PACKET SPEECH INTERPOLATION IN
MOBILE TELEPHONE SYSTEMS

A thesis submitted in fulfilment of the requirement for the degree of

DOCTOR OF PHILOSOPHY

in the

Department of Electrical Engineering, The University of Adelaide

by

JOHN CHARLES ELLERSHAW, B.Sc., B.E.(Hons)

June 1979

Awarded April 1980

TABLE OF CONTENTS

ABSTRACT

Mobile telephone services are presently provided by frequency division multiplexing techniques. A voice circuit in the form of a radio frequency channel is allocated to a subscriber for his sole use during a telephone call. An alternative to this is to employ time division multiplexing and provide voice circuits on a digital basis.

In such a scheme speech is digitized and formed into packets for transmission. It is possible to interpolate packets from different conversations onto a single high capacity radio channel. This is a form of time division multiple access and is best implemented by a technique termed reservation ALOHA. When a packet is prepared at a mobile telephone a request is sent, over a small capacity channel, for transmission space in the main or speech channel.

To determine the efficiency of this process the characteristics of speech in packets are investigated through computer simulation. An optimum packet length is shown to exist and is obtained for a wide variety of different systems.

Various delays are inherent in packet speech interpolation and these are considered in turn. The most important is the delay involved in assigning space in the speech channel in times of greater than average speech activity. Limiting this delay to maintain reasonable speech quality results in certain packets not being transmitted. Such packets are said to have been glitched.

The nature and extent of glitches is studied both theoretically and by further simulation. Their subjective effects are also considered

and acceptable rates of speech loss are determined. Within this constraint and another imposed upon the total delay, an optimized packet system is designed. This is capable of servicing over 4000 subscribers in a total capacity of 2.16 Mbit/s.

A comparison is performed between the above packet system and alternative small cell frequency modulated (FM) systems which employ radio channel reuse. It is shown that with similar channel repetition the packet scheme can provide more telephone connections within a given bandwidth. There are however penalties of greater cost and complexity. A compromise system incorporating interpolation with the small cell FM structure is shown to be superior to both, at least for applications in the near future.

Alternative areas for packet speech interpolation such as pulse code modulation telephone links, satellite systems and integrated data and speech networks are also considered. It is found that these very much parallel the mobile telephone situation and that the results obtained for the latter are of direct relevance. In some ways these areas are better suited to the use of packet speech interpolation than is a mobile telephone system and they offer the best opportunity for its implementation at the present time.

STATEMENT

I declare that this thesis contains no material which has been accepted for the award of any other degree at this or any other University, and that, to the best of my knowledge, it contains no material previously published or written by any other person except where otherwise acknowledged in the text.

John Charles Ellershaw

ACKNOWLEDGEMENTS

I offer sincere thanks to my two supervisors Dr. B.R. Davis and Professor R.E. Bogner, both for their technical guidance and for their enthusiastic support. They have created a most suitable atmosphere for productive research and have often provided inspiration through innovative approaches to problems.

Also I thank Telecom Australia for permitting study leave and for granting financial support in the form of a Study Award. The information provided by Jack Steel of Telecom Research Laboratories is gratefully acknowledged. Financial assistance was also given by the George Fraser scholarship for which I am grateful both to the donor and to the University of Adelaide.

The efforts of all those involved in preparing this thesis are very much appreciated. I thank both of the typists, Doreen and Janet Ellershaw, who have done an excellent job, and also the proof readers, Harold Leek, John A. Ellershaw and Bruce Davis.

LIST OF SYMBOLS

A	Excess capacity factor on an ALOHA channel.
B	Number of bits in an information packet (bits).
C	Nominal number of voice circuits in a speech channel.
D	Bit rate of digitized speech (bit/s).
E	Expected number of entries in an infinite M/D/1 queue.
F	Freezeout fraction in a TASI system.
G	Total traffic on an ALOHA channel (packets/slot).
GP	Glitch probability.
GR	Glitch rate, or fraction of packets glitched.
I	Interpolation gain of speech in packets.
I_f	Full channel interpolation gain.
I_s	System interpolation gain.
K	Maximum random delay in packet retransmission on an ALOHA channel (slots).
L	Maximum length of slot assignment queue (speech packet slots).
L_f	Number of request slots within a subpacket recording time.
L_t	Number of request slots within the total of a subpacket recording time and the slot assignment limit.
L_r	Number of request slots required for resequencing.
M	Number of subpackets per packet length.
N	Number of simultaneous users in a system.
N_c	Number of calls commencing per second.
P	Speech activity factor.
P_{col}	Probability of a collision.
Π_i	Steady state probability of being in state "i".
Q	Matrix of state transition probabilities.
R	Acknowledgement packet time-out wait in an ALOHA system (slots).

R_a	Ratio of acknowledgement channel to speech channel capacities.
R_r	Ratio of request channel to speech channel capacities.
S	Throughput of an ALOHA channel (packets/slot).
S_r	Request channel information rate (bit/s).
T	Packet length; time of speech recorded in a packet (seconds).
U	Usage; average number of requests arriving per transmission slot.
V	Time limit for slot assignment (seconds).
W	Capacity of the speech channel (bit/s).
W_a	Acknowledgement channel capacity (bit/s).
W_r	Request channel capacity (bit/s).
W_s	Assignment channel capacity (bit/s).
W_t	Total one way radio channel capacity (bit/s).
Z	Time of a speech channel transmission slot (seconds).
Z_r	Time of a request channel transmission slot (seconds).

1. INTRODUCTION

1.1 Background

Telephone service to mobile subscribers has grown rapidly in recent years and is expected to continue to expand in the foreseeable future [1]. To meet this demand the telephone administrations in many countries are planning and implementing high capacity mobile telephone systems [2,3].

At present mobile telephones are usually found in automobiles. However, as their size and weight are reduced, hand held, portable units will become possible [4]. Indeed there appears to be no reason why such devices cannot eventually be made as small as radio pagers. Given the present trend toward rapid accessible personal communication, the demand for such telephones could well be enormous. What sort of radio system can provide a telephone service to such a large number of people within a relatively compact city area? This thesis investigates one possibility based upon digital techniques.

The main problem with any proposed large scale mobile telephone system is to provide sufficient services within the available radio frequency (RF) spectrum. It is impossible to provide every subscriber with a unique radio channel. Instead, present systems employ a number of voice-width channels which are allocated to particular users when a telephone call is initiated. After call completion the channels are freed for reallocation [5]. Frequency division multiplexing (FDM) is used to split the RF bandwidth into voice channels [6].

An alternative to this is to use time division multiplexing (TDM), in which a voice circuit occupies the entire RF bandwidth available, but only in short bursts. Speech in this system is digitized and arranged into packets [7,8] for transmission during the allocated time.

The main advantage of a digital system is its compatibility with future digital telephone systems. As the cost of digital equipment is falling relative to that of its analogue equivalent, more and more of the normal telephone network is becoming digital. This is nowhere more evident than in the switching field where it is now cheaper in certain circumstances to employ digital rather than analogue switching, even though the speech has to be digitized prior to switching and afterwards converted back to analogue.

Another advantage of the digital technique is the possibility it offers for improved spectrum utilization as progress is made in digitizing speech at low bit rates. Further, there are inherent advantages in digital transmission because of its greater resistance to interference.

1.2 Packet Radio Techniques

In a digital mobile telephone system many users transmit packets over a common radio channel. This process is termed time division multiple access (TDMA) [9] and it relies upon each user having an average data rate which is a small fraction of the channel capacity. This is very common in computer communication networks and the majority of research in TDMA relates to this field [10,11,12].

The simplest form of TDMA is the random approach embodied in the "ALOHA" system. Here users transmit packets immediately upon their formation. Since the transmissions are entirely random two or more packets will, at some stage, overlap in time and interfere with each other. Such packets cannot be correctly received and are said to have collided. To prevent loss of information through collisions, an acknowledgement is returned to the user when a packet is correctly received. If no acknowledgement is received the packet is retransmitted.

This scheme was first studied and implemented at the University of Hawaii in 1970 in a system connecting remote terminals to a computer [13,14]. It was quickly realised that as well as providing a scheme for ground radio communications [15], the technique could be used in satellite systems [16,17,18].

The major problem with the ALOHA method is that only a small fraction of the radio channel capacity can be used without overloading. Proposals allowing greater use have been presented by Kleinrock with Lam [18,19], and with Tobagi [20,21] and by others [22,23]. The most promising technique for mobile telephone applications is some form of reservation ALOHA [17,21,24,25,26]. Here space for transmission in the radio channel is allocated by a central controller upon receipt of a request for space. In a mobile telephone scheme this entails transmitting a request over a second radio channel, the request channel, to the central controller. Reservation ALOHA is the basis of all the mobile telephone schemes proposed in this thesis.

Another important consideration for the radio connection is the environment through which the signal must pass. In a city area multiple echoes of the original signal are detected at the receiver. This multipath propagation causes severe fading with the signal amplitude varying randomly with position. The signal is also spread in time thereby limiting the achievable baud rate. This is of particular importance in wideband digital transmission. The characteristics of multipath propagation have been examined by Cox [27-29] and several other authors [30-34] and will be discussed in detail in chapter 2.

Man made noise is also a problem in transmission through the urban environment. Noise levels are much higher than in free space and impulses from automobile ignitions can swamp even high level signals [35,36]. Needless to say this can cause significant numbers of errors in digital systems. Methods for overcoming these limitations are discussed in chapter 2, where a practical radio channel arrangement for a digital mobile telephone system will be determined.

1.3 Interpolation

In any telephone conversation each person speaks for only 25% to 40% of the time. The resulting spaces may be used to carry other conversations and thereby effectively increase the number of voice circuits. This process, termed interpolation, was first used in the time assignment speech interpolation (TASI) system on the transatlantic underwater telephone cable [37]. It enabled the number of simultaneous conversations the cable supported to be doubled with only a slight degredation in speech quality [38,39].

Digital speech interpolation in which speech is interpolated in a digital format has also been proposed. It is usually considered within the structure of pulse code modulated (PCM) telephone lines [40-47], though satellite applications have also been investigated [48,49]. Few authors however, have considered the interpolation of speech in packets and none have previously investigated such an arrangement in a mobile telephone system.

The operation of packet interpolation requires that each telephone has a speech detector to ensure packets are generated only when speech is occuring. For every packet, a request must be sent to a central controller which allocates space for the packet in the speech channel. Clearly this results in a significant load on the request channel which must therefore have a correspondingly high capacity. This aspect is investigated in chapter 3 where the gain in the number of simultaneous conversations, achieved through interpolation, is determined.

Although interpolation can increase the efficiency of telephone systems, it also introduces certain degradations. Potentially large delays occur in the formation and transmission of a packet. In fact, in practical systems, there is a finite probability that a packet will not be transmitted at all. In this case the speech in that packet is lost and a glitch is said to have occurred.

The probability of a glitch in any particular system can be obtained from queue theory as shown in chapter 4. Several authors [40,50-53] have investigated this queue problem, however, there are some discrepancies in the solutions derived. These difficulties are resolved and extensions to the theory required for the mobile telephone system are considered in chapter 4.

1.4 Comparison of Mobile Telephone Schemes

In chapter 5 a computer simulation of the mobile telephone system is described. This provides a check on the validity of the theory and also enables other previously ignored aspects of the system to be considered. The simulation results suggest methods for optimization within a given set of performance criteria. The subjective effects of glitches are also investigated and it is shown how these can be minimized by correct design.

To determine the usefulness of a packet mobile telephone scheme, an optimized system is compared with the alternative present technology in chapter 6. The most efficient mobile telephone systems now in existence use the small cell concept. Here voice circuits are provided on individual radio channels as in the older FDM schemes, but transmission power is kept low so that the channels may be used simultaneously by several telephones, separated by sufficient distances. There results at least a five-fold increase in the number of voice circuits available within the allotted RF bandwidth.

At present small cell systems are being implemented in the U.S.A. by the American Telephone and Telegraph Company [3,54-56], and by the American Radio-Telephone Service Inc. [4,57,58]. A small cell system is also being installed in Japan [2,30,59]. These systems are compared theoretically and practically with the proposed digital system.

A variation on the digital scheme, considered in chapter 7, employs a simple TASI arrangement without packets. This is analysed and compared to all other systems.

1.5 Further Applications

Digital speech interpolation has been proposed for use in PCM telephone lines as described earlier. It is possible to employ a packet system in this context and the results obtained for the mobile telephone system may be applied directly to this new application. Differences in the two systems, however, enable an alternative arrangement in the PCM case and this proves to be beneficial.

In chapter 8 a packet PCM system is designed and compared with a TASI scheme without packets. Other proposed advanced PCM interpolation schemes [44,46,48] are also discussed. Finally, the use of packet speech interpolation in the fields of satellite communication and integrated speech and data networks are briefly considered.

2. A DIGITAL MOBILE TELEPHONE SCHEME

2.1 ALOHA Schemes

Consider a mobile telephone system employing digital transmission over a wideband radio channel. Time division multiplexing is used to provide each active source with a voice circuit. The easiest means of achieving this in such a distributed network is for transmission to take place in packets holding up to a second of recorded speech. This minimizes the amount of switching involved and ensures that little time is wasted due to the radio propagation delay.

At each telephone, speech must be digitized, recorded and formed into a packet with addressing information and possibly error correcting code. The time of the speech stored in each packet is termed the packet length. In transmission, the packet occupies only a small fraction of this time since it is sent at a bit rate far greater than the recording rate. This enables packets from many other sources to be transmitted while the subsequent packet is being formed.

Packets are initiated when speech begins and therefore their transmission times are random. Since packets from all sources must be transmitted on a single channel, some time division multiple access protocol must be used to prevent information loss when packets collide. The simplest technique is that of pure ALOHA. This employs a positive acknowledgement scheme in which an acknowledgement packet is transmitted whenever a speech packet is successfully received. The acknowledgement packet contains the address of the telephone from which the speech originated and is transmitted to the telephone on a return radio channel. If, within a certain time after a packet transmission, the telephone has not received an acknowledgement, it assumes that the speech packet has collided with one from another telephone. It therefore retransmits the packet after a random delay. This delay ensures that collided packets do not inevitably collide again.

If the input of packets to the speech channel is assumed to be completely random and derived from an infinite number of sources, the input process can be modelled by a Poisson distribution [52]. The channel throughput, S, is defined as the average number of correctly received packets per packet transmission time. The channel traffic rate, G, is defined as the average number of packets (both new and previously collided) transmitted per packet transmission time. It has been shown [60] that under equilibrium conditions where the throughput equals the average input rate, the relationship between S and G is

$$S = G e^{-2G} \tag{2.1}$$

A useful description of the performance of an ALOHA system is the relation between average packet delay and throughput. The average delay is directly related to the average number of transmissions a packet requires for successful reception. This is given be $\frac{G}{S}$ and is easily derived from (2.1). Figure 2.1 shows how this quantity varies with throughput.

It should be noted that if the transmissions were perfectly scheduled so that no collisions took place, the channel could be used at its maximum capacity and the throughput would be 1.0. However, there is in reality no possibility of the throughput exceeding 0.184 ($= \frac{1}{2e}$). If at any time the input rate exceeds this value the system becomes unstable, the throughput reduces toward zero and the delay increases toward infinity.

