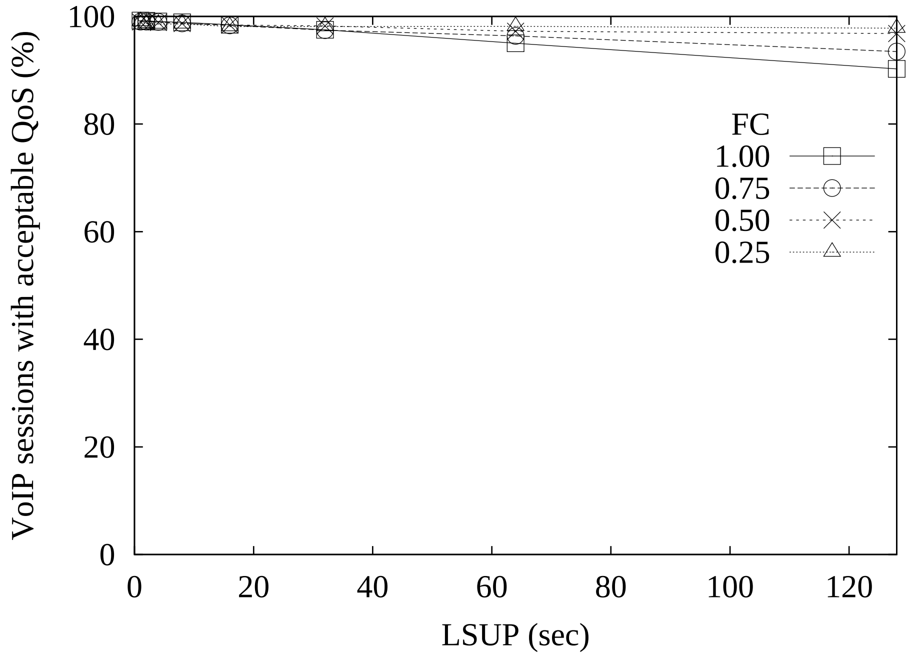


Figure 11: Fraction of VoIP sessions with acceptable QoS versus LSUP for different Frequency Coefficients (FC) on the triangular topology

5 Conclusions

We have studied the quality of service perceived by concurrent VoIP sessions in networks operating under three routing policies: 1) classic shortest path best effort routing, 2) QoS routing with two alternative shortest paths, and 3) asynchronous deflection with small buffers. Our experiments show that, especially under high loads, deflection performs at least as well as the other two alternatives, and frequently outperforms them with respect to the overall QoS measures perceived by the application. The better load distribution over the network brought about by deflection translates into lower losses and end-to-end delays.

We have argued that, contrary to the common belief, the deflection paradigm is not necessarily inferior (from the viewpoint of service guarantees) to the more deterministic alternatives involving pre-arranged predictable routes. Despite the difference in appearance, neither paradigm is in fact deterministic and predictable, and the application-level perception of its operation does not necessarily endorse determinism within the core.

Deflection appears to offer more flexibility within the network layer regarding contention reso-

