ARCHITECTURAL CHOICES FOR PACKET SWITCHED TELEPHONE NETWORKS

Mario Baldi and Davide Bergamasco
Politecnico di Torino — ITALY

Eugenio Guarene
Centro Studi e Laboratori Telecomunicazioni (CSELT) — ITALY

Abstract

Today, Internet telephony is a reality but this cannot be regarded as a true telephone service because its quality is highly variable depending on network load. The end-to-end delay (i.e., the time between when the speaker talks and when the listener hears) is a major quality index for telephone services. This delay must be shorter than 100-200 ms in order to allow interaction among users. We refer to this bound as the interaction bound.

This work has two main goals:

- *study the end-to-end delay of voice transmission over a packet switched network from an analytical point of view, and*
- *identify some possible architectural solutions for packet switched networks expressly designed to carry telephone traffic (we call them packet switched telephone networks).*

We first identify the delay components and the main factors in determining their upper bound. This analysis shows that the current Internet architecture is not suited for providing telephone services with end-to-end delay shorter than the interaction bound.

Then we analyze the end-to-end delay more deeply to identify the key factors in the design of a packet switched telephone network providing commercial quality telephone services, i.e., for possibly replacing traditional telephone networks. We show that by limiting the number of hops in the path between any pair of users and dimensioning link capacity accordingly, the end-to-end delay can be kept below the interaction bound. Thus, packet switching could be exploited to provide a high quality telephone service even though the raw capacity needed is larger than with circuit switching. This is justified by the possibility of carrying best effort traffic on the same network and by the lower installation and management costs.

In this work we consider two enabling technologies for a packet switched telephone network: the Asynchronous Transfer Mode (ATM) and the Internet Protocol (IP). Our analytical model shows that ATM outperforms IP as was expected since the former was expressly designed to support low bandwidth real-time traffic. Nevertheless, because of the large number of existing applications based on IP and its internetworking capabilities, it must be taken into account as a candidate technology for carrying both best effort and high bandwidth real-time multimedia traffic in packet switched telephone networks.

We propose integrated services IP/ATM models to exploit both IP and ATM in the implementation of packet switched networks for telephony. These models are derived from the integrated IP/ATM approaches currently being exploited for an effective operation of IP over ATM networks. Integrated services IP/ATM models provide ATM based services for telephony and IP (over ATM) based services for any other kind of traffic (namely, best effort and multimedia) in order to get the best from the two technologies.

1 Introduction

A big deal of emphasis is being put on Internet telephony. Even though it is possible to transmit voice over the Internet, this cannot be regarded as a telephone service because quality is generally low. In fact, depending on the network load along the communication path, quality can vary widely.

A major quality index in *interactive* voice services is the delay perceived by the users. In order for a conversation not to be annoying, the *end-to-end delay* (i.e., the time between when the speaker talks and when the listener hears) must be shorter than 100-200 ms. We call this maximum acceptable delay the *interaction bound*. If the voice service is used for interactive communication, the end-to-end delay is required to be always below the interaction bound.

This work has two main goals:

- study the end-to-end delay of voice transmission over a packet switched network[1] from an analytical point of view, and
- identify some possible architectural solutions for packet switched networks expressly designed to carry telephone traffic. We call such networks *Packet switched Telephone Networks* (PTNs).

In Section 2 we identify the delay components and the main factors in determining the delay bound. This first analysis shows that the current Internet architecture is not suited to the provision of telephone services with end-to-end delay shorter than the interaction bound.

In Section 3 we analyze the end-to-end delay more deeply and identify the key factors in the design of a PTN conceived for providing commercial quality telephone services, i.e., for possibly replacing traditional telephone networks. This is attractive because unused capacity can be exploited to carry data traffic, and network deployment and maintenance are easier since the network itself can be used for management purposes.

Section 4 shows some numerical results obtained by applying the analytical equations devised in Section 3 to a PTN designed after the Telecom Italia's telephone network (i.e., using the same topology and number of supported calls). Various network configurations, which differ for enabling technology, link capacity, and speech encoding techniques, are compared, showing that some of them guarantee the interaction bound. In this work we consider two enabling technologies for a PTN: the *Asynchronous Transfer Mode* (ATM) and the *Internet Protocol* (IP).

ATM was standardized by the ITU-T as the foundation of the *Broadband - Integrated Services Digital Network* (B-ISDN)] and was explicitly designed for the provision of real-time services while taking advantage of statistical multiplexing for an efficient usage of network resources.

On the other side, the Internet protocol suite is currently the most widely deployed network architecture. IP was designed as an internetworking protocol for carrying best effort traffic among heterogeneous networks and lots of applications that use its services have been developed. The Internet community is actively working on the *Integrated Services Internet*], an Internet architectural evolution aiming at providing real-time services over IP. This will be fundamental to the videoconferencing and telephony applications which are being deployed over the Internet.

We claim that ATM and IP together can be successfully exploited for devising service integration over a PTN. IP is desirable for its broad diffusion and the large number of existing applications. Moreover, due to its internetworking capabilities, it allows systems to communicate through heterogeneous technologies. When deployed together with mechanisms for guaranteeing *Quality of Service* (QoS), it is also suited to real-time applications which require both high bandwidth and low latency like videoconferencing. On the other side, ATM has been designed to provide QoS guarantees, but it requires all the network to be implemented with the same technology and all the existing applications to be rewritten in order to take advantage of its capabilities. Nevertheless, ATM outperforms IP when dealing with telephony, i.e., low bandwidth real-time applications, as reported in Section 4.

