Experiments with packet switching of voice traffic

P.N. Clarke, B.Sc., Ph.D., and Prof. L.F. Turner, B.Sc., Ph.D.

Indexing terms: *Telephone exchanges and networks, Voice traffic*

Abstract: There has been much interest recently in integrated services digital networks carrying both voice and data traffic. Packet switching is being used to carry data in an attempt to make better use of trunk capacity than with circuit switching. In a telephone conversation, for most of the time only one person is talking, and it has been suggested that packet switching can lead to economies in carrying voice traffic also. In view of the variable delays associated with store and forward switching, buffering is usually required at the receiver to enable received speech to be reconstituted at the proper rate. Simulation experiments of packet switching of voice traffic with fixed packet routing have been carried out. The results of these simulation experiments, which are described in this paper, show that, for a single link between two exchanges, 22 conversations can be carried by packet switching with reasonable delay. For the same inter-exchange-link capacity, only 15 conversations can be carried by circuit switching. For a larger network with more exchanges and links per path, a similar advantage is also found with packet switching. The results show that the standard deviation of interpacket delay for successive packets of the same talkspurt is an order of magnitude less than the standard deviation of packet transit time for all packets. This suggests correlation of flows of packets within the same talkspurt. The wider variation of transit delay applies to each talkspurt as a whole and all packets within the talkspurt have correlated transit times, and hence interarrival times. The fact that the standard deviation of interpacket delay is small as compared with the standard deviation of packet transit time suggests that the receiver buffering requirement is less than that indicated by the standard deviation of the packet transit time.

1 Introduction

As a result of the recent increases in data traffic, various suggestions have been put forward relating to the use of separate data networks. The existing analogue circuit-switched telephone network has transmission and noise characteristics which vary significantly through the network, and call set-up times of the order of seconds are involved. Although this situation is acceptable in so far as voice traffic is concerned, it is unacceptable for many data applications. On account of the burst-like nature of the data, in many applications store and forward switching methods, such as packet switching, have been proposed and implemented [1–6]. Packet switching makes better use of expensive high-capacity interexchange trunks by transmitting small blocks of data, or packets, only when there are data to be sent. If there are, for short periods, more packets for transmission than can be dealt with, some are stored for forwarding in later less busy periods. Packet switching makes efficient use of trunk capacity at the expense of variable delay.

Rather than have two separate networks, one for data and one for voice traffic, a single network for both types of traffic may be more economical. As digital transmission and switching methods are being used increasingly for speech, and as data are best handled in digital form, integrated services digital networks (ISDN) are being proposed. These might be implemented using the new electronic digital circuit-switching exchanges, such as in System X [7]. Alternatively, depending on the relative costs of switching and transmission, packet switching might be used to make use of trunk capacity during silent periods. Systems such as TASI [8] have been used in the past on both transocean cable and satellite circuits in order to make use of silent periods.

Packet switching with its variable delays might be considered unsuitable for real-time application such as conversational speech. If, however, a buffer is used at the receiver then the variations in packet arrival times can be smoothed out and the received speech reconstituted at the correct rate. This does, of course, add to the total speech delay. The total delay resulting from packet creation, network transit time, and receiver buffering and decoding must not be too long (cf. 270 ms 1-way delay through a satellite link). It has been observed [9] that delay in excess of 900 ms can give rise to considerable difficulties. Replies and nonverbal responses, together with their relative timings, provide the speaker with clues as to the listener's understanding and thus aid the conversation process.

Minoli [10, 11] considered theoretically talker behaviour and end-to-end, that is, packet transit delay for a link packet-switched voice system. He also considered delay dependencies on packet size and the effects of the number of queue buffers at the link output. Coviello [12] also considered end-to-end delay for a variety of network parameters and a variety of alternative network protocols to facilitate packet switching of voice traffic. Gruber [13] reviews a variety of switching techniques for voice traffic and is again concerned with end-to-end delays. A variety of speech coding techniques are reviewed and the results of some ARPA network voice experiments are described by Gold [14].