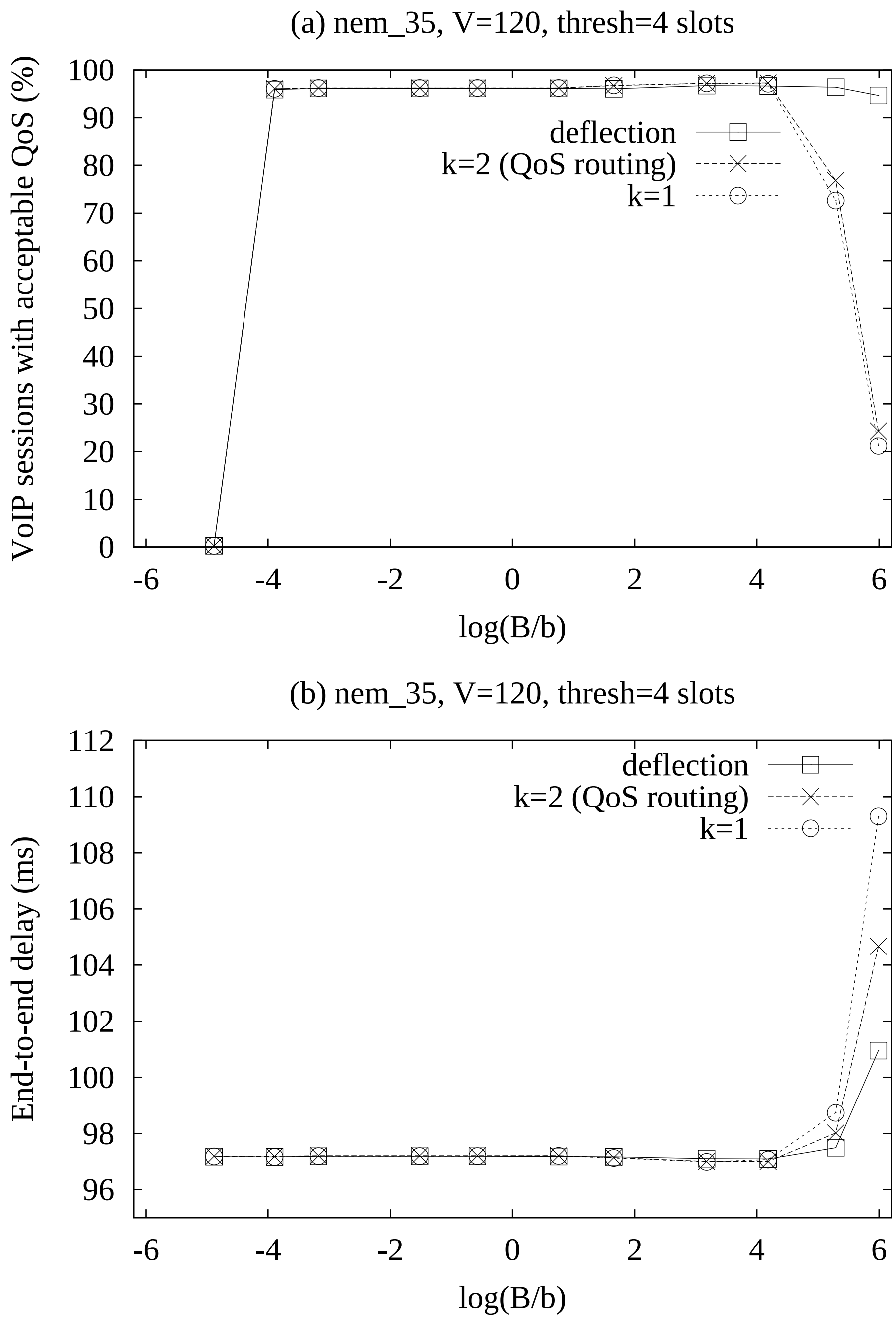


Figure 12: Results for an irregular topology with 35 nodes: (a) fraction of VoIP channels with acceptable QoS and (b) end-to-end delay

lution and load balancing—because of its immediate and automatic responsiveness to intermittent overload conditions. This flexibility extends over the utilization of buffers in the network core: the buffer space becomes effectively shared and, together with the underutilized links, becomes a global means of accommodating locally excessive traffic. Even if the directly measurable benefits of deflection appear somewhat diluted in more realistic irregular networks (because of the reduced number of alternative paths), one advantage that still remains is the reduction of the buffer space at the routers and its more economical utilization as a global network resource.

References

[1] P. P. Mishra and H. Saran, "Capacity management and routing policies for voice over IP traffic," *IEEE Network*, vol. 14, no. 2, pp. 20–27, 2000.

[2] B. Davie and Y. Rekhter, *MPLS Technology and Applications.* Morgan Kaufmann Publishers, 2002.

[3] G. Apostolopoulos, R. Guerin, S. Kamat, and S. K. Tripathi, "Quality of service based routing: a performance perspective," in *Proceedings of the ACM SIGCOMM '98 conference on Applications, technologies, architectures, and protocols for computer communication*, pp. 17–28, ACM Press, 1998.

[4] Y. Jia, I. Nikolaidis, and P. Gburzynski, "Multiple path routing in networks with inaccurate link state information," in *Proceedings of IEEE International Conference on Communications (ICC'2001)*, vol. 8, pp. 2583–2587, June 2001.

[5] X. Yuan and A. Saifee, "Path selection methods for localized quality of service routing," in *Proceedings of The 10th International Conference on Computer Communications and Networks (ICCCN'01)*, pp. 102–107, October 2001.

[6] W. Xiao, B. H. Soong, C. L. Law, and Y. L. Guan, "Evaluation of heuristic path selection algorithms for multi-constrained qos routing," in *IEEE International Conference on Networking, Sensing and Control*, vol. 1, pp. 112–116, March 2004.

[7] G. Feng, "On the performance of heuristic H_MCOP for multi-constrained optimal-path qos routing," in *18th International Conference on Advanced Information Networking and Applications (AINA 2004)*, vol. 2, (Fukuoka, Japan), pp. 50–53, March 2004.

[8] G. Appenzeller, I. Keslassy, and N. McKeown, "Sizing router buffers," in *ACM SIGCOMM'04*, (Portland, Oregon), September 2004.