The mechanism behind this collapse is as follows. At low input

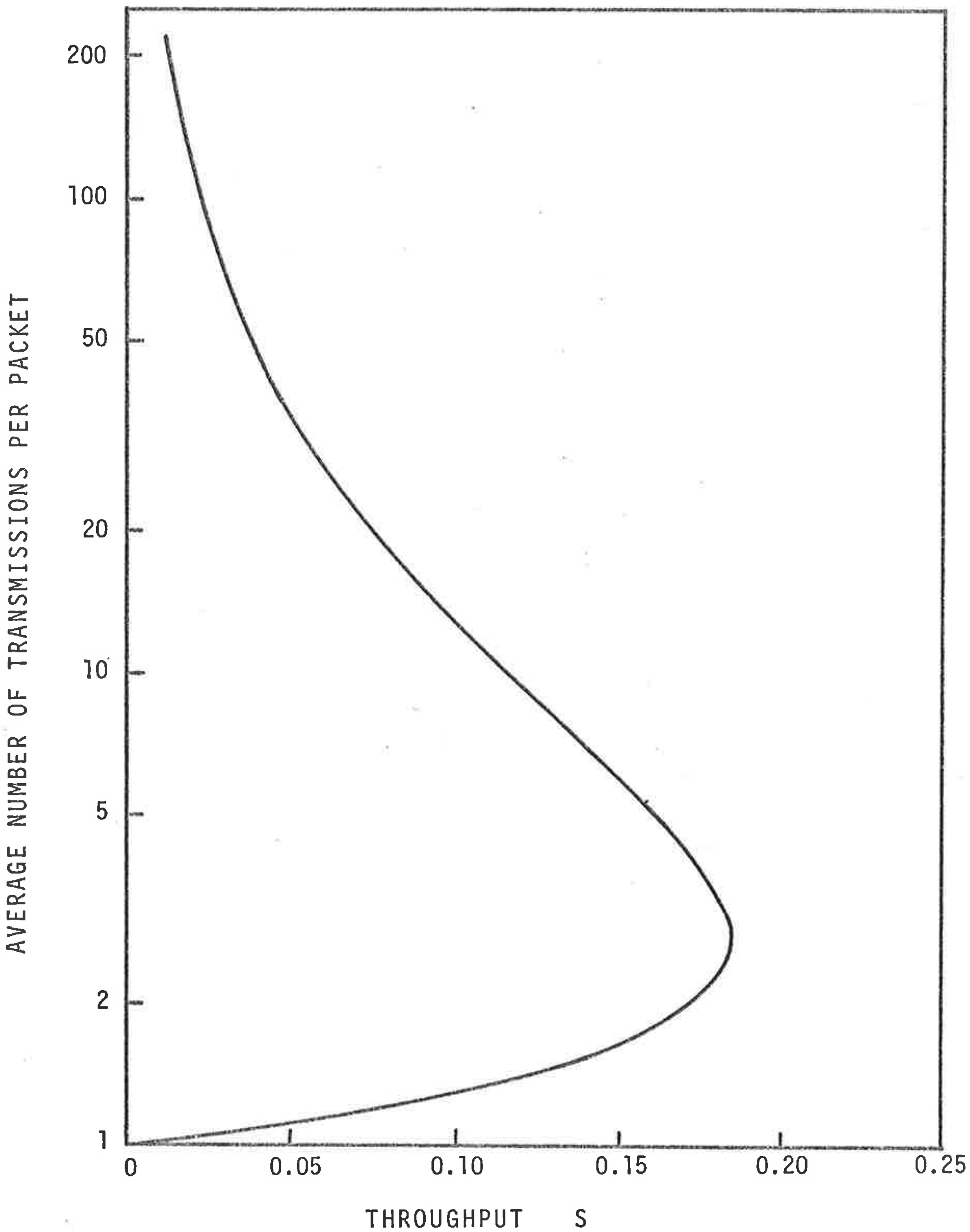


Figure 2.1 Characteristic behaviour of a pure ALOHA system

rates the throughput equals the input. An increase in the input results in greater numbers of retransmissions per packet and increases the average delay. When the input rate reaches 0.184, the total traffic rate, G, is 1.0. Then, even a small increase in input rate overloads the channel and sets off a chain reaction with packets colliding and being retransmitted repeatedly. Eventually, virtually every packet transmitted collides and there are almost no successful transmissions.

An improvement on this simple or "pure" ALOHA system results if a time base is established and the radio channel is divided into slots of length equal to the transmission time of a packet. Users are required to be synchronized and may only begin a packet transmission at the start of a slot. This reduces the probability of a collision and results in a throughput-traffic rate relation of

$$S = G e^{-G} \qquad (2.2)$$

In figure 2.2 the throughput-delay curve for "slotted" ALOHA is shown together with that of pure ALOHA. As a result of slotting the radio channel, the maximum throughput is doubled to 0.368 ($= \frac{1}{e}$). However the possibility of a catastrophic breakdown still exists.

Equations (2.1) and (2.2) are in fact only approximations due to the Poisson input assumption used in their derivation. In practice the retransmitted portion of the input is far from random and hence does not have a Poisson distribution. Consider the packet transmission procedure in the slotted ALOHA case. After a telephone transmits a packet it waits a time-out period of length R slots, for an acknowledgement. If at the end of this period, no acknowledgement has arrived, it waits a random number of slots up to a further K before retransmitting.

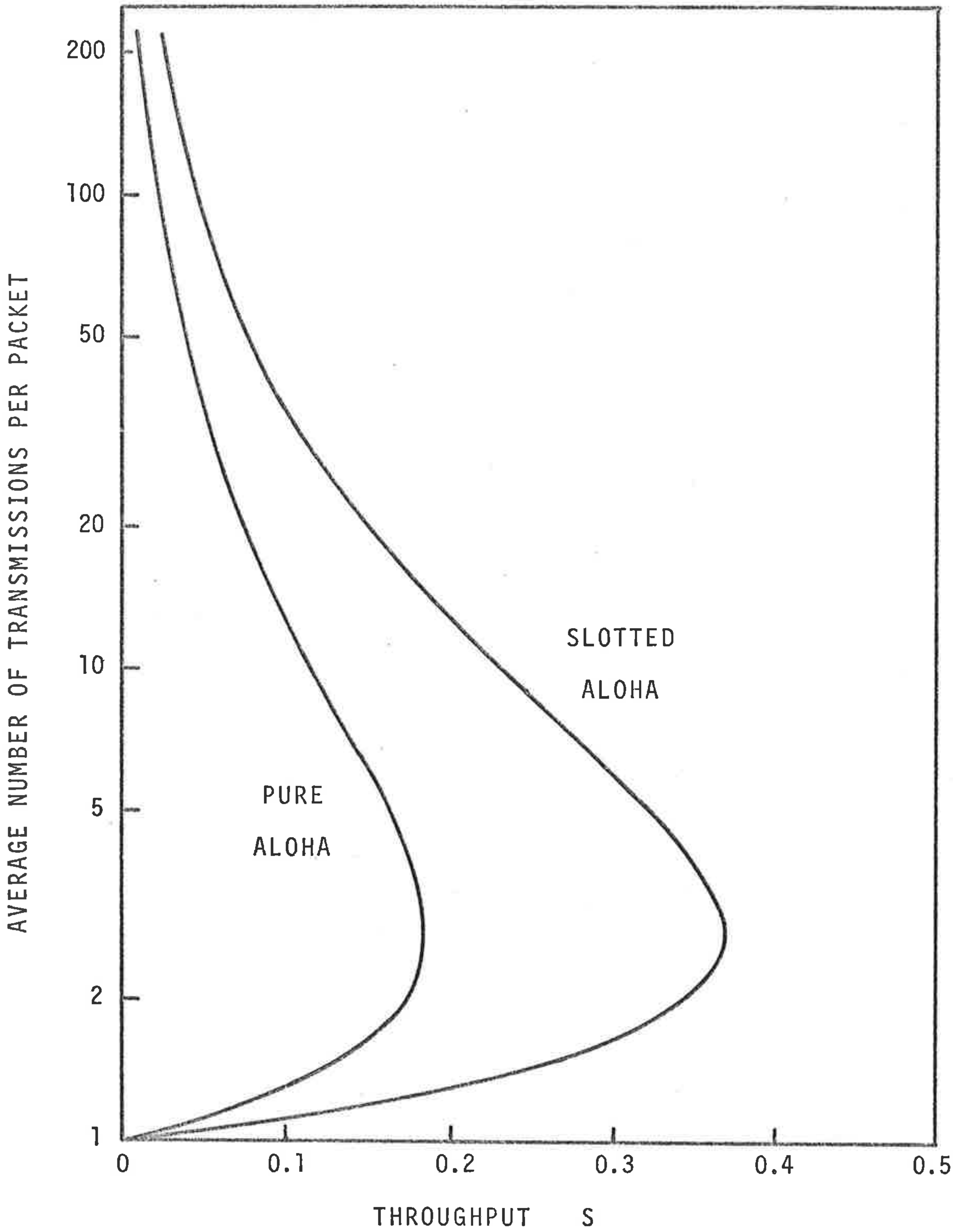


Figure 2.2 Characteristic behaviour of ALOHA systems

An analysis incorporating this complex input has been performed by Kleinrock and Lam [18,61]. They prove that the exact nature of the delay-throughput relation depends upon K as shown in figure 2.3. In fact K may be used to stabilize ALOHA systems to a certain degree as will be seen later.

Clearly neither pure nor slotted ALOHA are suitable for a mobile telephone system. The low input rate required to avoid a system break-down means that the RF bandwidth would be used inefficiently. This must be avoided at all costs. Several proposals have been made for variations on the ALOHA schemes which result in more efficient channel usage. The most important of these are carrier sense multiple access (CSMA) and reservation ALOHA.

In a CSMA system each user senses the RF channel for a short time before transmission to determine whether any other user is transmitting. Various procedures may be adopted for rescheduling the transmission if the channel is occupied. This scheme was analysed by Kleinrock and Tobagi [20] who found that the delay-throughput curve was similar in shape to that of other ALOHA schemes but that the maximum throughput was 0.857. Unfortunately this figure decreases rapidly if even a few users' transmissions cannot be detected by the remainder. In a mobile telephone system, where low power transmitters and omnidirectional antennas would be used, this situation is virtually certain to occur.

A solution to this hidden user problem is to have the receiving station transmit a tone on a separate radio channel, whenever it detects a packet transmission. Users listen to this busy tone channel to determine whether the main channel is in use. The resulting maximum system

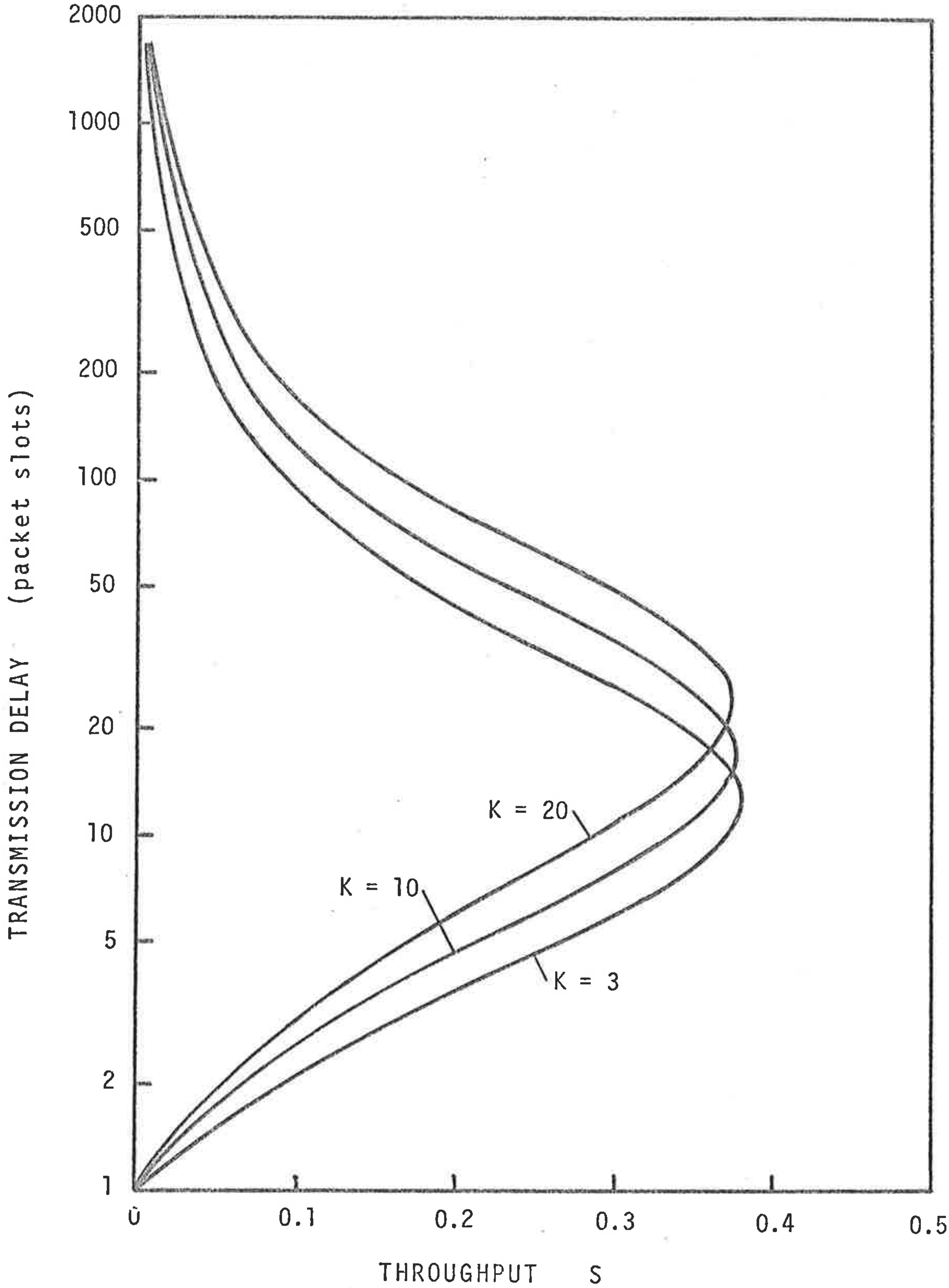


Figure 2.3 Transmission delay in a slotted ALOHA system : R = 8

throughput is 0.720, though in practice a lower rate must be used to avoid overload. This scheme could be used in a mobile telephone system. It is considerably superior in efficiency to slotted ALOHA but requires an extra radio channel.

Reservation ALOHA also requires an extra radio channel; this time used by the telephones to carry their requests. The main channel is entirely controlled in this scheme and may be used at its full capacity. The requests however are still transmitted at random and hence the request channel must use one of the ALOHA protocols. This technique was analysed in detail by Kleinrock and Tobagi [21] who found that in certain cases the system throughput could exceed 0.9. Because of this high efficiency and also the versatility that multiple radio channels provide, this technique is the most suitable for mobile telephone systems. It will therefore be used in all the systems presented in this thesis.

Before these schemes are considered in detail, it is necessary to investigate two fundamental aspects. The first of these is the bit rate necessary to digitize speech at a quality appropriate for a mobile telephone network. The second is the digital capacity that can be achieved on an RF channel in the urban environment. These two quantities influence the choice of virtually all other system parameters and in particular they together define the number of available voice circuits. This will be seen to be a major determinant of the system performance.

2.2 Digital Encoding of Speech

Digitized speech often has the disadvantage of requiring much greater transmission bandwidth than the original analogue version. To limit the bandwidth required to a reasonable value advanced encoding techniques must be employed. Such techniques fall into one of two broad categories: waveform coding methods and the more complex analysis/synthesis methods. These will be examined in turn.

The most common form of waveform coding is pulse code modulation (PCM). Here a speech signal is sampled about 8000 times per second and the amplitude of each sample is digitally encoded into 7 or 8 bits [7]. To increase the permissible dynamic range of the input and yet retain good signal to noise ratios at low levels, non-linear encoding is used. This involves spacing the quantization levels in an exponential manner so that effectively the logarithm of the input signal is linearly encoded.

This technique is termed LOG-PCM and it is universally used for digital speech transmission in the present telephone network. It does however require a relatively high bit rate of 56 kbit/s to 64 kbit/s and is quite sensitive to errors in the transmission channel. Differential encoding techniques exist which provide comparable quality at half the bit rate and which are much less sensitive to errors.

The basic structure of a differential speech coder is shown in figure 2.4. It is the difference between adjacent speech samples which is encoded with this arrangement. The output from the quantizer is transmitted to the receiver and also fed back to a predictor which reconstructs the waveform, just as the receiver does. This reconstructed (and delayed) signal is subtracted from the input to produce the error signal for quantizing. The performance of differential pulse code modulation (DPCM) is superior to that of PCM at any bit rate [62].