These works [10–14] have been concerned very largely with the end-to-end, or packet transit, delay and its variation, and the workers involved have considered this variation to be the principal factor determining the buffering requirement at the receiver; with the buffer being necessary to even out irregular packet arrivals. Although the packet transit time, if large, and its variation may have a significant effect on conversational behaviour (see Reference 9), it is, however, the variation of interpacket delay, rather than packet transit time, which determines the receiver buffering requirement. In the experiments carried out and described in this paper the interpacket delay (that is, the delay between arrivals of successive packets within the same talkspurt) and its standard deviation were measured in order to investigate the correlation of packet flows. The results of simulation experiments carried out with a fixed packet routing system show that the standard deviation of interpacket delay for successive packets of the same talk-

Paper 2586G, first received 3rd June 1982 and in revised form 17th February 1983

Professor Turner is, and Dr. Clarke was formerly, with the Department of Electrical Engineering, Imperial College of Science & Technology, South Kensington, London SW7 2BT, England. Dr. Clarke is now with British Telecommunications, Gower

ation of the packet transit time. This thus suggests that the receiver buffering requirements are significantly less than suggested by packet transit-time statistics.

This paper describes an investigation into the delays involved in the use of packet switching for voice traffic. In the course of the investigation, a computer simulation model was devised and this is described in Section 2 of the paper. The experiments carried out and the results obtained are described in Section 3, and some conclusions to be drawn from the work are presented in Section 4.

2 Packet-switched voice network simulation model

The simulation model developed will be described in two parts:

(*a*) the talker activity model (Section 2.1)
(*b*) the packet-switched network model (Section 2.2).

Part (*a*) deals with the nature of the interaction between the talkers, and part (*b*) with the packet-switched network itself, which transports the speech in packet form.

2.1 Talker activity model

In most conversational speech between two people, one is silent at any given time (listening while the other is talking). There are, however, occasions when both are silent and or when both are talking simultaneously (e.g. when one person interrupts the other). Talker activity can be thought of in terms of active periods (talking) or silence periods. These periods can be the main active periods of significant utterances, such as sentences, and the silence of a listener while another person is talking. Alternatively, the fine structure of the significant utterances can be taken into account. This fine structure refers to the actual time during which a sound is being made by a talker and the pauses between sentences, words and syllables.

The principal object of a packet-switched network is to make efficient use of network transmission capacity. It is thus clear that packets should only be carried by the network for any conversation, while either of the parties of that conversation is actually speaking. In this way, the silence periods of conversation can be filled in on the high-capacity trunks which are shared by many talkers. A larger number of talkers can thus use a given trunk capacity than with circuit switching. Speech detection equipment should produce an output to be put into packets according to the coarse or the fine structure of talker activity, depending on the speech-detector sensitivity and switching speed.

Studies have been carried out of the talker activity during telephone calls. Norwine and Murphy [15] consider principally the coarse structure of the interactions between talkers. Brady describes an experimental arrangement for measuring fine structure of talker activity [16], the analysis of data gathered using this apparatus [17] and the fitting of such data to a theoretical model for generating probabilities of transition between states of talking, silence, interruption etc. [18].

The talker activity model used for the simulation experiments, and reported on in this paper, was based on the results given in Figs. 3 and 5 of the paper by Norwine and Murphy [15], and thus does not take account of pauses within talkspurts. It would have been possible, using a Markov chain model, to obtain finer details of talkspurt activity, but as this approach is considerably more difficult to implement than the probability density function approach, it was not adopted in the simulations leading to the results presented in this paper. However, an talkspurt activity may well be of value, and could form the basis of a more extensive further consideration of packet-switched systems used for the transmission of voice traffic. In Reference 15, graphs are given of talkspurt length and response time distributions, with response time being defined to be the length of time between the end of one talker's talkspurt and the beginning of the next talker's talkspurt. The distribution of response time includes negative values, that is, interruptions. A positive value of response time corresponds to the more normal period of mutual silence between talkspurts before the next talker begins. Using the talkspurt duration statistics given in Reference 15, the talkspurt duration was approximated in the work reported on in this paper, using a lognormal distribution [19] having the same mean and modal values.