In Section 5 various integrated IP/ATM architectural models for the exploitation of IP and ATM in PTNs are described, while conclusions are drawn in Section 6.

2 Delay of Voice Transmission over Packet Switched Networks

The end-to-end delay in the transmission of voice over a packet switched network has three components:

1. The *processing* delay D_{proc} is introduced when processing the audio signal. Digital transmission of voice requires the audio signal to be sampled 8000 times per second. Samples are quantized, encoded, and transmitted to the receiver which plays them at a fixed pace. The *Pulse Code Modulation* (PCM) encoding, which is used on traditional digital telephone networks, uses 8 bits to encode each sample, thus generating a 64 kb/s flow for each voice connection. The processing delay introduced by PCM is negligible. The voice signal can be encoded with different techniques in order to produce a bit stream at rates as low as 8 kb/s with audio quality comparable to PCM]. Some of these techniques introduce a delay up to 100 ms], thus not being suited to interactive telephone services.

2. The *network* delay D_{net} is given by the time to inject into and propagate through the network the data stream. It has five components:

i. The voice encoder produces a bit stream at a rate of R bits per second. Before being transmitted through the network, bits are clustered in packets. In each packet P bits are stuffed in the payload, this clustering introduces a *packetization* delay

$$D_{pkt} = \frac{P}{R} \qquad (1)$$

ii. The *transmission* delay

$$D_{tr} = \frac{P_s}{C}$$

is introduced to send a packet of size P_s over a link having capacity C.

iii. D_{pr} is the delay due to signal *propagation* through the physical links connecting network nodes.

ii. The *node processing* delay D_{np} is introduced by a network node each time it has to forward a packet.

iii. The *queuing* delay is the time spent by packets in nodes' buffers while contending for the same output port. It is a relevant component in determining the end-to-end delay of a voice connection. The queuing delay Q_i experienced by a packet in the i^{th} node on the route to the destination, depends on the instantaneous status of the output buffer associated to the link on which the packet must be forwarded. As a consequence the queuing delay is not known a priori; whenever a best effort service is provided (as currently is by the Internet) a bound is given by

$$0 \le Q_i \le \frac{B_i - P_s}{C_i}, \qquad (2)$$

where B_i is the dimension (in bits) of the buffer and C_i is the capacity of the output link[2].

3. A good quality reconstruction of the voice signal at the receiver requires samples to be played at regular intervals (e.g., every 125 ms). Whenever the two previous delay components are variable in time (i.e., the end-to-end delay presents a *jitter*),

different samples experience different end-to-end delays. If they were played as soon as they are available to the receiving application, they would not be uniformly spaced (in time). Thus, the receiving application uses a *replay buffer* to store the samples that experience a delay shorter than the maximum: it retrieves and plays the samples at a regular pace. The *compensation* delay, i.e., the time spent by early samples in the replay buffer, should be between 0 and $d_{RB} = DD_{proc} + DD_{net}$, where DD_{proc} and DD_{net} are the maximum[3] variations of the processing (item 1 above) and network (item 2) delays, respectively. Actually, since the delay experienced by samples when entering the replay buffer is not known (see] for details), the compensation delay is bound by $2 \cdot d_{RB}$. Thus, the replay buffer also introduces the *excess compensation* delay $E_c = [0, d_{RB}]$.

Due to the jitter compensation, the delay perceived by the user is

$$D_{ete} = \max(D_{proc}) + \max(D_{net}) + E_c$$

which can be rewritten as

$$D_{ete} = D_{proc} + D_{pkt} + (N + 1) \cdot D_{tr} + D_{pr} + N \cdot D_{np} + Q_{max} + E_c \qquad (3)$$

where N is the number of network nodes on the path between sender and receiver and Q_{max} is the maximum queuing delay given by

$$Q_{max} = \sum_{i=1}^{N} \max Q_i = \sum_{i=1}^{N} \frac{B_i - P_s}{C_i} \qquad (4)$$

Equations and show that the end-to-end delay strongly depends on the number of network nodes on the path between sender and receiver. As it is shown by the numerical example in Section 4, queuing delay is the main contribution to the end-to-end delay[4].

The current architecture of the Internet has been designed to carry data traffic with a *best effort* service. In order to limit loss due to congestion in network nodes, buffers are large, thus leading to a large Q_{max}. Moreover, the number of hops on the path between sender and receiver can be very large (even some tens). Many applications for voice transmission over Internet heuristically choose the compensation delay and possibly adapt it to the actual delay experienced by samples. As a result the quality of a voice call is unpredictable: when the nodes on the communication path are lightly loaded the quality is acceptable. Instead, during heavy traffic periods, either the received signal is not intelligible or the end-to-end delay is long (up to 1 second)].