Improvements may be made to DPCM by tailoring the encoder to

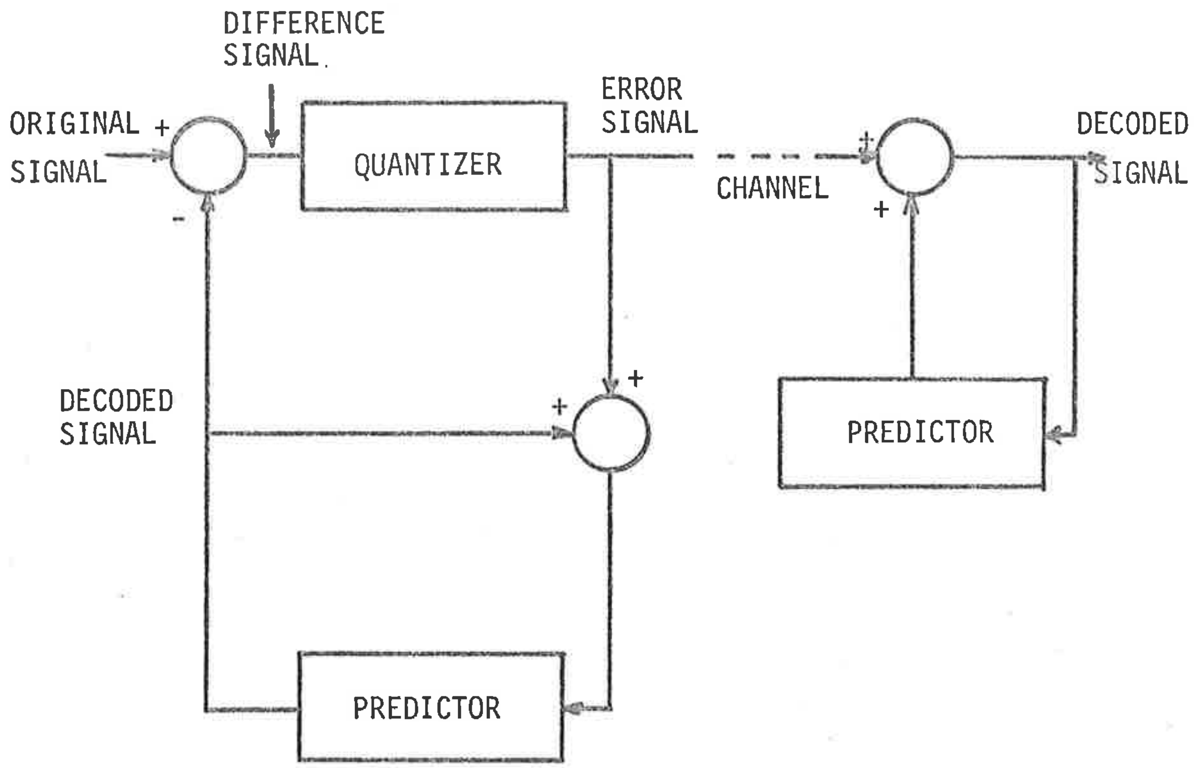


Figure 2.4 Differential encoding of speech

follow the speech characteristics. This involves the use of one or both of adaptive quantizers and adaptive predictors which vary their parameters, with the input, to maintain an optimum signal to noise ratio [63,64]. With any of these either forward or backward estimation can be used. In the latter case the quantizing levels or predictor coefficients are calculated from past samples by both the encoder and the receiver. Forward estimation involves using a sequence of up to 256 samples to calculate the most appropriate values for this group prior to encoding. A description of the quantizer and predictor parameters used, in addition to the encoded speech, must be forwarded to the receiver.

All of the adaptive schemes outperform their non-adaptive counterparts and predictably the most complex perform best. A signal to noise ratio (SNR) comparison is shown for all of these schemes in figure 2.5. Differential encoding by itself provides a 7dB advantage over LOG-PCM and this increases to 11dB with adaptive prediction. Adaptive quantization provides an extra 7dB without adaptive prediction and 5dB with adaptive prediction. Further, where forward adaptive techiques are used and asynchronous coding is possible, entropy coding [64] can be used to give a further 2 to 3dB improvement.

A DPCM system incorporating all of the above features can at a bit rate of 24 kbit/s, perform as well as a LOG-PCM system at 48 kbit/s. The resulting SNR of 27dB is sufficient to provide good quality telephone speech since, at a SNR of 21dB, the quantization noise is barely perceptible [45,65]. Forward estimation techniques are well suited to a packet system where speech has to be recorded for a packet length in any case.

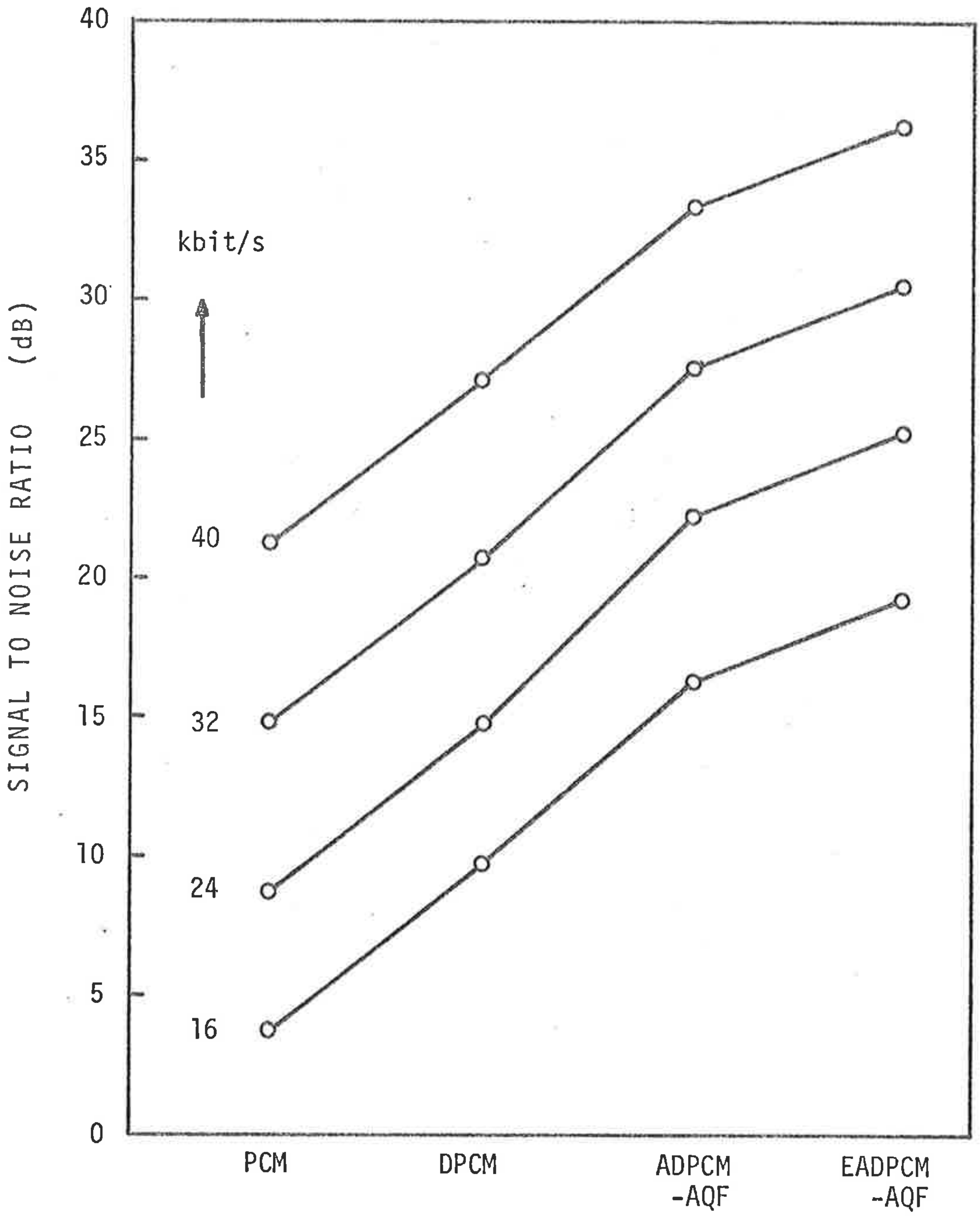


Figure 2.5 SNR comparisons for digital encoding schemes

PCM = Pulse code modulation

DPCM = Differential PCM

ADPCM-AQF = Adaptive DPCM with forward adaptive quantizing

EADPCM-AQF = Entropy coded ADPCM-AQF

Another common waveform coding technique is delta modulation (DM). This is very similar to DPCM but only one bit is used to encode each error signal. To compensate for this, the sampling frequency is increased to between 16 kbit/s and 64 kbit/s [62]. The basic structure of a DM encoder is again as shown in figure 2.4 and again both the quantizer and predictor can be made adaptive with forward or backward estimation.

The SNR's of adaptive delta modulation (ADM) encoders are generally a little smaller (by 3dB to 6dB) than that of the equivalent DPCM encoders. However, the former are very rugged in their ability to withstand channel errors. With some ADM arrangements intelligible voice with speaker recognition may be attained with a bit error rate of 10% [66]. It is for this reason that ADM at bit rates of 24 and 32 kbit/s is being used for voice communication with the space shuttle.

The final waveform coding technique to be considered here is adaptive transform coding (ATC). This involves storing between 16 and 128 speech samples and performing a transform on them prior to coding. Zelinski and Noll [67] showed that an ATC coder using a discrete cosine transform and forward adaptive quantization and prediction could outperform the equivalent DPCM coder by 3B to 6dB. In fact this technique can provide good quality speech at a bit rate of 16 kbit/s [68].

The second approach to speech encoding involves speech analysis and synthesis. This is embodied in linear predictive coding, vocoder and formant based techniques [69-71]. These can provide quite intelligible speech at bit rates as low as 2.4 kbit/s [8], but only at considerable cost and complexity. Substantial computing power such as a dedicated 16 bit minicomputer is required for real time speech encoding in the most complex versions. Significant amounts of other hardware may also be needed. Clearly this is out of the question for a mobile telephone scheme in which an encoder is required by each telephone.

Thus, in conclusion, speech encoding at bit rates below 30 kbit/s is achievable at the present time with waveform coding techniques of moderate complexity. Speech so encoded is quite suitable for mobile telephone applications as long as the bit error rate is small. If this is not the case then error protection and consequently a larger encoding rate is required [65].

In light of the above considerations it would seem that a suitable choice for the speech digitization bit rate is 24 kbit/s. This figure reflects the need to have as small a bit rate as possible and yet still provide good quality speech. It should be kept in mind that bit rates of 16 kbit/s are quite feasible and that with advances in integrated circuit technology, some of the analysis/synthesis techniques may become considerably cheaper to implement. Thus in the future the use of bit rates lower than 24 kbit/s must be considered a strong possibility.

2.3 Digital Transmission in the Urban Environment

The provision of high rate digital radio transmission in the urban environment is a very complex process. Indeed a thorough study of the subject would require years of research and experimenting. Its treatment in this thesis will therefore of necessity be restricted to a discussion of the major problems and solutions applicable in a mobile telephone situation. The most critical feature is the channel capacity available. Methods of providing capacities up to 1 Mbit/s will be examined.

Mobile telephone systems operate in the ultra high frequency (UHF) band between 500 MHz and 2 GHz. At such frequencies transmissions in the urban environment are subject to multipath propagation, high range loss, background noise, man made noise and interference [36]. Undoubtedly the major problems are caused by multipath propagation.

A radio signal received in the urban environment consists of many components reflected from large buildings. Because of these obstructions it is very unlikely that a direct path signal from the transmitter will be present. The arriving components are out of phase and spread in time and when they combine destructively the resultant signal amplitude can decrease by up to 40 dB. This is termed fading. Figure 2.6 [72] shows a typical variation in signal amplitude as a receiver moves. Cox [28] found that this variation was well approximated by a Rayleigh distribution as shown in figure 2.7.

A moving vehicle will pass through fades at a rate directly dependent upon the vehicle's speed. There is on average a fade every half wavelength of the carrier signal (every 15cm for a 1 GHz signal). In addition to this the mean of the signal strength varies over greater distances due to shadowing by buildings. The mean follows a log normal distribution with a variance of around 6 dB [30].

Another aspect of multipath propagation is the high probability of receiving echoes of the original signal 8 or 9 ms after the first arrival. This is termed excess delay and several examples have been documented by Cox [27-29]. The type of results obtained are illustrated in figure 2.8. In cases where large peaks in the signal occur at delays of several milliseconds they are often the result of reflections from a single large structure. However broad peaks at large delays also occur