The lognormal distribution has a PDF, $f(x)$, given by

$$f(x) = \frac{1}{x\sqrt{2\pi\sigma^2}} \exp\left\{\frac{-(\log_e x - \mu)^2}{2\sigma^2}\right\}$$

where μ and σ^2 are the parameters of the distribution. With the mean and mode of the distribution, as given in Reference 15, $\mu = 0.485$ and $\sigma^2 = 1.871$. As regards response time, this was approximated using a normal distribution with mean 0.32 and standard deviation 0.584 (all times in seconds).

With the model used, the talker activity, which is defined to be:

$$\text{talker activity} = \frac{\text{mean talkspurt length}}{2(\text{mean talkspurt length} + \text{response time})}$$

can be seen to be:

$$\text{talker activity} = \frac{4.14}{2(4.14 + 0.41)} = 0.45 \qquad \text{(or 45\%)}$$

In the model used in the simulation, a talker was allowed to talk for a talkspurt length, with the length being drawn from the lognormal distribution. The response time for the second talker was drawn from the normal distribution. After the talkspurt length, the first talker stops, and the second talker is allowed to begin at a time equal to the sum of the talkspurt length and the response time after the start of the first talker's talkspurt. The length of the talkspurt for the second talker was determined from the lognormal distribution. In this way the times for the second talker to stop and for the first talker to begin again were determined. If as a result of a combination of interruptions and long talkspurts a talker was scheduled to start a new talkspurt during the course of an existing talkspurt, it was arranged for the current talkspurt to be completed before the start of the next, which was then allowed to begin immediately afterwards. These points are illustrated by the simple example shown in Fig. 1.

In the Figure; at time *A*, talker 1 (T1) begins to speak until *B*. T2 is idle at time *A* and is scheduled (by T1) to start speaking at *C*. At time *B*, T1 stops and becomes idle and T2 is idle but waiting to start at time *C*. At *C*, T2 begins to speak until *E* and schedules T1, who is idle, to start at time *D*. This represents an interruption by T1 who will start talking before T2 has finished. At time *D*, T1 begins to speak until *H*. T2 is scheduled by T1 to start his next talkspurt at time *F* which is thus an interruption of T1. T2 stops talking at *E* and awaits a new start at *F*. At time *F*, T2 interrupts T1 and schedules T1's talkspurt to start at *I*. T2 stops talking at *G* and T1 carries on until *H*. T1 stops at time *H* and remains idle until the next start at *I*. At time *I*, T1 begins to speak until time *M* and schedules T2 to start at time *J*. T2 starts speaking at *J*, interrupting