Nevertheless, we believe that a voice transmission service with a commercial like guaranteed quality can be provided over a packet switched network provided that (1) it exploits some mechanisms for supplying QoS guarantees, and (2) its topology be designed so that the path of any voice connection (i.e., phone call) encompasses few nodes. In the following section we identify the key parameters for dimensioning such a network.

3 Designing a Packet Switched Telephone Network

Equation shows that queuing delay depends on the ratio between buffer size and link capacity. Thus, the end-to-end delay can be decreased by increasing link capacity while keeping buffer dimension fixed. Nevertheless, the probability of losses due to congestion in network nodes is high if buffers are small with respect to link capacity. The problem can be overcome by creating separate queues for best effort and voice traffic, the latter being given higher *priority* than the former. In this scenario, Equation still holds, B_i being the size of the voice queue alone.

The voice queue can be made small and losses avoided by introducing a *call acceptance control* mechanism, which limits the maximum number M_i of voice connections routed through a link. This determines the maximum number of packets present in the voice queue to

$$M_i - \left\lceil \frac{M_i}{I_i} \right\rceil \qquad (5)$$

where I_i is the number of input ports of the i^{th} node. The voice queue is dimensioned so that it can contain the above amount of packets thus leading to a maximum queuing delay. Actually, due to the variation of delay experienced in upstream buffers, more packets than the number expressed by Equation can be present in a queue. Nevertheless a maximum total queuing delay Q_{max}, as given by Equation , is not exceeded. Details can be found in].

$$Q_{max} = P_s \cdot \sum_{i=1}^{N} \frac{\left(M_i - \left\lceil \frac{M_i}{I_i} \right\rceil \right)}{C_i}. \qquad (6)$$

Equation shows that given the capacity of a link, the less phone calls are allowed on it, the smaller the maximum queuing delay of the connections traversing the link. The overall voice traffic routed on a link accounts only for a fraction of the link capacity, leaving to best effort traffic the remaining capacity. In order to provide an indication of the fraction of link capacity dedicated to voice traffic, we define the *voice allocation factor* a_i on link *i*, so that

$$\eta_{pkt} = \frac{P}{P_s} \qquad (7)$$

where P is the number of bits sent in each packet[5] and R is the rate of the voice stream, i.e., P/R is the time elapsed between two subsequent packets of the same voice connection. Since P_s is the size of packets, $(P_s \cdot R)/P$ is the (gross) bandwidth used by the connection, taking into account the overhead due to packet headers. Assuming that all the phone calls on the network exploit the same coding (same bit rate R) and the same packetization technique (same P_s and P), the second term of Equation is the bandwidth used by telephone traffic on link *i*.

We define the *packetization efficiency* as

$$\eta_{pkt} = \frac{P}{P_s} \qquad (8)$$

and use it to rewrite Equation :

$$\alpha_i \cdot C_i = M_i \frac{R}{\eta_{pkt}}$$

The packetization efficiency shows the fraction of link capacity wasted due to packet overheads (i.e., the header).

When dealing with variable size packets having fixed length header (e.g., in IP networks), the packetization efficiency is particularly relevant to the end-to-end delay. Increasing h_{pkt} requires packets to be enlarged, thus increasing some of the delay components. This can be made explicit by writing the packet size P_s in terms of the number *OH* of overhead bits

$$P_s = \frac{OH}{1-\eta_{pkt}}$$

and substituting it in Equation thus obtaining

$$Q_{\max} = \frac{OH}{1-\eta_{pkt}} \cdot \sum_{i=1}^{N} \frac{\left(M_i - \left\lceil \frac{M_i}{l_i} \right\rceil \right)}{C_i} \quad (9)$$

4 Numerical Results

In this section we consider the implementation of a PTN and provide some numerical results obtained by applying to it the equations devised in the previous section. The network topology is designed after the Telecom Italia's telephone network exploiting both IP and ATM technology.

In a circuit switched telephone network, telephones are connected through the local loop to local exchange offices which both have concentration functionality and encode analog voice signals into 64 kb/s PCM flows. Local exchange offices are connected through digital links to local offices that are circuit switching nodes. In metropolitan areas, local offices are connected together and with toll offices which build up a higher layer of switching offices connected to each other, as shown in .

The Telecom Italia's telephone network serves about 28 million users connected to about 11,000 local exchange offices which are, in turn, connected to 660 local offices. Local offices are fully meshed within metropolitan areas and each one is connected to at least two toll offices. The higher hierarchical level counts 60 toll offices connected in a full mesh. Finally, 6 international gateways connected to a number of toll offices provide international connectivity. The end-to-end delay of any call on the considered circuit switched telephone network, unless satellite hops are present along the communication path, is well below the interaction bound (usually shorter than 20 ms).

We analyze the delay perceived by users of the PTN obtained by substituting switching offices with packet switching nodes (IP routers or ATM switches). Each local exchange office is connected to a packet switching node. We assume that local exchange offices are the initiators of the call acceptance control procedures when users dial calls. Moreover, they are responsible for the voice packetization process. The capacity of the link between two nodes is determined according to the number M_i of telephone calls routed on the corresponding link of the real network, i.e.,

$$C_i = M_i \cdot n \cdot 64\ \text{kb/s}, \quad (10)$$

where 64 kb/s is the bandwidth used for each phone call and *n* is a factor used to oversize the link capacity in the PTN. Given the meshing of the Telecom Italia's network and the capacity of the links between nodes as obtained from Equation , a packet switching node is required to have between 80 and 150 SDH STM-1interfaces[6].