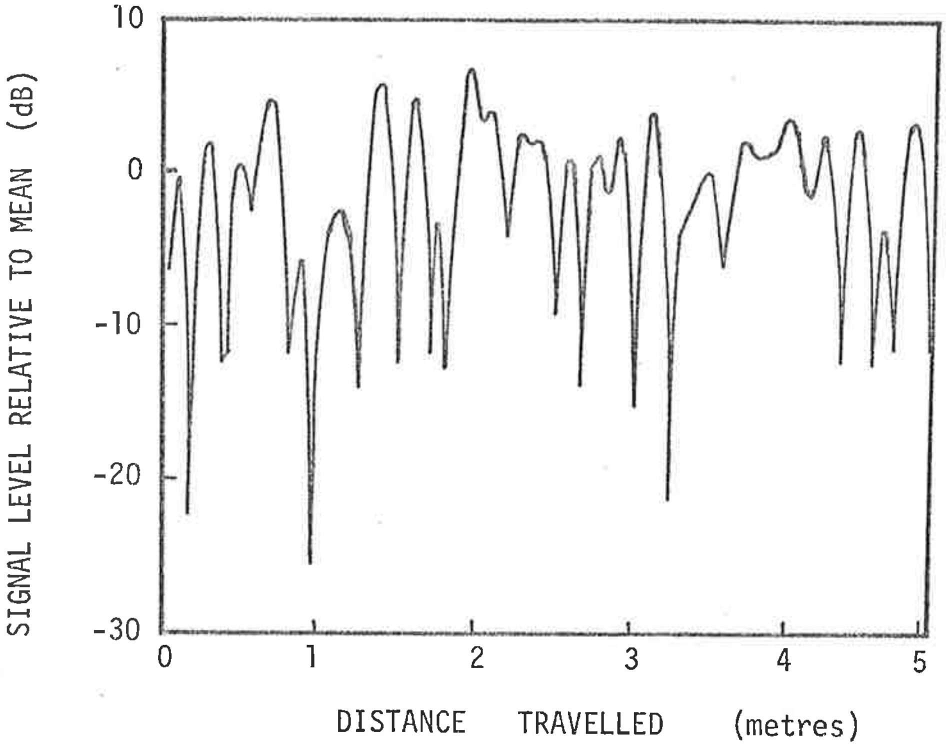


Figure 2.6 Example of variation in received signal level

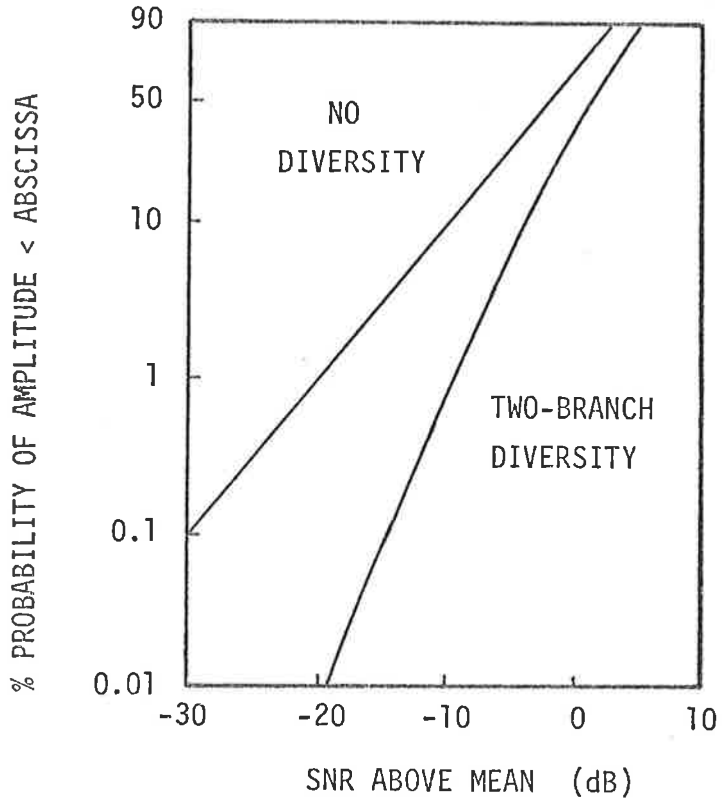


Figure 2.7 Distribution of received signal level

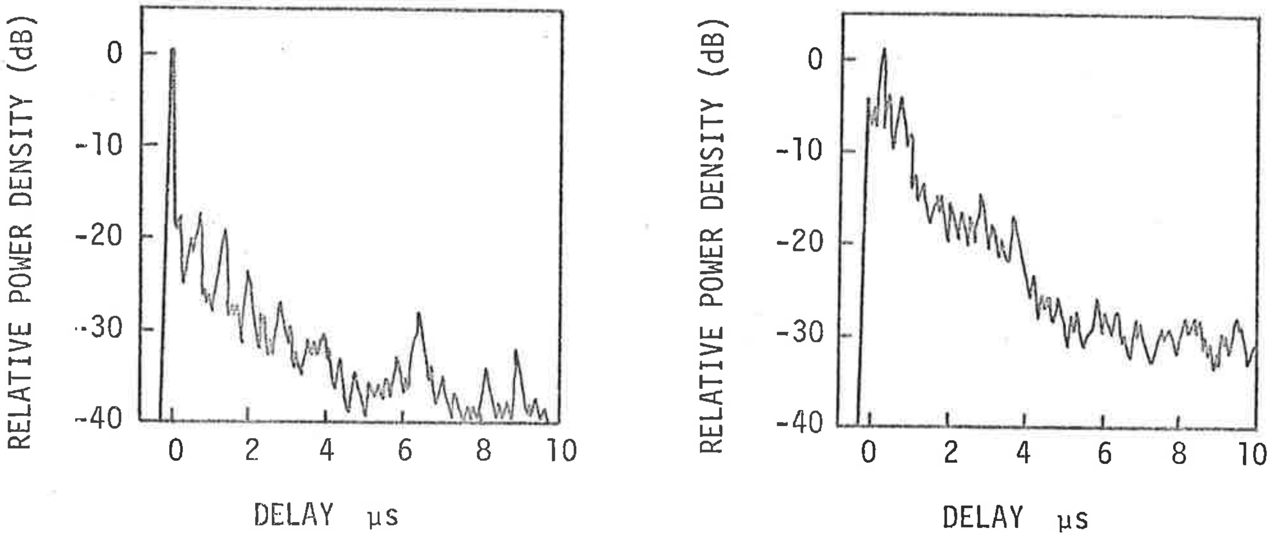


Figure 2.8 Example of delay spreads in urban areas

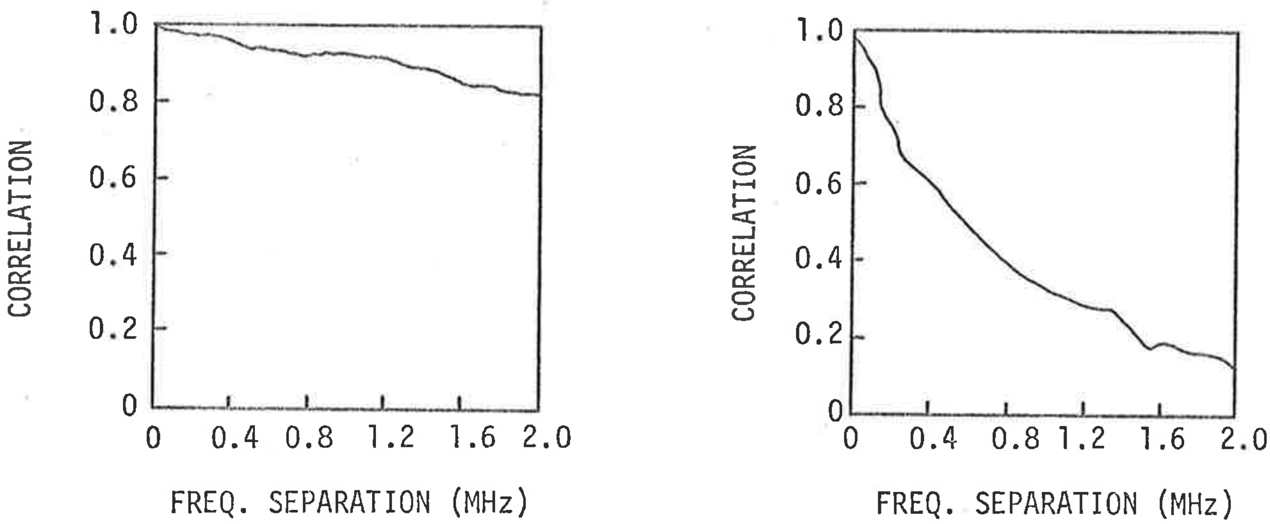


Figure 2.9 Frequency correlation curves for delay spread examples in figure 2.8

and are associated with a multitude of unresolvable reflecting surfaces. Needless to say such characteristics result in excessive intersymbol interference at high baud rates. Fralick and Garrett [36] suggest that multipath effects limit the achievable baud rate to 100 to 400 kbaud.

Related to the excess delay is the correlation (or coherence) bandwidth. The resultant signal at any position depends upon the signal frequency, thus, two signals of different frequency will not necessarily fade at the same place. The extent to which signals of different frequency behave in the same manner is measured by their correlation. This may be determined by taking the Fourier transform of the excess delay curves. The correlation bandwidth is the separation between two frequencies which have a correlation of 0.9. Figure 2.9 indicates how this can vary. Cox [29] concludes that it can be as low as 20 kHz in inner city areas, while in more favourable positions it may exceed 1 MHz.

The range loss urban environments is much greater than in free space. It has been found to be inversely proportional to distance to the power 3 or 4 rather than 2. For practical ranges therefore it is 40 to 50 dB greater than the free space loss [34,35]. This limits the useful range for transmission in urban areas to about 5 km.

Noise in the urban environment may be characterised as background or as man made. In the UHF band background noise is primarily of thermal origin and has a uniform spectral density of around -170 dBm/Hz and a Gaussian amplitude distribution [36]. Far more important than this however is man made noise of which the major component is impulsive noise. Impulses some mainly from automobile ignition systems but other potential sources include the power distribution system and electric trains. The average level of all man made noise can exceed that of the background noise by 20 to 40 dB [35,73]. More importantly however, from a digital point of view, is the fact that noise impulses can often exceed the background noise by 60 to 80 dB [36].

It is impractical to attempt to overcome these impulses by increasing transmitter power. The errors that they cause must simply be accepted, or if possible corrected by using coding techniques. Fortunately the duration of the impulses is of the order of only nanoseconds [73] and the number of errors created can be minimized by correct receiver design. The interval between impulses in heavy traffic areas is fairly linearly distributed between 1 and 10 ms [36]. At a bit rate of 1 Mbit/s this will result in an error rate from this source alone of 10^{-4} in the worst areas.

One method of achieving high data rates in lostile environments is to use spread spectrum techniques [34]. These operate by transmitting information in an RF bandwidth many times that actually required. When the signal arrives at the receiver it may be completely submerged in noise, but by "despreading" the signal it can be restored to a useful level. The amplitude gain achievable with spread spectrum techniques is simply the ratio of the bandwidth used to that required [74].

The obvious problem with this method is its very inefficient use of the RF spectrum. There are however two possible solutions. Firstly, since the transmitted power required is so small, it is quite possible that the frequencies could be used for other purposes at the same time. Secondly there is room within any one system for many people using orthogonal codes. Such code division multiple access would enable potentially hundreds of sumultaneous conversations in a bandwidth of several tens of MHz [33,74].

Spread spectrum techniques have been utilized in a multihop, multiple access, packet radio computer network termed PRNET [15,34,75,76]. Here repeaters are used in the radio path between the mobile terminals and control stations. Bit rates of 100 kbit/s are used on the terminal to repeater link and of 400 kbit/s between repeaters and stations. The repeaters allow lower power terminals and increase the system reliability. This network is designed to operate over 100 square miles and provides virtually error free performance (one undetected error in 10^{10} packets). It is at present being tested in San Francisco, USA.

Such spread spectrum systems would be capable of providing a mobile telephone service to a large number of people within an urban area. However, they are undoubtedly quite wasteful of the RF spectrum and probably could not match the efficiency of current alternative systems (see chapter 6). To achieve high bit rates without spread spectrum techniques it will be necessary to use multiple parallel channels and to take advantage of the opportunities that packet transmission provides.

Excess delays of up to 10 ms appear to limit the transmitted symbol rate to 100 kbaud. Also high noise levels will probably prevent multi-level coding. Thus to achieve a bit rate of 1 Mbit/s it is necessary to use say 10 parallel channels each of 100 kbit/s capacity. Unless precautions are taken with this arrangement, fading could easily result in a bit error rate exceeding 0.1. The most reliable method of reducing fading is by the use of diversity.

With a two branch equal gain diversity system signals received from two antennas are cophased and added. Bacause of the physical separation of the antennas it is very unlikely that both will be in a fade at the same time. Thus the likelihood of deep fades is reduced as shown in figure 2.7. With an allowable loss of 15 dB from the mean signal strength the probability of having an unusable signal is less than 0.1% and the expected duration of such an event is about 1 ms (at 60 km/hr) [72].

Assuming that each packet contains 1000 bits then the packet transmission time is also 1 ms. Thus whenever any of the 10 parallel channels are in a fade its contents will almost certainly be lost entirely. Under the conditions assumed above the probability of such an event is around 10^{-3}. Thus if the 10 channels are well spaced in frequency so that their correlation is low, the probability of one or more of them fading and thereby destroying the packet is unacceptably high at around 10^{-2}. However, if for instance one redundant channel is added, containing a checksum for the other 10, then any one can fade without affecting the packet. The probability of losing a packet is then reduced to around 10^{-4}.

An alternative to adding an extra channel is to require that any packets destroyed by fading be retransmitted. This adds slightly to the channel traffic and increases the delay until final packet reception but it avoids a 10% increase in the RF bandwidth. With this strategy the correlation between parallel channels should be as large as possible to reduce the probability of a packet being destroyed. Multiple retransmission can then reduce the packet loss to insignificant proportions.

If 15 dB of loss is not allowable or diversity cannot be used the fading frequency increases significantly. The only way then to reduce fading is to ensure that some specular (direct path) component is present in the received signal. This can be achieved by correctly positioning, and perhaps increasing the number of repeater stations so that a line of sight path to a station exists from virtually all points in the service area. Nettleton and Cooper [33] showed that this can reduce the

SNR required for a given fade probability by more than 5 dB.

Finally it is worth noting that the reception at the repeater stations is far better than that at the mobiles. Even if diversity cannot be implemented for each telephone it can certainly be used at the far less numerous stations. Also it is very likely that the stations will be positioned high on buildings where the man made noise (and in particular the impulse component) is at a much lower level [36]. Thus less power is required by the mobile transmitter to produce an acceptable SNR. This is a factor of particular importance in hand held mobile telephones.

In summary, the radio environment in urban areas is extremely hostile for digital transmission. However, it does appear that a bit rate of 1 Mbit/s can be achieved in a packet system. With diversity and packet retransmission the problems of fading can be minimized and intersymbol interference can be avoided by the use of parallel channels. The bit error rate must be expected to be around 10^{-3}, with most of the errors being due to noise impulses from automobile ignitions, If, for the speech encoding technique used, this error rate proved to be unacceptably high then error correcting code would be needed. Because of the nature of errors caused by the impulses, single error correcting codes should be particularly efficient.

For the remainder of this thesis a channel capacity of 1 Mbit/s will be assumed. The difficulty of achieving this however will be kept in mind.

2.4 A Basic Mobile Telephone System

In a digital mobile telephone system the telephones are scattered randomly throughout the service area and are connected via radio channels to a network of repeater stations. Repeaters provide more reliable radio communications by minimizing the radio path length and maximizing line of sight coverage. The radio paths end at the stations which are connected via landline (or microwave link) to a control centre incorporating a computer. Here circuit or slot assignment and packet switching take place and interconnection to the remainder of the telephone network is provided.

With reservation ALOHA four separate radio channels exist between the mobile telephones and the stations. These are the main or speech channels, in which packets containing speech are transmitted in both directions, and the smaller request and acknowledgement channels. These last two will be referred to collectively as the information channels.

Consider the sequence of events when a mobile telephone wishes to make a call. As soon as its handset is raised the telephone will transmit a packet containing its identification on the ALOHA request channel. If this packet does not collide an acknowledgement packet will be returned by the station receiving the strongest signal. Signal strength comparisons can be made either at the control centre or at a station, in the immediate area, equiped for the task. In the second case, some interconnection between stations would be required. This expense however would be justified by the need to continuously compare signal strengths for moving telephones.

Upon receipt of the acknowledgement packet the telephone issues a dial tone and, when dialling is completed, it transmits on the request channel, another packet containing the dialled number. An acknowledgement packet from the station is returned immediately and when a circuit to the called party is established, a packet denoting this follows. In response, the telephone gives the usual ringing signal. When the called party answers, two slot numbers are transmitted in a further packet on the acknowledgement channel. These denote the slots, in the forward and reverse speech channels, which contain speech packets from the mobile telephone and the station respectively.

Now consider the simple situation in which each slot number repeats every packet length. The telephone must record speech for a packet length and then form it into a packet in time for transmission in the assigned slot. Clearly this is simply time division multiplexing of active telephones in the speech channel. The request channel is not required again until the call is terminated when a suitable notification packet is transmitted.

If any of the mobile telephone's request packets are not correctly received because of collision, fading or excessive numbers of errors, then no acknowledgement packet is transmitted. The telephone then retransmits the packet after the normal delay thereby ensuring that no information is lost. Similarly, if any of the station's acknowledgement packets are not correctly received the request is repeated and an acknowledgement is retransmitted.

To prevent problems with the loss of a ringing packet from the station, a telephone can repeat the dialled number packet after a time-out period. The station can easily detect if the slot assignment packet is not correctly received, by the absence of any transmission in the appropriate speech channel slot. Finally, if any speech packet is lost through fading it can be repeated in a later speech slot (if one is available). This arrangement is therefore quite reliable and flexible. Virtually no packets should be lost and the only cost is an occasional small delay, which will cause few problems.

If a call from the main telephone network is destined for a mobile subscriber, a packet containing the appropriate identification number is transmitted on the acknowledgement channel. This cannot be done from every station at the same time because of possible intersymbol interference. However simultaneous transmission is possible from stations sufficiently separated. Probably an average of five to ten transmissions will be required to obtain a response from the target mobile telephone. When this is done the slot numbers are transmitted and the call proceeds as before.

2.5 Design of a TDM System

To completely specify the characteristics of the above mobile telephone system, the information channel capacities must be determined. These depend upon the number of bits in the request and acknowledgement packets. A request packet contains the following fields

- synchronization or preamble
- source identification
- destination identification
- packet information
- error protection

These will be examined in turn.

Since the request channel is of the slotted ALOHA type, system synchronization must be closely maintained to ensure correct transmission timing. This can be done by accepting timing from packets in the acknowledgement channel. Each telephone must receive and decode all of these packets in any case to determine if it is being paged. Timing obtained in this manner however cannot be exactly correct because of variable propagation delays.

The round trip radio propagation delay is 6.7 μs for each kilometer of distance to the appropriate station. In addition, when multipath effects are severe there may be a further delay of several microseconds until the strongest part of the signal arrives. Thus synchronization is required for every packet. The PRNET mentioned in section 2.3 uses a 48 bit preamble in a packet length of 2 kbits [34]. Hence between 20 and 50 bits will be required here.

Source identification should take no more than 20 bits. This allows over a million subscribers which should be sufficient for any imaginable system. The actual telephone number of a mobile telephone can be stored with the identification number at the control centre.

There are advantages from an organization viewpoint if request packets are directed to particular stations. This prevents multiple copies of a packet being presented to the central control computer and requires no more than 10 bits for station identification. A station that received a packet not destined for it would discard the packet only if the correct station had also received it.

The function of any particular packet must be identified. However, since there are very few different types of packet this could be encoded within 4 or 5 bits. It is also possible that timing information may be required, for instance to denote when a packet was first transmitted. An allocation of 10 bits for this purpose should provide sufficient accuracy. Thus, in total, 45 bits are required to convey the information component of a packet.

Request packets containing the dialled number will require special arrangement. Dialled digits must be stored as binary coded decimal (BCD) to avoid conversion to true binary at the telephone. Thus the normal seven digit telephone number occupies 28 bits. Rather than increase the packet size significantly, just to handle dialling packets, it is better to rearrange the packet fields and use more than one packet. For instance, if one information bit is dedicated to denoting a dialling packet the others become free. Also the timing information can be abandoned and the destination identification abbreviated to provide space for four or five digits. Two packets are therefore adequate in normal circumstances but if more than seven digits are dialled extra packets will be needed.