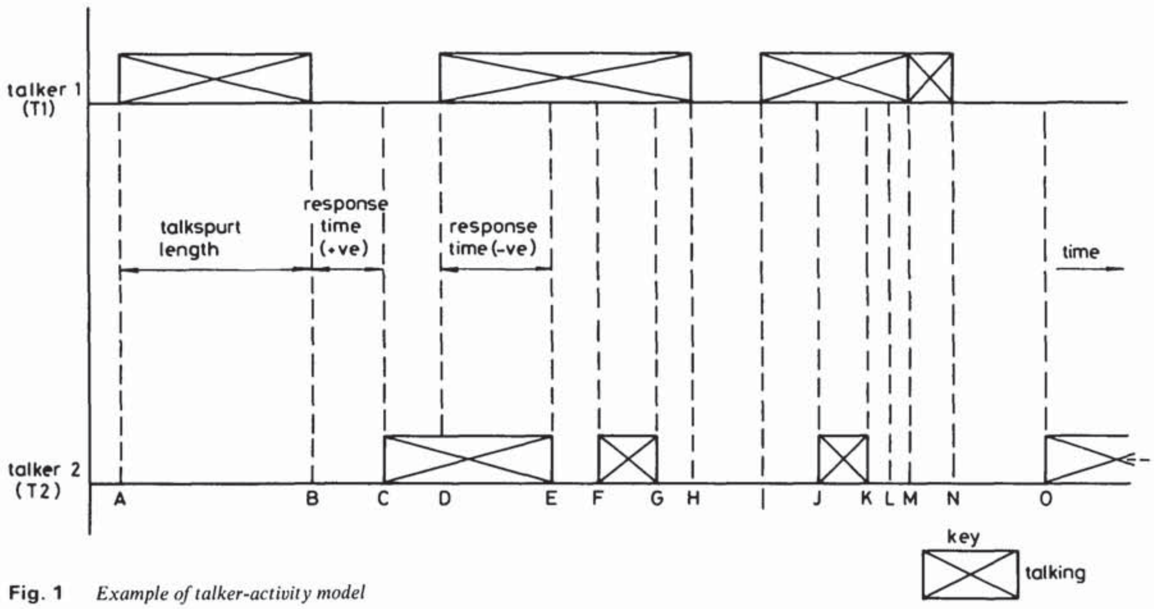


Fig. 1 *Example of talker-activity model*

At L, T1 is still talking, so he continues the current talkspurt (until M) and restarts immediately until N. Also at time M, T2 is scheduled to start at time O, and so on.

In the simulation, the following procudure was adopted. During talkspurts, the speech from talker's equipment was taken as having been digitised with all talker pairs in the network having the same speech bit rates. When enough 8-bit (byte) speech digits to fill a packet had been received from a talker, a packet was created at the exchange. An appropriate header was added to the packet which then went for transmission through the network. The next packet of the talkspurt was then filled up, and so on. At the end of the talkspurt, the packet which was being filled up was completed by filling with 'blank' information at the speech bit rate (see Fig. 2). All speech packets in the network were thus of the same length. All packets as well as being of the same length were created at regular intervals during the talkspurt.

Clearly, this simple model of the coarse structure of talker activity, and regular packet generation, makes no allowance for the possibility, depending on the nature of the interruption, of a talker stopping when interrupted. No allowance was made in the simulation model for the effects on talker behaviour of delay in packet creation, of cross-network delays, nor of buffering and speech reconstruction delays. All of these delays will in general be variable, except the regular packet creation delay. Delays in telephone channels do affect talker behaviour, as has been reported by Brady [20] in the case of fixed delays. The simple model was chosen to provide approximate conversational talker activity.

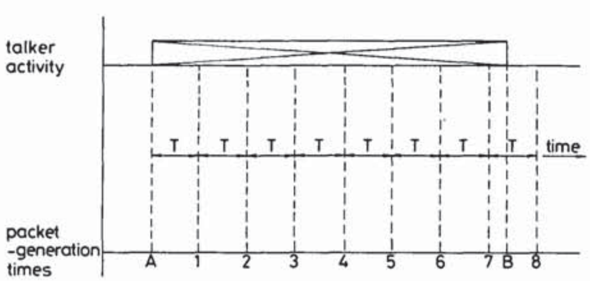


Fig. 2 *Talker activity and packet creation*

$T = \frac{\text{maximum data content of packet (bits)}}{}$

The rationale behind the simplfied approach was that if this model which does not allow for delays in speech, and operates by generating full packets at regular intervals, can handle more calls than a circuit switched system of the same trunk capacity, then a more complicated model, allowing for delays and pauses within talkspurts, may allow even more calls to take place.

2.2 *Packet-switched network model*

The network of the simulation model was made up of packet-switching exchanges (PSEs) connected by full-duplex trunks. The talkers were connected to the exchanges by lines which can be assumed to be either analogue or digital (operating at the speech bit rate). In all the examples, each talker was associated with another talker at another PSE in the network. These talker pairs were assumed to be engaged in conversation before the start of each experiment.

On generation of a speech packet at a talker's interface, the required outgoing trunk was determined by consulting the route table. Fixed routing was used in all of the experiments. Packets entering the network from a talker could only be put into the queue for this trunk if there were more than two free queue buffers. This gives some priority to transit traffic, i.e. to packets which have been accepted into the network, for example, at node 4 in Fig. 4*b*, or at node 2 in Fig. 4*c* for packets between 1 and 3, and between 3 and 1. If an originating packet could not be accepted, it was held in a buffer associated with the talker's interface to the network. That talker's identity was put into a queue associated with the trunk output queue. Whenever a packet was sent along the trunk and a queue buffer became free, the list of talkers with waiting packets was inspected. If there were sufficient buffers to allow in an originating packet, the first one waiting joined the trunk output queue. If that talker had further waiting packets, he rejoined the list of talkers with waiting packets.