When computing the queuing delay according to Equations and (for an ATM and IP network, respectively), we introduce an approximation by considering that all the hops on the path give the same contribution (i.e., we assume that ($_iQ_i = N \cdot Q$). This approximation is not a significant one since queuing delay basically depends on C_i/M_i (see] for more details) which is constant because link capacity is dimensioned according to Equation [7]. The value used for M_i is the maximum number of circuits on the link between two toll offices, i.e., M_i = 2,000.

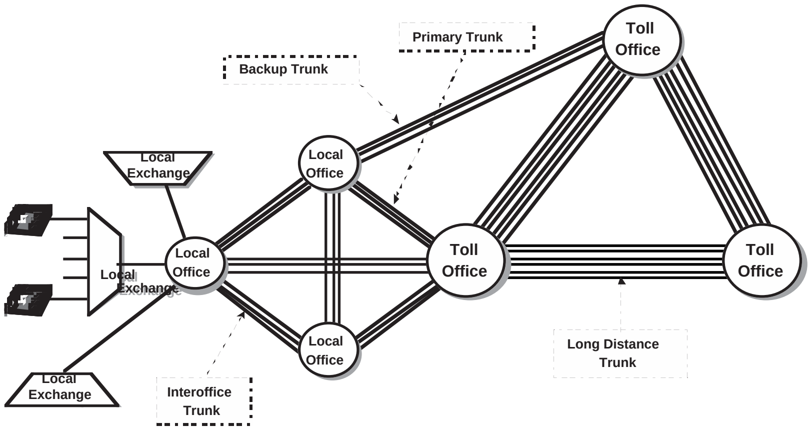


Figure 1: Excerpt from the Topology of a Circuit Switched Telephone Network.

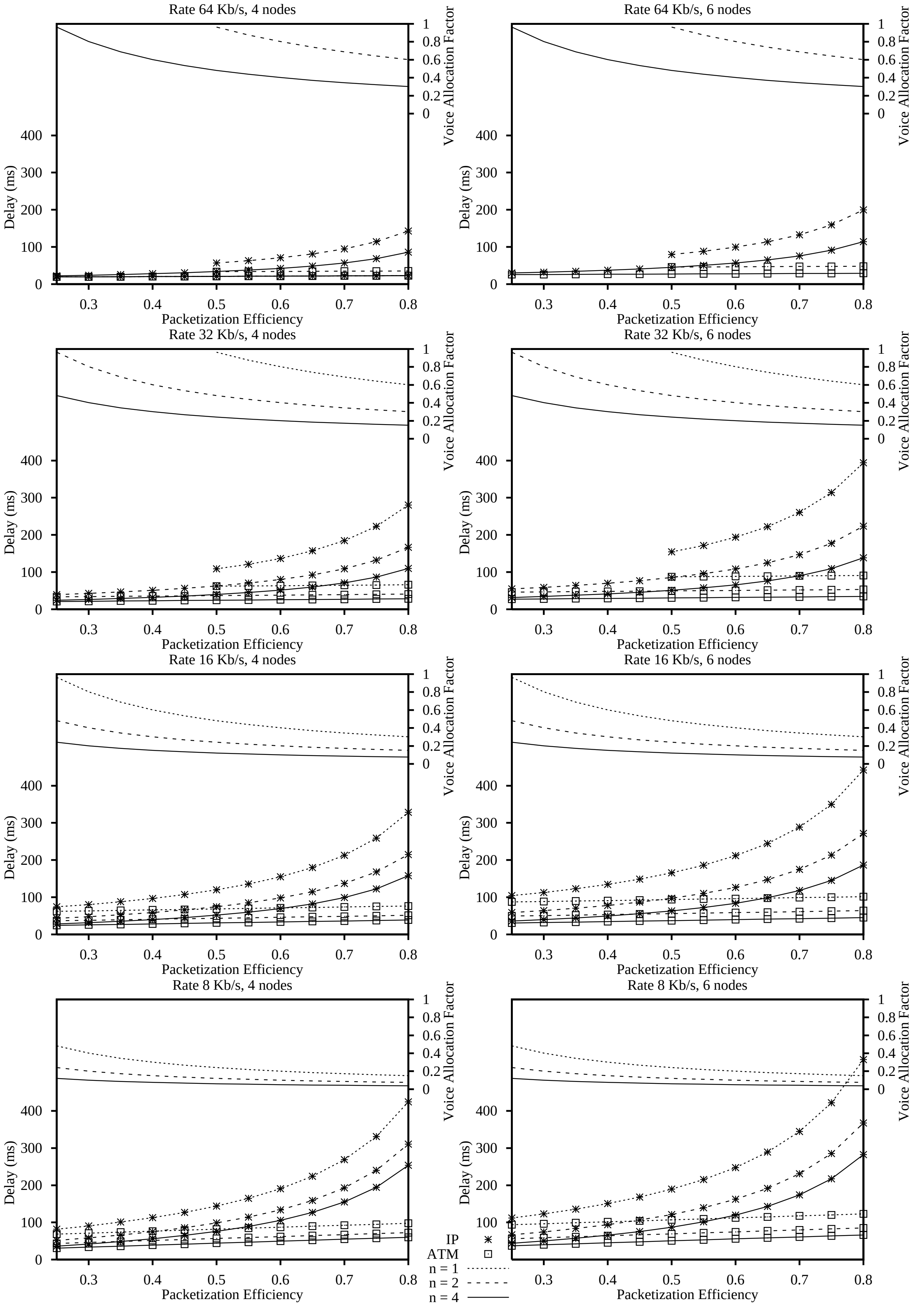


Figure 2: End-to-end Delay of a Long Distance Call.