The length of error coding depends upon the number of bits to be checked and upon the complexity of the coding technique. Simple error detection may be used with a retransmission procedure. However, this can result in significant increases in the channel traffic and delay. With a bit error rate of 10^{-3} the probability of one or more errors occuring in the 45 information bits approaches 5%. The number of retransmissions drops drastically if even a samll amount of error correction is incorporated.

Repetition of the information an odd number of times is the simplest form of error correction. For example, with a best-of-three arrangement 90 error bits are required and the probability of a packet error is reduced to around 10^{-4}. Although this represents an adequate figure when combined with packet retransmission, the cost in bits required is clearly substantial. More complex coding techniques are far more efficient.

Cyclic or recurrent codes can correct both random and burst errors in blocks of information. Typically these codes will correct a certain number of errors and simultaneously detect a greater number. This property is ideal for a packet system. A total of "e" errors can be corrected in a block of "m" bits with "k" error correction bits given by

$$k \geqslant \lg \left[1 + n + \frac{n(n-1)}{2} + \cdots + {}_nC_e\right] \qquad (2.3)$$

where n = m + k and lg = logarithm to the base 2.

For a block of m = 45 bits, the number of error bits required and the resultant probability of an undetected error are shown in table 2.1 for various numbers of corrected errors.

Number of errors corrected	1	2	3
Number of error bits	6	11	16
Total number of bits	51	56	61
Probability of uncorrected error	1.2×10^{-3}	2.7×10^{-5}	5.3×10^{-7}

NOTE Bit error probability = 10^{-3}, m = 45 information bits.

Random errors have been assumed.

Table 2.1 Error probabilities with a cyclic error correction code.

If less complex error correcting codes are employed then more bits are required to achieve the same protection. Thus allowing around 20 bits, a request packet will contain in total between 80 and 120 bits.

Acknowledgement channel packets require exactly the same fields as request channel packets. There are again a limited number of packet types and only those containing slot numbers require special consideration. No more than 10 bits are necessary for a slot number, thus the bits allowed for timing will suffice. To transmit both forward and reverse slot numbers, two packets are needed, or, alternatively, station identification may be omitted. Acknowledgement channel packets will therefore also have between 80 and 120 bits.

It is now possible to calculate the capacity required by the information channels as a function of the speech channel's capacity. To maintain generality the following variables are defined.

Let	W be the capacity of each speech channel	(bit/s)
	D be the bit rate of digitized speech	(bit/s)
	B be the number of bits in a request and an acknowledgement packet	(bits)
	A be the ALOHA factor, i.e. the ratio of the request channel capacity to its average throughput.	

The number of TDM voice circuits in a speech channel, referred to as the circuit capacity, is

$$C = \frac{W}{D} \qquad (2.4)$$

If it is assumed that a telephone conversation lasts an average of 100 seconds then the number of calls changing over per second is

$$N_C = \frac{C}{100} = \frac{W}{100\ D} \qquad (2.5)$$

Now there are three or four request packets transmitted throughout a call whether it is initiated or received by the mobile telephone.

To make allowance for a possible extra dialling packet or for a retransmission due to packet loss (not including collisions), a total of five request packets will be assumed for each call. Then the required bit rate in the request channel becomes

$$S_r = 5\ N_C\ B = \frac{B\ W}{20\ D} \tag{2.6}$$

Because the request channel is of the slotted ALOHA type, its capacity must exceed its input rate by a factor A. The minimum value of A is 2.72. In practice a larger value of between 5 and 10 must be used to avoid overload. Thus the final capacity of the request channel as a fraction of the speech channel capacity becomes

$$R_r = \frac{A\ S_r}{W} = \frac{A\ B}{20\ D} \tag{2.7}$$

The values of these variables may be taken as follows

$A = 10$	- request channel throughput is 0.1.
$B = 120$ bits	- largest size is taken for safety.
$D = 25$ kbit/s	- allows 24 kbit/s for speech digitization and 1 kbit/s for packet synchronization and error correction.
$W = 1$ Mbit/s	- see section 2.3.

Then $R_r = 2.4 \times 10^{-3}$.

In the acknowledgement channel there are an average of ten packets transmitted per call (assuming a search for a called subscriber requires on average ten packets). However, this channel is completely controlled and may be used at full capacity. Thus the ratio between the acknowledgement and speech channel capacities is

$$R_a = \frac{B}{10\ D} \tag{2.8}$$

With the above values this becomes

$$R_a = 4.8 \times 10^{-4}.$$

Clearly the theoretical capacities of the information channels are negligible compared to that of the speech channel. In practice larger capacities than those suggested by (2.7) and (2.8) should be used to reduce the transmission time and hence the delay. With a transmission slot length of 10ms the capacity required is only 12 kbit/s. Thus virtually 99% of the total RF capacity is dedicated to carrying speech packets; obviously a very efficient arrangement.

To put this TDM mobile telephone system into perspective it will be necessary to compare it with present day alternative systems. This will be delayed until chapter 6 when another digital system will be available for comparison. It is sufficient at this stage to say that the TDM system is roughly comparable with the non digital systems in performance though obviously very different in implementation.

2.6 Conclusion

In this chapter various aspects in the design of a digital mobile telephone system have been described. A consideration of packet transmission, leads to a choice of a reservation ALOHA protocol for the mobile to repeater station radio link. This results in very high RF efficiency because the random component of the transmissions is restricted to the low capacity request channel.

A study of the various methods available for digitizing speech showed that waveform coding techniques are the most suitable for use here. These can provide adequate quality speech at bit rates down to 20 kbit/s, with modest amounts of hardware. A rate of 24 kbit/s was adopted initially, however allowance must be made for synchronization and error correction in speech packets. To nominally account for this an effective speech digitization rate of 25 kbit/s will be used throughout the remaining work.

Another aspect of a digital system is the limitation imposed upon radio transmission by the urban environment. The most important impairment is multipath propagation. This results in signal fading and multiple echoes at relatively long delays. To overcome this fading it is necessary to have an extra 15 dB of SNR and also to use a diversity reception technique. Fortunately, because of the automatic packet repetition, the remaining fading has no effect on the request and acknowledgement channels. It has been shown here, that fading of speech packets may also be overcome by packet repetition.

To achieve a speech channel capacity of 1 Mbit/s, multiple parallel transmission can be used. This will increase the probability of fading and also the error rate, which emphasises the need for the fade prevention techniques outlined above and also for error correction in speech packets. It is indeed a difficult task to arrive at a procedure which provides the desired 1 Mbit/s capacity without an unacceptable error rate and without sacrificing efficiency in the packet format. However, as long as the probability of incorrect speech packet reception is below 1%, retransmission should increase the quality to a suitable level.

All of the aspects considered above have been brought together in the design of a time division multiplexed mobile telephone system. A very flexible and efficient system has been designed using only 12 kbit/s for each information channel and a speech channel capacity of 1 Mbit/s. A comparison of this scheme with analogue alternatives will be performed in chapter 6.

3. INTERPOLATION OF SPEECH IN PACKETS

3.1 Interpolation

Interpolation of speech is a process through which a number of conversations may be transmitted in a smaller number of channels. This is achieved by transmitting little or no information whenever no speech is occurring. Effectively conversations are interpolated into each others' silence intervals.

The first commercial application of interpolation was the time assigned speech interpolation (TASI) system used on the transatlantic undersea cable in 1960 [37]. This relied upon circuit switching of the analogue speech signals. Since that time several digital interpolation schemes using time division switching have been proposed [44,47]. In the packet schemes considered here, packet switching is used to interpolate speech. This becomes quite complex in a mobile telephone situation.

To avoid recording packets when speech is not occuring, each telephone must possess a speech detector. A packet is begun when speech starts and a packet length is recorded. If at the end of this time speech is still occurring, then a further packet is begun. Note that because of the fixed length of packets some will inevitably contain periods of silence.

A mobile telephone scheme based upon packet interpolation can have the same physical arrangement as the TDM scheme. However, the various radio channels differ in their use. The stream of packets from any one source is not now at a constant rate but varies with speech activity. Thus individual circuits in the speech channel are no longer assigned. Instead all speech slots are treated separately and must be competed for by all telephones with packets ready for transmission.

Each telephone must send a request via the request channel whenever it begins to record a packet. This request is forwarded to the controlling computer which allocates an appropriate slot in the speech channel. The slot number must then be returned to the telephone, via the acknowledgement channel, in time for its packet to be transmitted.

Various aspects of this proposal require considerable study to ascertain the scheme's usefulness and viability. Firstly, the gain in numbers of users through packet interpolation must be determined. This will be seen to be a relatively simple function of the packet length. Secondly, the request channel suffers much greater usage with interpolation as one request must be transmitted for each packet readied. This must be investigated as must the timing arrangements. The latter are much more critical than in the simple TDM system since requests must be satisfied in real time,so as not to disrupt the flow of speech.

3.2 Speech Characteristics

The gain achievable with packet speech interpolation depends upon the efficiency with which speech can be transmitted in fixed length packets. This is determined by the duration of the three elements which characterise telephone conversations: talkspurts, pauses and response times.

A talkspurt is speech by one party which is preceded and followed by speech from the other party.

A pause is a length of silence between two periods of speech within a talkspurt.

A response time is the interval between the end of a talkspurt of one party and the beginning of the other's reply.

The standard conversation interchange consists of a talkspurt by one party which may contain several pauses, followed after a brief silence by a talkspurt from the other party. Non standard interchanges such as interjections and interruptions can be described in terms of negative response times.

The duration of these three elements were measured by Norwine and Murphy [77] on a total of 13,000 seconds of telephone conversation. They presented probability density functions for the length of talk-spurts, pauses and responses, and distribution funtions for the number of pauses in a talkspurt and the length of talkspurts with a given number of pauses (figures 3.1 to 3.5).

These curves are quite complex and must be approximated by well known distributions if a mathematical analysis is required. This is inadequate for speech stored in packets since the actual short term behaviour of the speech is lost. The best way to determine the true efficiency of the packet format is by computer simulation. Fortune-ately, this may be done relatively easily by using the curves in figures 3.2 to 3.5.

The Fortran programme used to simulate speech and place it into packets is shown in appendix C. At each stage in speech generation, a

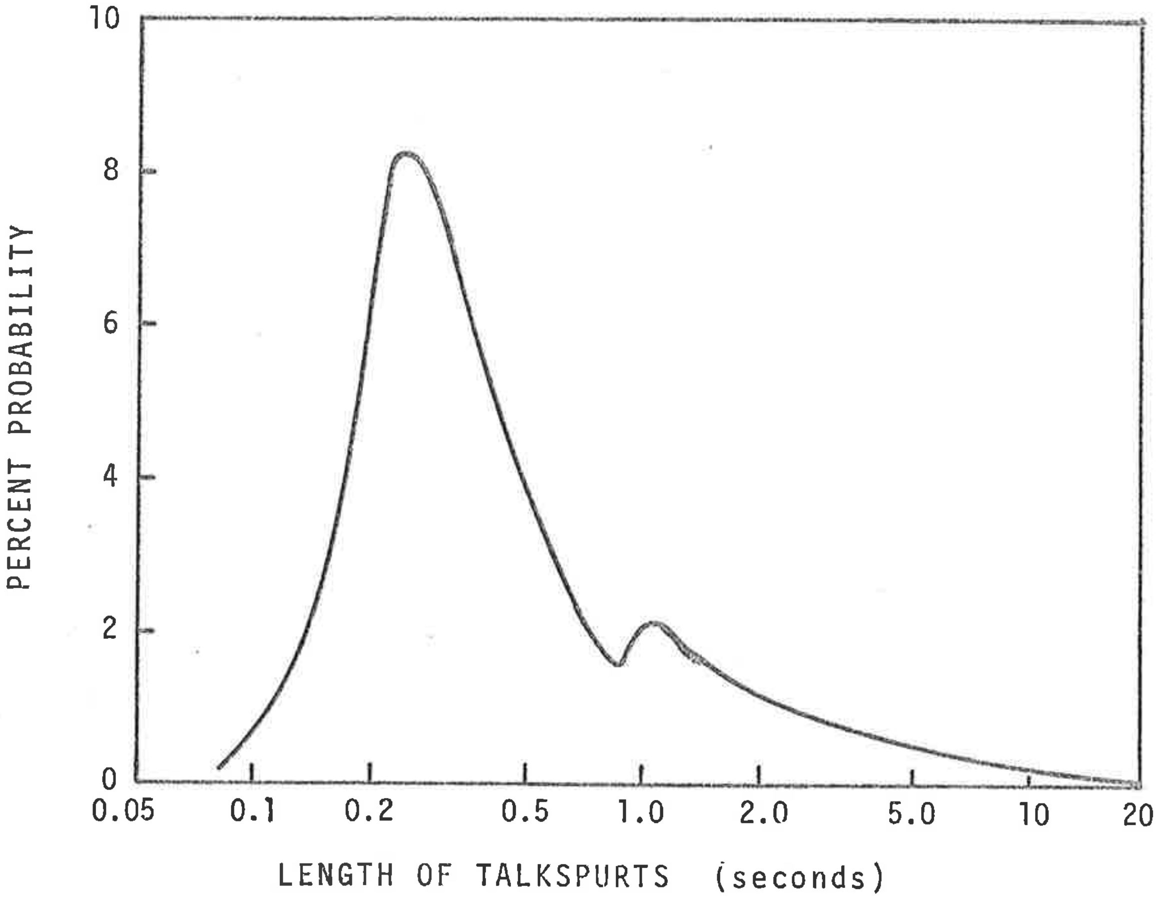


Figure 3.1 Probability density function of talkspurt length [77]

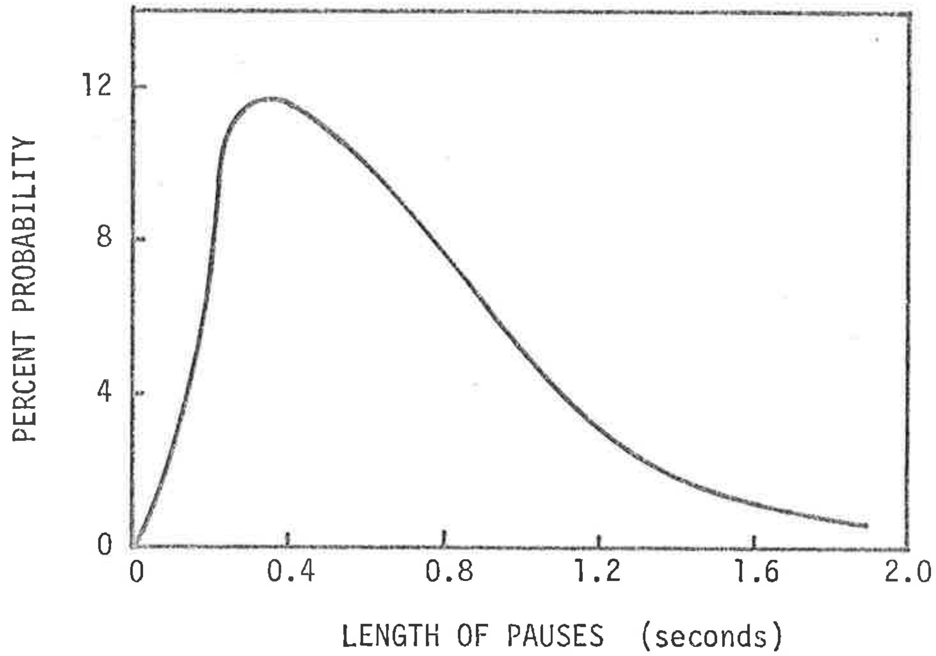


Figure 3.2 Probability density function of pause length [77]