Packets were transmitted over the trunks at the trunk bit rate. Copies of all transmitted packets were kept, pending acknowledgments received from the other end of the trunk. Associated with each copy of a packet, kept in the retransmission queue, was a time by which that packet must be acknowledged. This time was based on the worst possible case of acknowledgment delay. Acknowledged packet were to exceed its time in the retransmission queue, then it would be retransmitted, followed by its successors (unless these had meanwhile been deleted) before any new packets were transmitted. However, as transmission errors were not simulated, the only condition under which the retransmission procedure could have been evoked was that in which a packet was discarded at a transit node because of there being no free buffers in the output queue.

The acknowledgment process was carried out using the send-and-receive sequence numbers carried by all packets as used in the ISO's HDLC and in the CCITT's X.25 recommendation [21, 22]. Any packet carrying a send sequence number greater than that expected was discarded and a REJ (Reject) packet sent in the reverse direction. This REJ packet indicated the last correctly received packet and instructed retransmission to start at the appropriate point in the packet sequence. Only one REJ was allowed in a given direction until the next expected packet was received. If a REJ was corrupted by noise and thus discarded, the correct packet sequence was maintained by retransmission invoked by the timeout mechanism. In the case of no outgoing packets when one was correctly received, a RR (receiver ready) packet indicating correct reception was sent. This reduced the use of the timeout mechanism under conditions of light trunk loading.

Packets made their way through the network to their destination. Here they were assumed to be passed to the receiver interface for conversion to speech (after any buffering, if necessary). On arrival of every packet, the packet statistics were updated. Packet statistics measured included:

(i) the number of packets received

(ii) the mean and standard deviation of packet transit time for the packets of (i). Packet transit time was measured as the difference between the arrival time at the destination PSE and the packet creation time at the source PSE

(iii) the mean and standard deviation of packet interarrival time. Packet interarrival time was defined as the difference between arrival times of successive packets of the same talkspurt.

The simulation program was written in Simula [23, 24] and was designed to be as flexible as possible. A wide variety of networks and conditions could be simulated by choosing appropriate input data for the program. The input data required for this were:

(i) the number of PSEs

(ii) the number of trunks

(iii) for each trunk: (*a*) the source and destination of PSEs, (*b*) the trunk capacity, (*c*) the bit error probability, and (*d*) the retransmission timeout period

(iv) the route table (this gives the next PSE en route to each destination)

(v) the talkers' speech bit rate (the same for all talkers)

(vi) the speech packet length (in bytes)

(vii) the number of PSE pairs with conversations between them

(viii) for each of (vii) above, the number of talker pairs

(ix) the duration of the simulation and intervals between statistics report

(x) the seed for the random-number stream used.

3 Packet-switched voice network experiments

3.1 General description

Three simulation experiments were carried out, and there were several model parameters common to the experiments (with two reserved for transit traffic). The talker speech bit rate* was 9600 bit/s. There were no local calls (i.e. calls between talkers at the same PSE). The program was run for 250 s in all experiments and the results of the first 100 s were removed in order to reduce the bias effects of no packets being present in the network at the start of the simulation. Analysis of the results has shown that a stable condition was reached in this time. Results were also collected for a single talker pair. The three experiments carried out were as follows:

(i) Two PSEs, one 144 kbit/s trunk (see Fig. 3); 128 + 8 (speech + header) byte packets; 25 ms retransmission timeout interval; varying number of talker pairs.

(ii) Two PSEs; one 144 kbit/s trunk (see Fig. 3); 15, 20 and 25 talker pairs, packet sizes of 32, 48, 64, 96, 128 (from previous experiment) 192 and 256 bytes (with 8 bytes of header in addition with correspondingly adjusted retransmission time.