Finally, to devise the results shown in this section, we consider encoding techniques which provide bit streams at 8, 16, 32, and 64 kb/s. The encoding schemes currently deployed introduce delays ranging from a hundred of ms up to hundreds of ms. Since there are techniques operating at the same bit rate which introduce significantly different delays, the numerical results plotted in the figures of this section have been obtained considering $D_{proc} = 0$.

shows the delay (left y-axis) of a long distance call routed through 4 (left column) and 6 (right column) IP routers (star sign) or ATM switches (square sign). The delay is calculated through Equation for a range of packetization efficiencies (x-axis). Three cases are considered for the capacity of links: $n = 1$ (dotted line), $n = 2$ (dashed line), and $n = 4$ (continuous line). Four encoding rates for the voice signal are considered (one in each row). On the right y-axis the voice allocation factor is plotted for each configuration. As shown by Equation , given a value for the packetization efficiency, the voice allocation factor does not depend on the packet characteristics (i.e., it has the same value for IP and ATM).

In the first two rows (voice encoding at 64 kb/s and 32 kb/s) the delay is not plotted some of the configurations for all or a range of the values of packetization efficiency. For example, in the upper left picture, for a link capacity double than in the circuit switched network ($n = 2$, dashed line), the delay corresponding to a packetization efficiency smaller than 50 % is not plotted. This happens because when $h_{pkt} < 0.5$, more than half of the link capacity is wasted to carry packet headers and thus there is not enough capacity to carry the same number of phone calls as in the original telephone network.

In IP networks the delay grows more than in ATM networks as packetization efficiency increases. A cell is assumed to be sent as soon as its payload has been filled. This happens when AAL1] encapsulation is used. Packetization efficiency is varied in ATM networks by filling only part of the cell payload[8]. This slightly varies the packetization delay, but does not change the queuing delay which, according to Equation , depends on cell dimension. Audio samples are assumed to be carried over IP networks using the protocol architecture shown in . The *Real-time Transport Protocol* (RTP)] provides timing relationship between sender and receiver. The protocol headers result in a fixed per packet overhead. Thus, in IP networks packetization efficiency is increased by enlarging packets which translates in an increase of both packetization delay and queuing delay, as reflected by the steep slope of the curves.

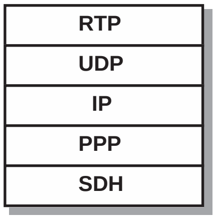


Figure 3: Protocol stack used to carry voice over IP networks

When exploiting ATM technology, the end-to-end delay can be kept below the interaction bound (it is shorter than 100 ms in all the configurations). When IP technology is exploited, the interaction bound is respected only when operating at very low packetization efficiency (less than 50 %). As a consequence, more than half of the links capacity is wasted for transmitting packet headers. The voice allocation factor curves show that when packetization efficiency is high, a large fraction of link capacity is still available to best effort traffic $(1 - a)$. Decreasing packetization efficiency reduces the amount of best effort traffic that can be carried by the network, even though the number of phone calls supported is the same. Header compression] can be exploited to increase packetization efficiency while keeping packets small. This introduces higher storage and processing requirements on IP routers but decreases the average amount of overhead traffic generated on links. The effect of header compression on the end-to-end delay, link utilization ((), and buffer occupancy are out of the scope of this work (see] for more details).

Lastly, speech compression leads to unacceptable delay when exploiting IP technology and packetization efficiency higher than 50%. Using ATM, speech compression slightly increases the end-to-end delay[9], but leaves more capacity in the network for the transmission of best effort traffic (the voice allocation factor decreases).

In any configuration, increasing link capacity substantially improves the performance because queuing delay (inversely proportional to link capacity) is a major component of the end-to-end delay. The extra capacity is not wasted as it can be used by best effort traffic. In addition, over-dimensioning links appears to be sensible since bandwidth is an ever cheaper resource in both links and node ports (when using packet switching technology). If the network carries guaranteed traffic other than telephony, more complex queuing policies (e.g., weighted fair queuing]) should be used in nodes and link capacity should be dimensioned differently (see] for details). General results remain almost the same.

5 Integrated IP and ATM Architectures for Packet Switched Telephone Networks

In this section we describe some architectural choices for a PTN based on the integration of IP and ATM.