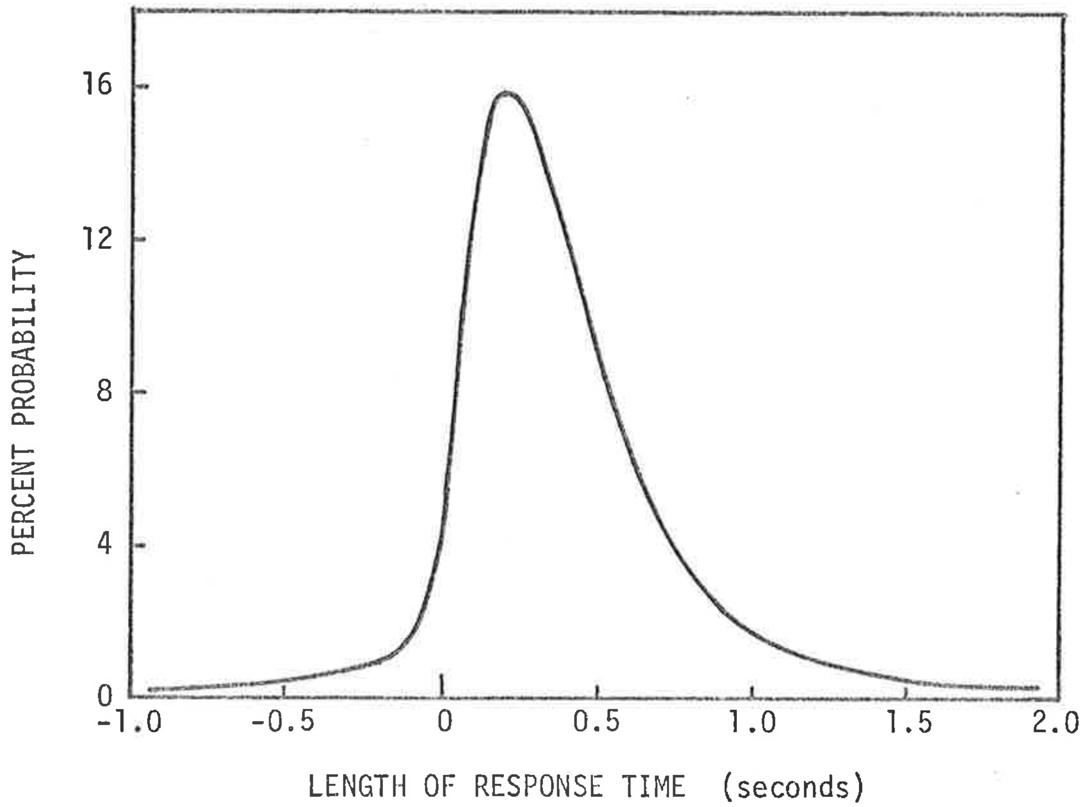


Figure 3.3 Probability density function of response time [77]

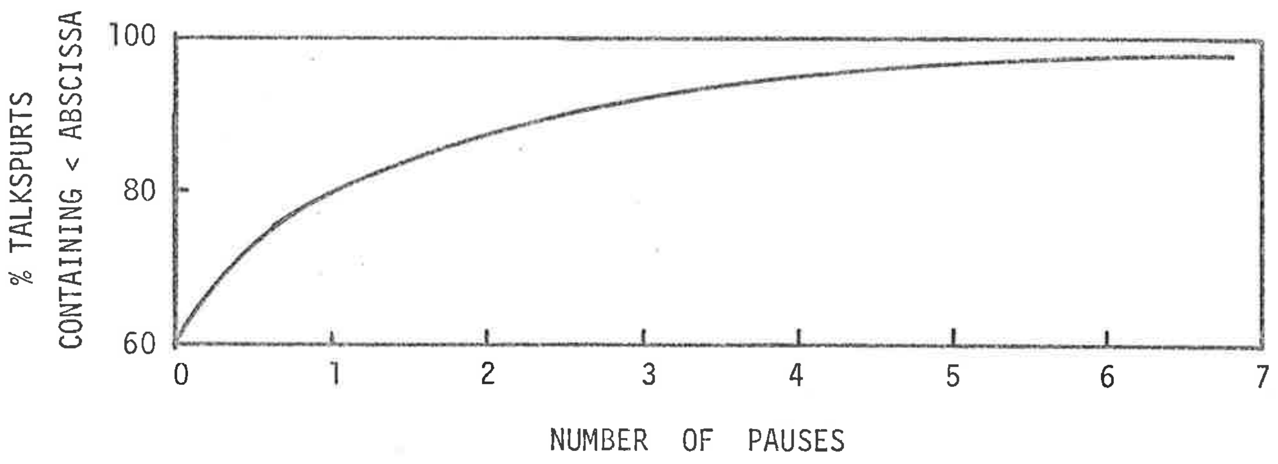


Figure 3.4 Distribution function of number of pauses in a talkspurt [77]

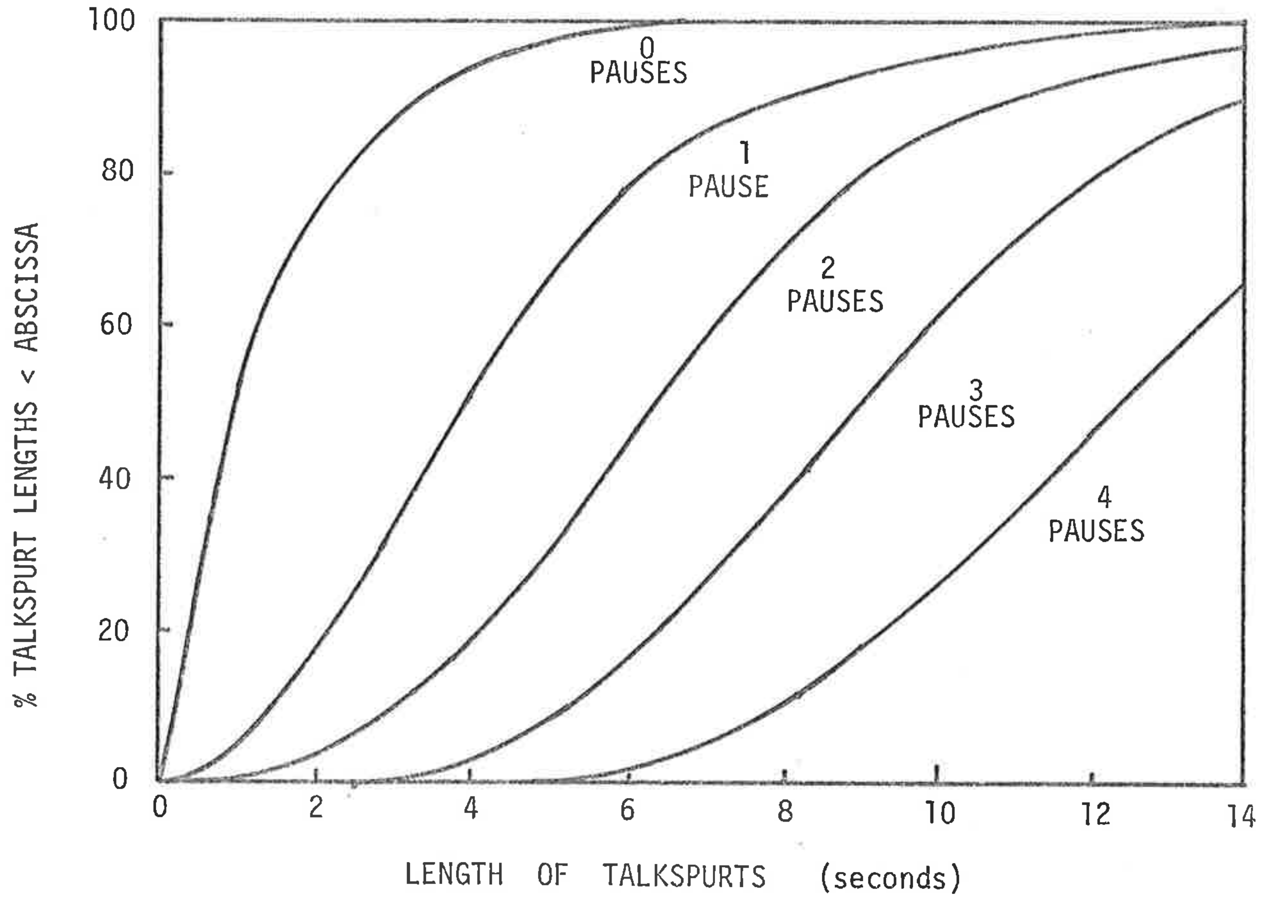


Figure 3.5 Distribution function of talkspurt length for variois numbers of contained pauses [77]

pseudo-random number generated by the computer is used in conjuction with the curves described, to determine a length for the variable in question. This results in quantities which are random but with the correct long term distribution.

Speech is formed in talkspurts as follows. Firstly the number of pauses is determined, then the length of a talkspurt with that number of pauses. The length of each pause is found and the total time left for speech is calculated by subtraction. This speech time is then randomly divided into the correct number of talk intervals.

A response time is determined and then another talkspurt and response time. The second talkspurt is not split into talks for it represents the speech from the far end party. This talkspurt is combined with the two response times to form a silence interval. The entire process is repeated at the end of this silence, thereby resulting in a realistic one way conversation of any desired length.

There are certain inaccuracies in this method of speech generation. For instance, the curves of figure 3.5 had to be extended to cover greater numbers of pauses per talkspurt. Also, curves of best fit were used for the tails of several distributions. The curves themselves were fed in as points on their respective distribution functions and intermediate points were determined by linear interpolation. There were also of course approximations and inaccuracies in determining points on the curve from the original reference. Finally, events such as negative total talk and silence times had to be corrected in the programme.

To ascertain the extent of the distortions caused by the above, the density functions of talkspurt, pause and response lengths were measured in long simulations (100,000 seconds of simulated speech). After a little tweaking of the curve data, very close agreement was obtained with the original curves as demonstrated by figure 3.6. The resultant silence interval distribution is shown for interest in figure 3.7.

As a final check on the simulation process, the probability density function (pdf) of talk length was determined. Although this information is not presented by Norwine and Murphy, many authors [37,78-80] have suggested that it is well modelled by an exponential distribution, i.e. the probability that a talk has length "t" is given by

$$P(t) = \frac{1}{a} e^{-t/a} \tag{3.1}$$

where "a" is the average talk length.

Figure 3.8 shows the talk length pdf obtained from simulation and a Poisson function with the same average length (1.79 seconds). Clearly there is remarkable agreement between the two. Significant discrepancies occur only at very short and very long talk lengths. In the former case, the simulation correctly reflects the fact that the probability of short talks, peaks at a length of 0.25 seconds. The close agreement at other lengths tends to confirm that both the Poisson process and the simulation are good approximations of real speech.

In the simulation programme speech is assigned to packets in the manner described earlier. This process is complicated by the possibility of a number of talkspurts being included within one packet. This is particularly likely if the packet length is long and the talkspurts and the silences between them are small.