(iii) Three PSEs (see Fig. 4): (*a*) a fully connected network with 144 kbit/s trunks; (*b*) a star network with 288 kbit/s trunks; and (*c*) a linear network with 288 kbits/s trunks; 128 + 8 byte packets; 12.5 ms retransmission

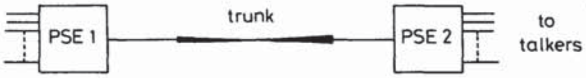


Fig. 3 *2-node packet-switched voice network*

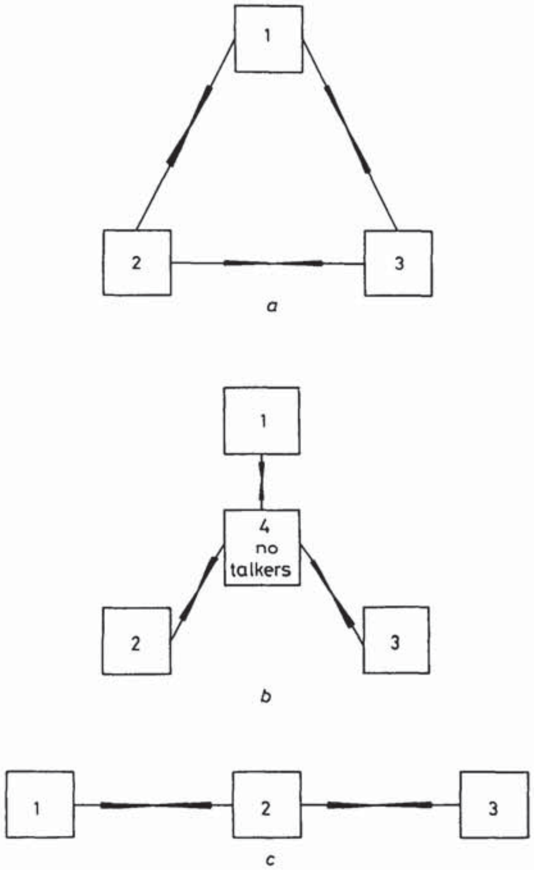


Fig. 4 *3-node packet-switched networks*
a Three nodes: FC, *b* Three nodes: star, *c* Three nodes: linear

* A 9600 bits/s speech rate was used in order to facilitate the simulation. The significance of the results so obtained is not, however, restricted by this. Appropriate time scaling of packet lengths and trunk-line rates would render them applicable

timeout period on the faster 288 kbits/s trunks of (*b*) and (*c*); number of talker pairs varied. (The abbreviation FC is used in the Figures to refer to the fully connected network configuration.)

3.2 Results of experiments

The results of the experiments will now be described. Related points in all Figures are joined by straight-line segments to identify related points in the multigraph Figures and to indicate trends, rather than to show exact behaviour, between the experimental points.

3.2.1 Two PSEs, 128 + 8 byte packets, varying number of talker pairs: The number of packets transferred in the 150 s (for each value of talker load) of the experiment for all talker pairs and for the single talker pair are shown in Fig. 5. The number of packets for all pairs rises almost linearly up to 25 talker pairs, with a smaller rise between 25 and 27. For the single talker pair, almost the same number of packets are carried at all loads (total number of talker pairs). The average packet transit time of Fig. 6 shows little increase up to 22 talker pairs but shows an increasing rate of increase above 22 pairs. The average packet transit time must be added to the packet creation time of $(128 \times 8/9600)$ s = 107 ms to obtain the total delay between speech being uttered and becoming available for reconstruction on arrival at the destination PSE. Any buffering to allow for variations in arrival times must be added as well. Up to 22 talker pairs, the transit time is less than 20 ms. The standard deviation of packet transit time, shown in Fig. 7, is low (less then 30 ms, suggesting receiver buffering of over 100 ms) up to 22 talker points, but it increases more rapidly as more talkers are added to the network. This suggests that up to 22 talker pairs with speech bit rate of 9600 bit/s, with 128 + 8 byte packets, can share a 144 kbit/s trunk with an average speech delay