5.1 Motivations

In the previous section ATM has been shown to be the best technology for telephony because it offers end-to-end delays lower than IP. Nevertheless, it is strongly desirable for a PTN to be equipped with IP forwarding capabilities for the following reasons:

- IP is the most widely deployed network protocol in both the Internet and many intranets, i.e., lots of applications based on the services provided by the TCP/IP protocol suite are daily deployed on many platforms. Many of them, for example electronic mail, news and the World Wide Web, are nearly ubiquitous.
- Being an internetworking protocol, IP enables communications among networks based on different technologies. On the contrary, native ATM applica-

tions require ATM to be exploited along the whole path between communicating entities.

- IP networks have a consolidated and widely supported management framework based on the *Simple Network Management Protocol* (SNMP) and the related definition of *Management Information Bases* (MIBs) for a variety of network devices.

Moreover, no particular advantage is given by the exploitation of ATM for high bandwidth real-time applications (like, for example, videoconferencing and video on demand), with respect to IP with QoS support. In fact, due to the high burstiness of the generated traffic, the gain in statistically multiplexing long burst of small cells over statistically multiplexing large packets is not significant, and queues in ATM switches grow almost as large as in IP routers.

Recently, various proposals for the operation of IP over ATM networks have been made. They aim at providing IP forwarding services while taking advantage of the high speed and, in some cases, the QoS guarantees offered by ATM. Some of these proposals exploit ATM as a data-link layer technology on which IP packets are transferred among routers (e.g., the Classical IP Model). Others, like *Tag Switching* by Cisco Systems Inc.], *IP Switching* by Ipsilon Networks Inc.], and *Cell Switch Router* by Toshiba Corp.] are good candidate foundations for PTNs since they integrate IP and ATM mechanisms.

These integrated IP-ATM approaches aim at maximizing the exploitation of ATM switching and minimizing the use of IP forwarding. This translates in transporting IP traffic over ATM *Virtual Connections* (VCs) whose endpoints are as close as possible (possibly coinciding) to IP endpoints. The use of ATM signaling and routing protocols is extremely limited, if not avoided at all, i.e., only ATM switching functionality and possibly QoS guarantees are exploited.

The basic ideas common to the various integrated IP-ATM approaches may be summarized as follows:

- Integrated IP-ATM routers are based on ATM switching fabrics so that they can forward ATM cells coming from either IP endpoints (hosts or routers) equipped with ATM interfaces or other integrated IP-ATM routers.
- IP routers somehow identify traffic flows between IP endpoints and autonomously decide whether these flows are best served by classical hop-by-hop IP forwarding (e.g., e-mail, DNS queries, etc.) or ATM cell forwarding, i.e., flows must be carried on dedicated VCs (e.g., multimedia traffic, file transfers, etc.).
- When needed, integrated IP-ATM routers along the communication path create a dedicated VC “on the fly”. IP packets are segmented into ATM cells at the source VC endpoint, forwarded cell-by-cell, and reassembled only at the destination endpoint.

In order to setup the dedicated VCs on which IP flows are transported, integrated IP-ATM routers communicate among themselves (and possibly with IP endpoints equipped with ATM interfaces) through simple and efficient service protocols. They allow integrated IP-ATM routers to exchange information about IP flows and the associated VCs. Integrated IP-ATM routers setup and tear down ATM VCs controlling directly their own switching fabric. The various integrated IP-ATM approaches essentially differ in the way these service protocols operate.

If a PTN is based on one of the integrated IP-ATM approaches discussed above, telephony has a quality (in terms of end-to-end delay) close to the one obtained over an IP network. Over an IP network, voice samples are put into IP packets and forwarded by routers, as shown in a. Over an integrated IP-ATM network, IP packets containing digital speech samples are segmented into ATM cells which are forwarded by the ATM switching fabrics to their destination, as depicted in b. The IP like performances stem from the fact that (1) cells are sent into the network in bursts of the dimension of an IP packet, thus affecting buffers like IP packets, and (2) at the receiver IP packets must be reassembled before the contained voice samples can be played back. Thus, speech samples cannot be played as soon as ATM cells arrive, but only when the last cell of each IP packet has arrived. As a result, the delay of the whole packet is that experienced by the last ATM cell (See] for further details).