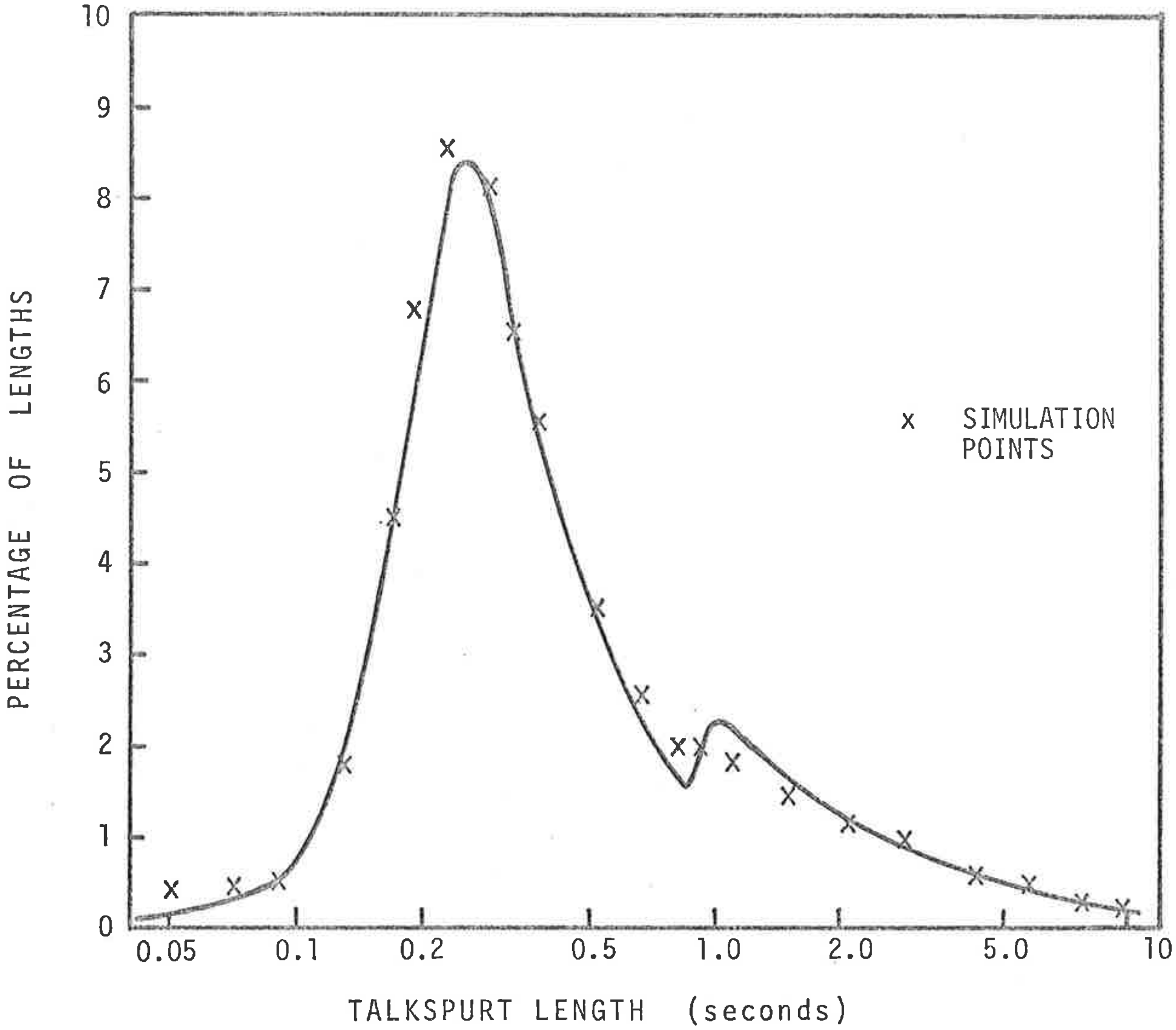


Figure 3.6 Confirmation by simulation of talkspurt length didtribution

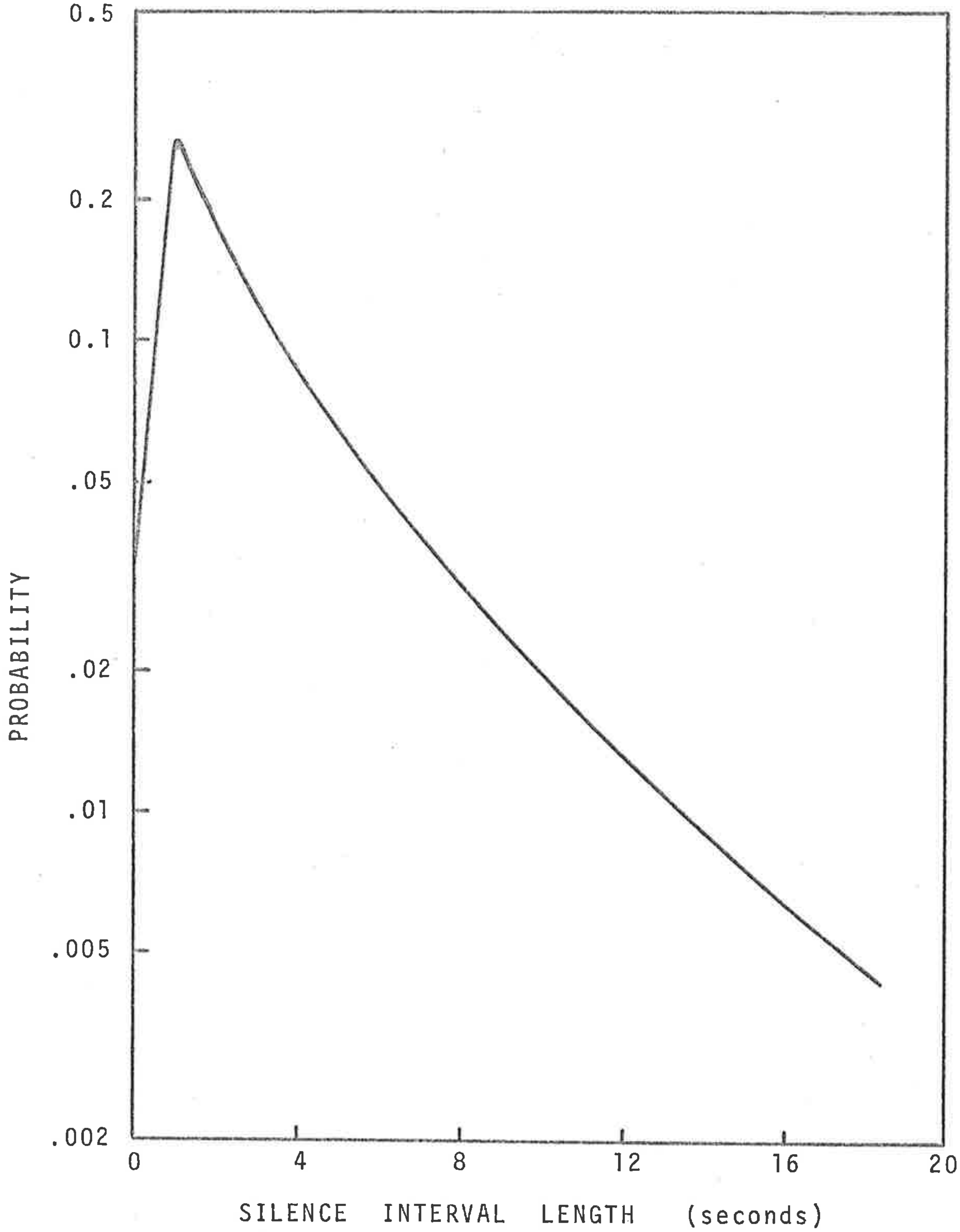


Figure 3.7 Probability distribution function of silence interval length

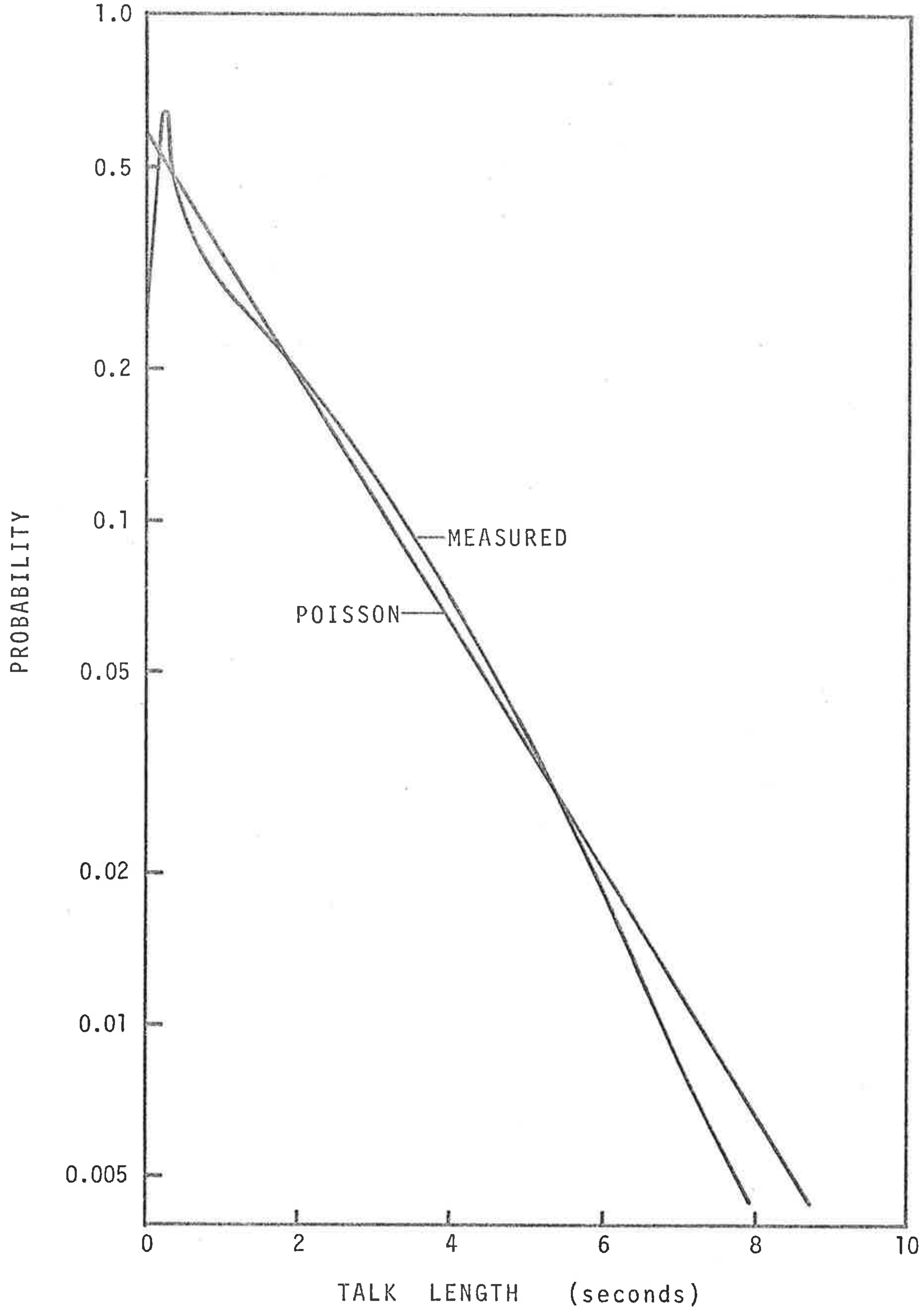


Figure 3.8 Probability density function of "talk" length

The information really desired from this simulation is the change in the interpolation gain, caused by transmitting speech in packets. This is determined by the amount of silence and pause time included within packets and depends upon the packet length. The interpolation gain, I, for any packet length, is the ratio of the total conversation time to the time included within the packets required by that conversation, i.e.

$$I = \frac{\text{Total Time of a Conversation}}{\text{Time of the Packets Conveying the Conversation}}$$

Figure 3.9 shows how the interpolation gain, as determined by simulation, varies with the packet length, T. Not surprisingly, the gain diminishes with increasing packet length since more silence is included in the packet.

For near zero length packets the total time contained in packets is virtually exactly the time for which speech occurs. Thus the interpolation gain at this point is merely the inverse of the average speech activity, which, in the programme used to determine figure 3.9, is 37.96%. This gives an interpolation gain at zero packet length of 2.634. At the other extreme, an infinite packet length corresponds to each speaker having a circuit dedicated to him alone. This is simply the TDM case for which the interpolation gain is unity. The curve is therefore asymptotic to the I = 1 line for large packet lengths.

An interesting feature of this curve is the relatively slow drop in gain with increasing packet length. For example, a gain of greater than 2.0 is found with a packet length of 1.2 seconds. This is somewhat surprising in view of the fact that the most common talk length is 0.25 seconds.

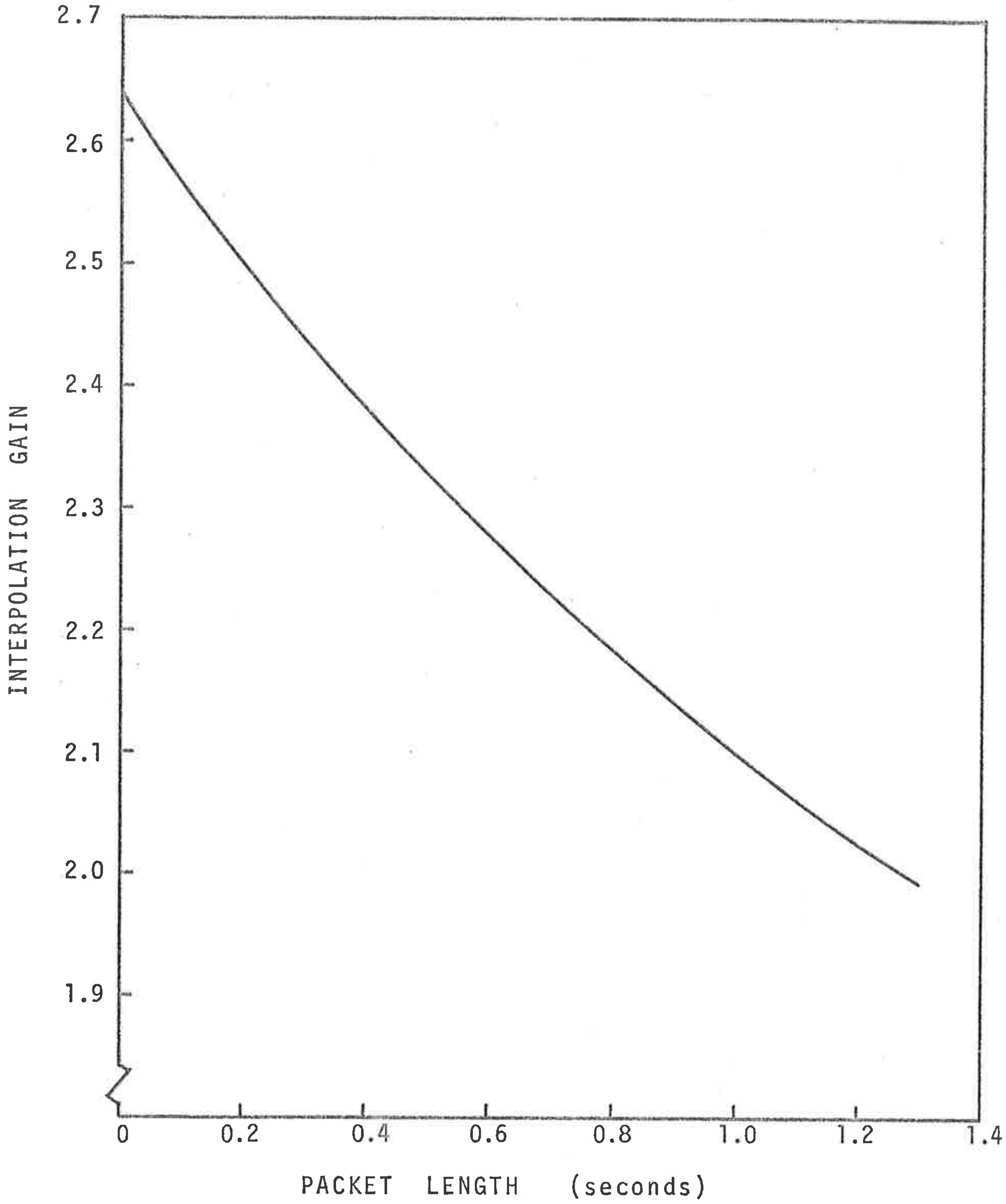


Figure 3.9 Interpolation gain of speech in packets

Another point requiring comment is the appropriateness of a speech activity around 38%. The actual percentage of the time that speech occurs is quite probably below this value. However, speech detection involves a compromise between excluding silence intervals containing noise, and including very low level speech. This usually results in a detection delay of up to 5 ms, to avoid unnecessary activity due to impulsive noise, and a hangover time of up to 100 ms. The hangover is a delay inserted after speech nominally stops and before the speech detector indicates this. It ensures that low level speech in the middle or at the end of a talk is not lost. Lyghounis [42] suggests that although speech activity can range down to 25%, a realistic speech detector produces an activity up to 40%.

In the case of packets no special hangover provision is required, since speech will virtually always end part of the way through a packet and the remainder is in fact hangover. Because of this, an activity factor of 38% is probably conservative here.

3.3 The Request Channel

In an interpolation system packets must be transmitted individually. Thus a request must be sent for each packet as described previously. Clearly, the request channel will require much greater capacity than in a TDM system, where only a few request packets are transmitted per call. The actual capacity required will now be determined.

The structure of a request packet can be identical to that in the TDM system. Call start up and end procedures are exactly the same and only one extra type of packet, a speech slot request, is required. The timing of requests is however more important in interpolation systems. Whereas a small delay arising from collisions make very little difference at the start and end of a call, such delays in the middle of a speech segment may cause disruptions. Therefore, it is critical that correct timing can be restored in the event of request packet collisions.

This need was recognised earlier when 10 bits were set aside for timing in TDM request packets. The actual starting time needs to be known to within an accuracy of 1 ms. A simple method of arranging this is to establish a time frame which repeats after a given number of speech slots and is divided into intervals of 1 ms length. Each telephone must keep a synchronised clock and use it to denote packet starting times by interval number. A 10 bit code provides for frame lengths as long as a second, which is sufficient to cover multiple collisions.

Frame synchronization can be maintained throughout the network by acknowledgement channel packets holding the number of the interval in which they are transmitted. This whole procedure is much simplified if the interval length bears a simple relationship to the packet transmission times.

If acknowledgement channel packets are required to carry the above timing information as well as a slot number for speech transmission, then an extra 10 bits will be required. However, it is certain that the central controller will need some time to process a request for slot assignment. Thus the station will reply to a request with an immediate acknowledgement packet and follow that sometime later with a slot assignment packet. If the acknowledgement packet carries frame synchronization bits and the assignment packet carries a slot number instead, adequate synchronization will be maintained without any extra bits.

The previous packet size range of 80 to 120 bits can therefore stand for all information packets in the interpolation scheme. An exact size can only be chosen after experimental transmissions in the urban environment, to determine exactly how many synchronization and perhaps error correction bits are required.

Consider again the general case in which W,D,B,A, and C are as defined in section 2.5 and in addition the packet length is denoted by T. The maximum rate of speech packet transmission (in packets per second) is

$$\frac{C}{T} = \frac{W}{D\,T} \tag{3.2}$$

This is also the maximum average rate of request transmission.

Thus recalling the ALOHA nature of the request channel, the capacity required to handle this traffic is

$$W_{r1} = \frac{A\,B\,W}{D\,T} \tag{3.3}$$

By comparison the capacity required by call set up and termination packets is (from (2.7))

$$W_{r2} = \frac{A\,B\,W}{20\,D} \tag{3.4}$$

Thus the total request channel capacity needed is

$$W_r = W_{r1} + W_{r2} = \frac{A\,B\,W}{D}\left(\frac{1}{T} + \frac{1}{20}\right) \tag{3.5}$$

Clearly that component of the capacity due to requests for slots, is much the larger for packet lengths under one second. With a smaller packet length the required capacity is increased as a greater number of packets must be transmitted per second.

Since two acknowledgement packets are required for each slot request, the total capacity of this channel must be (using (2.8))

$$W_a = \frac{2\ B\ W}{D\ T} + \frac{B\ W}{10\ D} = \frac{2\ B\ W}{D}\left(\frac{1}{T} + \frac{1}{20}\right) \qquad (3.6)$$

Again this is the smaller of the two information channels since A must be greater than 2.72.

Now consider the effect of a variation in the packet length. It was shown in section 3.2 that the length should be minimized to increase efficiency in packet formation. However this increases the capacity required by the information channels. Hence there must exist an optimum packet length which maximizes the speech transmitted in the total of speech and request channel capacities, W_t. This may be measured in terms of the full channel interpolation gain given by

$$I_f = I\,\frac{W}{W_t} = \frac{I\ W}{W + W_r} \qquad (3.7)$$

The request channel capacity is used here since it is the worst case.

Rearranging (3.7) gives

$$I_f = \frac{I\ D}{D + A\ B\ (1/T + 1/20)} \qquad (3.8)$$

This quantity possesses a maximum at a value of T dependent upon the system variables. Figure 3.10 shows the curve of I_f versus T for the following three examples.