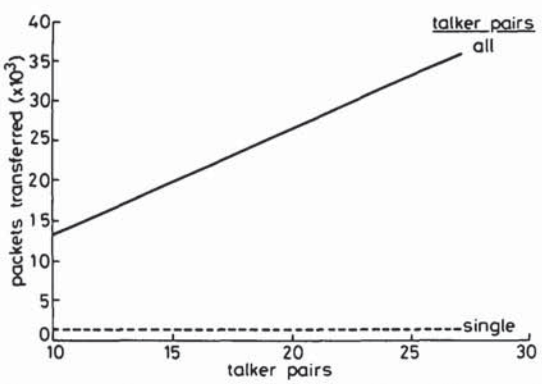
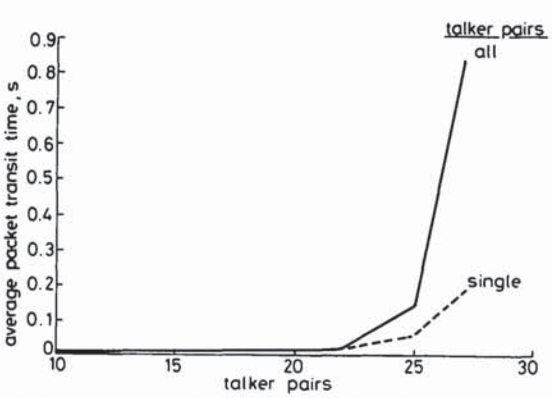


Fig. 5 *Packets transferred, varying talker load, two nodes, packet length = 128 + 8*



of 107 + 20 = 127 ms (before receiver buffering). For more than 22 talker pairs, the packet transit time and standard deviation will lead to even greater delays. It will be noticed that there is a difference between the packet transit time and standard deviation curves for all talker pairs and for single talker pair (see Figs. 6 and 7). This is because of the effects of the smaller sample size of packets from the single talker pairs (see Fig. 5). The single talker pair results will not be considered in the rest of this paper. Packet switching appears to be able to carry the conversations of 22 talker pairs (under the above conditions) before delays become unacceptable. A 144 kbit/s trunk operating under circuit switched conditions can carry (144 000/9600) = 15 conversations with no variable delay.

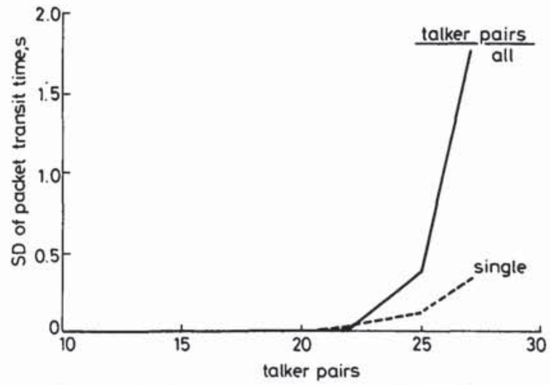


Fig. 7 *SD of packet transit time, varying talker loads*

The average interpacket delays of Fig. 8 are dominated by the 107 ms packet creation delay and are almost equal to it for up to 25 talker pairs. In fact, the variation in delay due to queueing was found to be approximately three orders of magnitude less than the transit time, and to exhibit no systematic variations.† This indicates that, even with the extra transit time (queueing for transmission over the trunk), there is little difference between the admission queueing and transmission delays for successive packets of talkspurts. The increase in interpacket delay for more than 25 talker pairs indicates that successive packets of each talkspurt take longer to reach their destination than each of their predecessors. A packet-switched network is clearly unsuitable for carrying speech traffic when operated in this region. The standard deviation of interpacket delay is less than 4 ms for less than 22 talker pairs. This is approximately an order of magnitude less than the standard deviation of packet transit time.

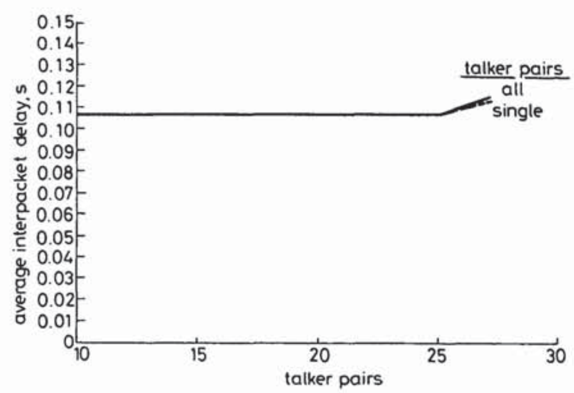


Fig. 8 *Interpacket delay, varying talker loads*

† Details of the effects of speech statistics on the perception of impairments arising

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.

WHAT WILL YOU BUILD? | **sales@docketalarm.com** | **1-866-77-FASTCASE**

fastcase®
Smarter legal research.