[9] N. Maxemchuk, "The Manhattan street network," in *Proceedings of GLOBECOM'85*, pp. 255–261, 1985.

[10] N. Maxemchuk, "Routing in the Manhattan Street Network," *IEEE Transactions on Communications*, vol. 35, pp. 503–512, May 1987.

[11] N. Maxemchuk, "Comparison of deflection and store-and-forward techniques in Manhattan-street network and shuffle-exchange networks," in *Proceedings of IEEE INFOCOM'89*, pp. 800–809, 1989.

[12] N. Maxemchuk, "Problems arising from deflection routing," in *High Capacity Local and Metropolitan Networks* (Pugolle, ed.), pp. 209–233, Springer Verlag, 1991.

[13] N. Maxemchuk and R. Krishnan, "A comparison of linear and mesh topologies–DQDB and the manhattan street network," *IEEE Journal on Selected Areas in Communications*, vol. 11, pp. 1278–1301, Oct. 1993.

[14] N. Maxemchuk, "Twelve random access strategies for fiber optic networks," *IEEE Transactions on Communications*, vol. 36, pp. 942–950, Aug. 1988.

[15] A. Krishna and B. Hajek, "Performance of shuffle-like switching networks with deflection," in *Proceedings of IEEE INFOCOM'90*, (San Francisco, CA), pp. 473–480, June 1990.

[16] H.-Y. Huang, T. Robertazzi, and A. Lazar, "A comparison of information based deflection strategies," *Computer Networks and ISDN Systems*, vol. 27, pp. 1399–1407, 1995.

[17] P. Gburzyński and J. Maitan, "Deflection routing in regular MNA topologies," *Journal of High Speed Networks*, vol. 2, no. 2, pp. 99–131, 1993.

[18] W. Dobosiewicz and P. Gburzyński, "A bounded-hop-count deflection scheme for Manhattan-street networks," in *Proceedings of IEEE INFOCOM'96*, (San Francisco), pp. 172–179, Mar. 1996.

[19] W. Olesinski and P. Gburzyński, "Real-time traffic in deflection networks," in *Proceedings of WMC'98: Communication Networks and Distributed Systems, Modeling and Simulation*, (San Diego, California), pp. 23–28, Jan. 1998.

[20] W. Olesinski and P. Gburzyński, "Service guarantees in deflection networks," in *Proceedings of MASCOTS'01*, (Cincinnati, Ohio), pp. 267–274, Aug. 2001.

[21] A. Choudhury and N. Maxemchuk, "Effect of a finite reassembly buffer on the performance of deflection routing," *Conference Record of the International Conference on Communications (ICC)*, vol. 3, pp. 1637–1646, 1991.

[22] F. Borgonovo and L. Fratta, "Deflection networks: architectures for metropolitan and wide area networks," *Computer Networks and ISDN Systems*, no. 24, pp. 171–183, 1992.

[23] C. Baransel, W. Dobosiewicz, and P. Gburzyński, "Routing in multi-hop switching networks: Gbps challenge," *IEEE Network Magazine*, no. 3, pp. 38–61, 1995.

[24] B. Goode, "Voice over internet protocol (VoIP)," in *Proceedings of the IEEE*, vol. 90, pp. 1495–1517, September 2002.

[25] Y. Jia, I. Nikolaidis, and P. Gburzynski, "Scalable QoS routing using alternative paths," *International Journal of Communication Systems*, vol. 17, no. 1, pp. 1–26, 2004.

[26] D. Magoni and J.-J. Pansiot, "Internet topology modeler based on map sampling," in *Proceedings of the Seventh International Symposium on Computers and Communications (ISCC'02)*, p. 1021, IEEE Computer Society, 2002.

[27] D. Magoni and J.-J. Pansiot, "Evaluation of internet topology generators by power law and distance indicators," in *10th IEEE International Conference on Networks. ICON 2002*, pp. 401–406, August 2002.

[28] F. Borgonovo and E. Cadorin, "Locally-optimal deflection routing in the bidirectional Manhattan network," in *Proceedings of IEEE INFOCOM'90*, pp. 458–464, 1990.

[29] P. DeLeon and C. Sreenan, "An adaptive predictor for media playout buffering," in *IEEE International Conference on Acoustics, Speech, and Signal Processing. ICASSP'99*, vol. 6, pp. 3097–3100, March 1999.

[30] A. Shallwani and P. Kabal, "An adaptive playout algorithm with delay spike detection for real-time VoIP," in *Canadian Conference on Electrical and Computer Engineering. IEEE CCECE 2003*, vol. 2, pp. 997–1000, May 2003.