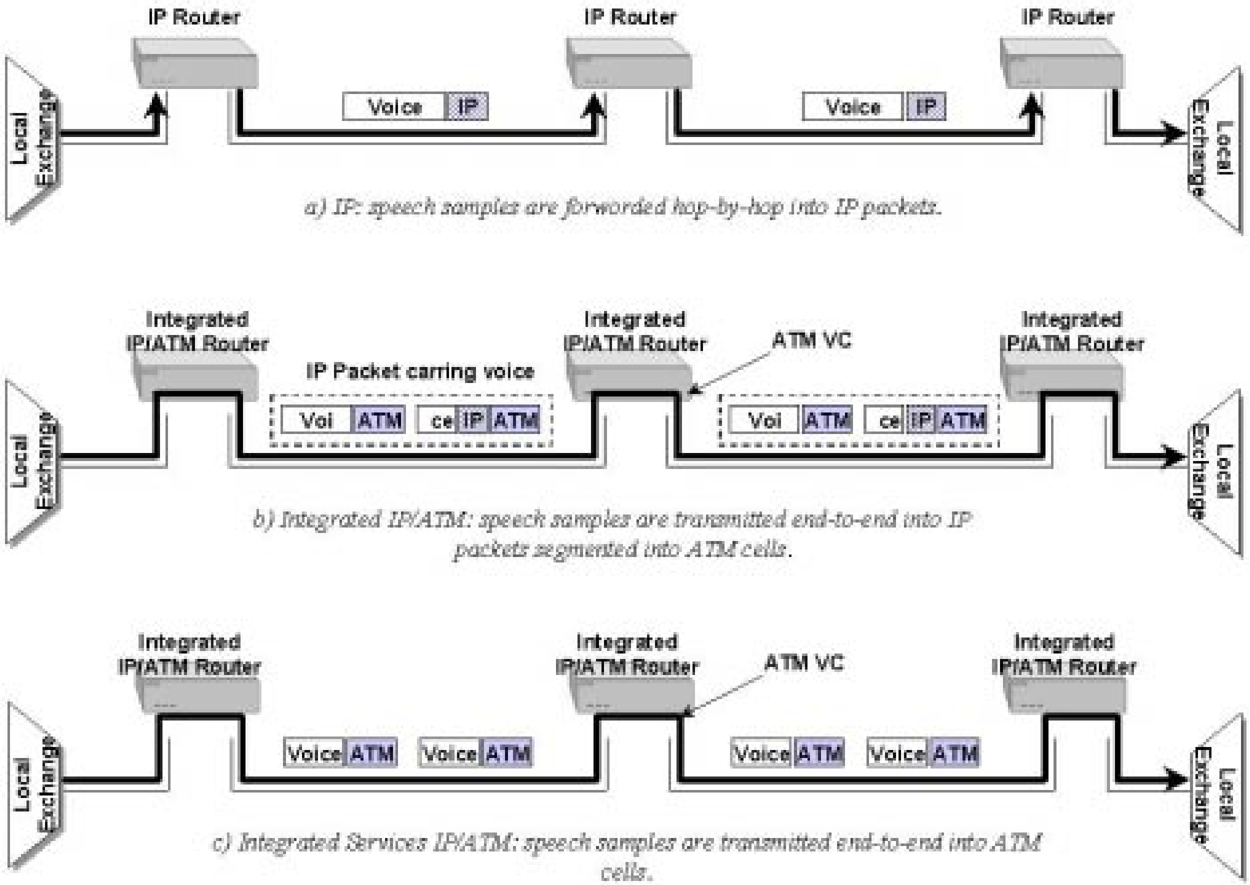
5.2 Integrated Services IP-ATM Models

In order to get better performance from a PTN, we propose to devise *integrated services IP-ATM* models derived from the integrated IP-ATM approaches described above. The basic idea behind integrated services IP/ATM is that once an end-to-end VC is in place, digital speech samples are encapsulated directly into ATM cells (c), i.e., they are transmitted like over AAL1. At the receiver, voice samples are played back as soon as the ATM cell carrying them arrives. Current integrated IP-ATM approaches use the *Unspecified Bit Rate* (UBR) class of service, i.e., they support only best effort traffic. On the contrary, telephony requires guaranteed QoS. Since most speech encoding schemes produce a constant rate stream, integrated services IP-ATM routers must provide the *Constant Bit Rate* (CBR) class of service.

In order for a phone call to immediately get its dedicated VC, integrated services IP-ATM routers should be able to set up the VC without having to previously identify a traffic flow. Some alternative solutions can be envisioned:

1. A “signaling” IP packet is used to announce the beginning of a phone call; all routers along the communication path should react to it by setting up a VC dedicated to the phone call. When a phone call has terminated, the corresponding VC must be torn down. This can be accomplished by means of a time-out mechanism (already exploited in some of the integrated IP/ATM approaches). This mechanism can be quite reactive since while a call is active, traffic is generated regularly.

2. Telephony applications can be written to exploit *User to Network Interface* (UNI) signaling]. Integrated services IP-ATM routers must implement UNI signaling and *Network Node Interface* (e.g., PNNI]) routing[10]. ATM signaling allows telephony applications to require the network for the needed QoS, while best effort and multimedia IP traffic is still handled as in integrated IP/ATM approaches. Moreover, native ATM applications (independent of their QoS requirements) are fully supported by the PTN. The main drawback of this solution is that integrated



a) IP: speech samples are forworded hop-by-hop into IP packets.

b) Integrated IP/ATM: speech samples are transmitted end-to-end into IP packets segmented into ATM cells.

c) Integrated Services IP/ATM: speech samples are transmitted end-to-end into ATM cells.

Figure 4: Forwarding models

services IP-ATM routers must run a considerably more complex and large software with respect to the previous model.

3. The amount of software running on integrated services IP-ATM routers can be reduced by exploiting the *Integrated Private Network Node Interface* (I-PNNI) routing protocol] to carry routing information for both IP and ATM. Routers still support both UNI signaling and service protocols exploited in integrated IP/ATM approaches, but IP routing protocols are not run any more.

6 Conclusions

Voice transmission over packet switched networks is here studied from the point of view of the quality perceived by the user in terms of delay. The results show that the Internet cannot provide good quality, except in particular conditions. Nevertheless, if a packet switched network is expressly designed for telephony by (1) limiting the number of hops in the path between any pair of users and (2) dimensioning link capacity properly, the end-to-end delay is low enough to allow for interaction. Thus, packet switching could be exploited to build a commercial like telephone network even though the raw capacity needed is larger than with circuit switching. This is justified by the possibility of carrying best effort traffic on the same network and by the lower installation and management costs.

The numerical results also show that ATM outperforms IP as was expected since the former was expressly designed to support low bit rate real-time traffic. Nevertheless, because of the large number of existing applications based on IP, it must be taken into account as a candidate technology for carrying both best effort and high bandwidth real-time traffic in packet switched telephone networks.

Thus, we propose that both IP and ATM be employed in the implementation of packet switched networks for telephony based on *integrated services IP-ATM* models. These models are derived from the integrated IP-ATM approaches currently being exploited for an effective operation of IP over ATM networks. Integrated services IP-ATM models provide ATM based services for telephony and IP (over ATM) based services for any other kind of traffic (namely, best effort and high bandwidth real-time) in order to get the best from the two technologies.

Another work shall be done in order to evaluate more carefully the performance of the proposed integrated services models.

Acknowledgments

We thank Prof. Silvano Gai for his advice and Agostino Vailati for the useful information provided. We thank Luca Fantolino for his important contribution.

References

[1] ITU-T, Recommendation I.327, “B-ISDN Functional Architecture”, March 1993.

[2] R. Braden, D. Clark, and S. Shenker, “Integrated Service in the Internet Architecture: an Overview”, RFC 1633, Internet Engineering Task Force, July 1994.

[3] J. Woodard, “Speech Coding”, http://www-mobile.ecs.soton.ac.uk/speech_codecs/

[4] T. Robinson, “Speech Analysis”, http://squid.eng.cam.ac.uk:80/~ajr/SA95/SA95.html

[5] M. Baldi and D. Bergamasco and S. Gai, “Telephony over Packet Switched Networks”, Technical Report, Politecnico di Torino - Dipartimento di Automatica e Informatica, December 1996 (http://www.netgroup.polito.it/publications/tr-pstn.ps).

[6] ITU-T, Recommendation I.363, “B-ISDN ATM Adaptation Layer (AAL) Specification”, March 1993.

[7] R. Gareiss, “Voice over the Internet”, Data Communications, September 1996, pages 93-100.

[8] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson, “RTP: A Transport Protocol for Real-Time Applications”, RFC 1889, January 1996.

[9] S. Casner, V. Jacobson, “Compressing IP/UDP/RTP Headers for Low-Speed Serial Links”, Internet Draft, November 1996.

[10] A. Demers and S. Keshav and S. Shenker, “Analysis and Simulation of a Fair Queuing Algorithm”, ACM Computer Communication Review (SIGCOMM’89),1989, pages 3-12.

[11] Y. Rekhter, B. Davie, D. Katz, E. Rosen, George Swallow, “Tag Switching Architecture Overview”, Internet Draft, January 1997.

[12] P. Newman, G. Minshall, T. Lyon, L. Huston, “IP Switching and Gigabit Routers”, IEEE Communications Magazine, January 1997.

[13] Y. Katsube, K. Nagami, H. Esaki, “Router Architecture Extensions for ATM: overview”, Internet Draft, October 1996.

[14] The ATM Forum, “ATM User-Network Interface Specification - Version 3.1”, September 1994.

[15] D. Dykeman and M. Goguen, “Private Network-Network Interface Specification Version 1.0”, March 1996.

[16] R. Callon, “Integrated PNNI for Multi-Protocol Routing”, ATM Forum 94-0789, September 1994.

[1] In the context of this work packet switching is intended in a broad sense, encompassing also cell switching.

[2] The time P_s/C_i a packet spends in the buffer while being transmitted on the output link has been already taken into account as transmission delay.

[3] Usually, the compensation delay is dimensioned with respect to some percentile of the delay variation, instead of considering its maximum value. As a result, the bound is smaller, but the samples experiencing delay larger than the chosen percentile are discarded. This yields unpredictable QoS because the distribution of the delay over the samples is not known a priori. In this work, the compensation delay is dimensioned according to the maximum delay variation d_{RB}; nevertheless, the obtained results still hold (with proper adaptations) also when the bound is probabilistic.

[4] Actually, in Equation the term Q_{max} accounts twice being $d_{RB} = Q_{max}$ because the processing delay is constant, the queuing delay is the only variable component of the network delay, and Q_{max} is the maximum variation of the queuing delay (see Equation).

[5] Throughout this work we assume P to be constant during a voice connection.

[6] For this calculation the *Synchronous Digital Hierarchy* (SDH) is assumed to be used to connect packet switching nodes. Due to the dimension of the problem, 155 Mb/s STM-1 carriers are the candidate links.

[7] Actually, a further approximation is introduced: the granularity of real SDH carriers is not taken into account when dimensioning the links among nodes. When actually building the network, an integer number of SDH carriers is installed between each pair of nodes in order to provide a capacity greater than the value computed using Equation . This yields a smaller value for a, i.e., actual delays shorter than those presented in this section.

[8] This feature is not currently supported by AAL1.

[9] The increase in the end-to-end delay is due to the packetization delay: the lower the bit rate at the exit of the voice encoder, the longer the time needed to fill the packet payload.

[10] These are usually implemented in into ATM switches.