(a) D = 30 kbit/s, A = 2.72, B = 80

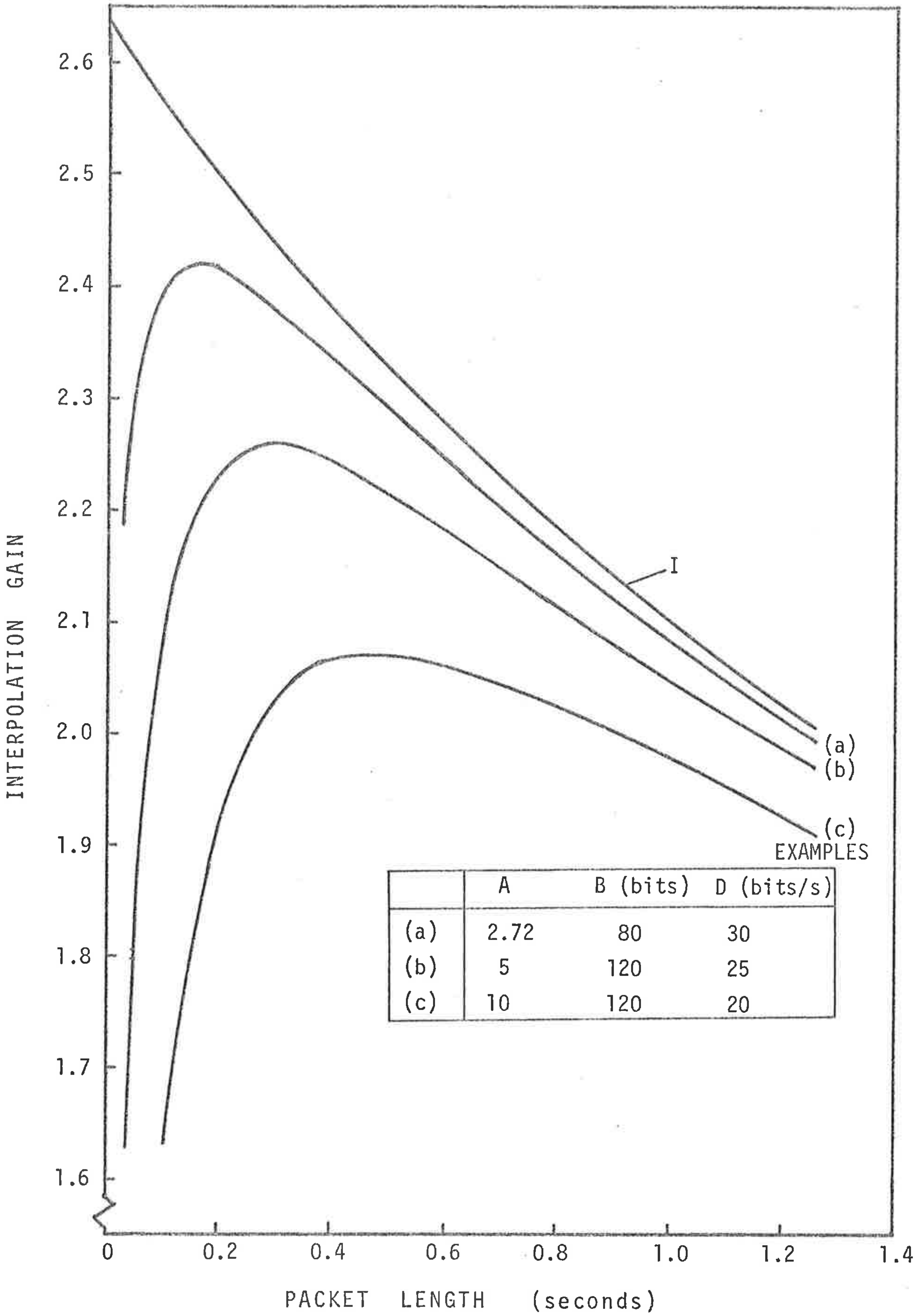
(b) D = 25 kbit/s, A = 5 , B = 120

(c) D = 20 kbit/s, A = 10 , B = 120

These cover the ranges of interest in mobile telephone systems. Example (b) is the standard case while (a) and (c) represent extremes. The optimum packet lengths are

(a) 0.17 s, (b) 0.32 s, (c) 0.47 s

This variable is shown for all feasible parameter values in figure 3.11. The optimum packet length increases with the request



	A	B (bits)	D (bits/s)
(a)	2.72	80	30
(b)	5	120	25
(c)	10	120	20

Figure 3.10 Examples of full channel interpolation gain

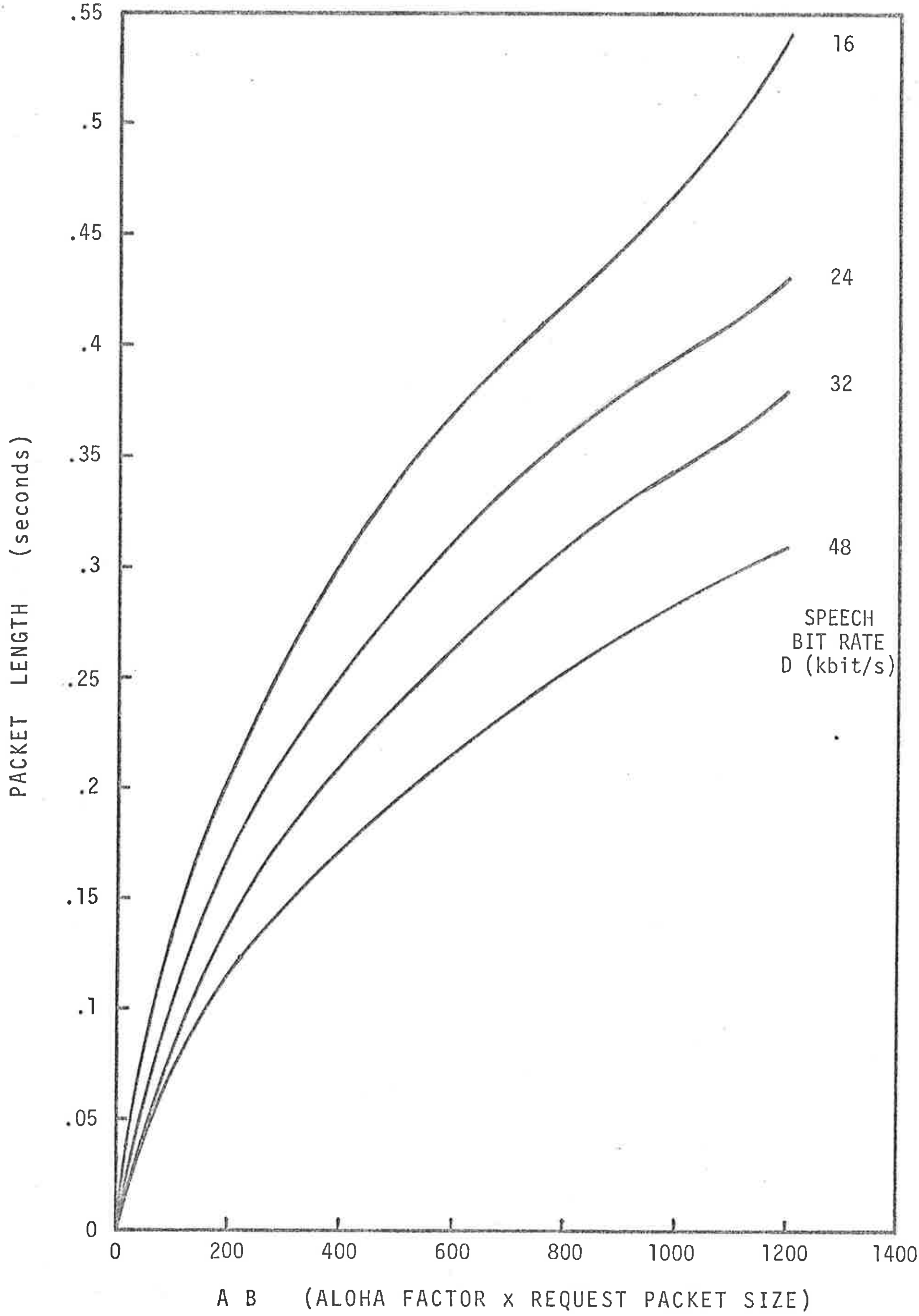


Figure 3.11 Optimum packet length for reservation ALOHA packet speech interpolation systems

channel capacity (A B) and decreases as the bit rate of digitized speech is increased. The first relationship follows from the fact that if more capacity is required for each request then the number of such requests should be reduced by increasing the packet length. As the bit rate of digitized speech is increased however, the relative effect of the request channel diminishes and a larger number of requests and therefore a smaller packet size, can be tolerated. The actual interpolation gain achievable is larger when the optimum packet length is smaller.

It should be noted that near its peak, the full channel interpolation gain curve is reasonably flat. Thus a packet length may be chosen which produces a gain close to the maximum for a wide range of system parameters. Since it is necessary to choose such a figure to proceed much further with the analysis, a value of 0.4 seconds will be used until more information allows a better choice.

With parameter values as in example (b) above, and with a packet length of 0.4 seconds, the full channel interpolation gain is 2.24. Now, with a 1 Mbit/s speech channel capacity, that required by the request channel is 61.2 kbit/s. If some larger but more convenient request channel capacity were chosen the effect would simply be to increase A. The acknowledgement channel can for simplicity have the same capacity. This channel will be considered in detail later.

The total one way capacity of the system is W_t = 1.0612 Mbit/s and the nominal number of speech circuits is, from (2.4), C = 40. Thus the theoretical number of simultaneous conversations possible is

$$N = \frac{W_t}{D} I_f = C\,I = 95.2 \tag{3.9}$$

Together with the gain achieved through interpolation however, there will be some reduction in the speech quality. The most obvious degradations are the delays inherent in the system. Various sources of delays and methods for minimizing their impact will now be considered.

3.4 Delays in the Digital Schemes

Both of the digital mobile telephone schemes considered above suffer from time delays in packet formation and transmission. The interpolation scheme incurs a further delay in the assignment of a transmission slot for each packet. Now delays do occur in the normal telephone network on long distance calls. However, delays in a digital mobile telephone scheme occur on every call and they can be quite large and even variable. The nature of these delays and their approximate values will now be defined.

A packet is transmitted at a much greater bit rate than that at which it is recorded. It therefore must be completely recorded before transmission can begin. Even if a packet is sent instantaneously immediately upon its completion, there is a delay, between the original speech and the reproduced version, equal to the packet length. This implies that the packet length should be kept as small as possible.

In the TDM system the choice of a packet size is governed largely by radio transmission considerations. A small packet is beneficial from a fading point of view. Also the main sources of error are impulsive. Therefore, smaller packets are less likely to contain multiple errors, and error correction is correspondingly easier. Another advantage of small packet size is that little memory is required at each telephone.

On the other hand, the slots for packet transmission require guard bands of fixed size to allow for variation in the propagation delays. Smaller packets therefore result in less efficient transmission. Also, a higher proportion of the bits must be dedicated to error correction in smaller packets. Thus, the final choice for a packet size must be a compromise based on these conflicting factors.

A reasonable figure for a TDM system is 1000 bits. With a 1 Mbit/s speech channel capacity the resulting slot size is 1 ms and, for a 25 kbit/s speech digitization bit rate, the packet length is 40 ms.

In the interpolation system however, the packet length is set at an optimum value to maximize the combined channel throughput. For the parameter values above, the optimum packet length is 0.32 s. If the length were to be reduced to 40 ms, figure 3.10 shows that the interpolation gain would drop to around 1.7, representing a 25% reduction in efficiency. Fortunately, the dual ideals of low delay and high efficiency can be achieved simultaneously by a different method, to be described shortly.

Other delays common to both digital systems arise during encoding, transmission and decoding. In the TDM system, packet recording is timed so that a packet is encoded and ready for transmission at the start of its slot. However, the transmission itself takes one slot length and at the station the packet must be checked and any errors corrected. From here the packet is transmitted over a land line to the central computer for switching to its ultimate destination. The bit rate on the land line will be the same as on the radio link since this enables cheap digital switching to be used at the controller.

Since advance knowledge is available of all speech packets in the system, the transmission delays can be reduced to the minimum incurred in receiving the packet, checking it and forwarding it to the next stage. Thus the total of these delays in the TDM case should be no more than a few speech slot lengths, or, under 10 ms for the example above.

It has been noted that one of the advantages of a packet system is the possibility of speech packet retransmission when the original packet is destroyed by fading or errors. In practice this operates as follows. If a station is unable to successfully decode a packet it requests the controller to allocate another transmission slot. If no other station has a good copy of the packet and if a slot is available, it is allocated and the station transmits the slot number to the appropriate telephone. The packet is then retransmitted in this slot and forwarded in the normal way.

Clearly there are delays involved in this process. Allowing for calculation time, at least 5 ms is required to notify the station of the new slot number. This must then be transmitted back to the telephone over the acknowledgement channel which has a slot length of 10 ms. Thus the average time taken for the telephone to receive the number is around 20 ms.

In addition, there is a delay until the designated slot arrives. Since in the TDM system, slots are dedicated to particular telephones in circuit form, a slot from the first appropriate free circuit must be used. In the worst case there will be only one such circuit with a slot occurring during the above 20 ms delay. Then the lost speech packet is delayed a complete packet length and is transmitted in place of the subsequent speech packet. This in turn uses the free slot in the next frame.

Thus to use this facility, each telephone must be able to store a packet for up to a full packet length after transmission. Also, the playback of packets at the far end has to be delayed by a packet length, so that a retransmitted packet can be inserted without a break.

If a packet from the station is destroyed, the mobile telephone must transmit a request for a repetition. A slot must then be assigned, just as above, and the consequent delay has the same maximum value. The cost of this retransmission facility is therefore a delay of one packet length. This brings the total one way delay for the TDM system to just under 100 ms. The acceptability of such a delay will be considered shortly. It must also be noted that retransmission is only possible if less than the maximum of C circuits are occupied.

In the interpolation system at least 5 ms is taken for the speech detector to operate. However, as a memory is available, it is simple to include speech during the detection time in a packet, and avoid initial clipping. Three request slot lengths are needed to formulate a request and transmit it to the central controller. This follows since speech begins at a random time in a slot and must be transmitted to the station and checked there before forwarding. In addition, there is some computation time required to assign the speech slot number and another two request slots to complete transmission of the slot number to the telephone.

With the example (b) system of section 3.3, the length of a request slot is 1.96 ms. Thus, the delay between speech beginning and a slot number arriving at the telephone is around 20 ms. If any collisions occur in the request channel there are further delays. The time taken for an acknowledgement packet to be received by a telephone after it has transmitted a request can be up to 5 slots (allowing for some queueing of acknowledgement packets at the station). This is referred to as the time-out number R (see section 2.1). If the maximum random delay associated with collisions is taken as K = 5 slots, then the average time for retransmission is 8 slots or 16 ms. This amount of delay is added every time a collision occurs.

Even with a very small packet length (for the interpolation scheme) of say 40 ms, one request collison will not prevent the slot number arriving at the telephone before packet recording is complete. With two collisons a small amount of extra delay does occur at this packet length. However, since the probability of two or more collisions under these circumstances is only 0.09, having to request a slot for each packet rarely results in any delay.

Again if a speech packet is destroyed during transmission it may be repeated. The procedure is exactly as in the TDM system except that the delay involved is smaller. This comes about because the acknowledgement slot length is smaller and hence the new slot number can be returned more quickly (in about 10 ms). Also since speech slot assignment is dynamic, there are no preassigned slots and the repeating packets can be given priority. Thus it is possible to retransmit the packet within 20 ms.

A potentially much more serious delay occurs if, due to several requests arriving in quick succession, some have to wait many slots for an assignment. This delay depends in a quite complex way upon the instantanious rate of request arrivals. It has significant consequencies and will be considered in detail in chapter 4. At this stage it is sufficient to state that the delay is random and may be large.

To summarise, the one way delay in the TDM scheme with retransmission allowed will be of the order of 100 ms. In the interpolation system, delay from transmission through the network and from retransmission can be kept below 30 ms in total, regardless of the packet length. However, packet recording and slot assignment can be far more serious sources of delay. The first of these will be considered now.

3.5 Subpackets

The use of an optimum packet length in the interpolation scheme maximizes the system efficiency. However, at the same time it introduces a large recording delay into the speech path. To reduce this delay and still maintain the optimum packet length, the packet must be split into a number of parts for transmission. If for instance a packet is divided into M subpackets, each of which is transmitted as it is readied, then the recording delay is reduced from T to $\frac{T}{M}$ seconds. In this case each request reserves M slots in the speech channel instead of one, but covers the same total amount of speech. Thus the optimum packet length calculation is not affected.

As an example, consider the interpolation system described above with M = 10 subpackets per packet length. The recording delay is reduced to 40 ms, just as in the TDM example, and there are 1000 bits in each subpacket. Transmission slots, now containing subpackets, have a length of 1 ms and the subpacket format is as illustrated in figure 3.12.

A problem arises with the use of subpackets if a transmission slot number has to be sent to the mobile telephone for each subpacket Since only one slot number can be sent per acknowledgement channel packet, there must be M such packets, plus the initial acknowledgement, for each request.

In addition, speech from the station to the mobile telephone has to be sent in subpackets. The destination address of each of these return subpackets can be included within the subpacket itself but this creates some difficulties. Firstly, as well as the address bits, special error correction is required because of the importance of the address information and the relatively high channel error rate. These extra bits will of course reduce the packet efficiency. Also it is now necessary for the telephones to monitor the speech channel in addition to the acknowledgement channel. To avoid these problems, packets must be sent on the acknowledgement channel to inform the telephone when to receive speech subpackets.

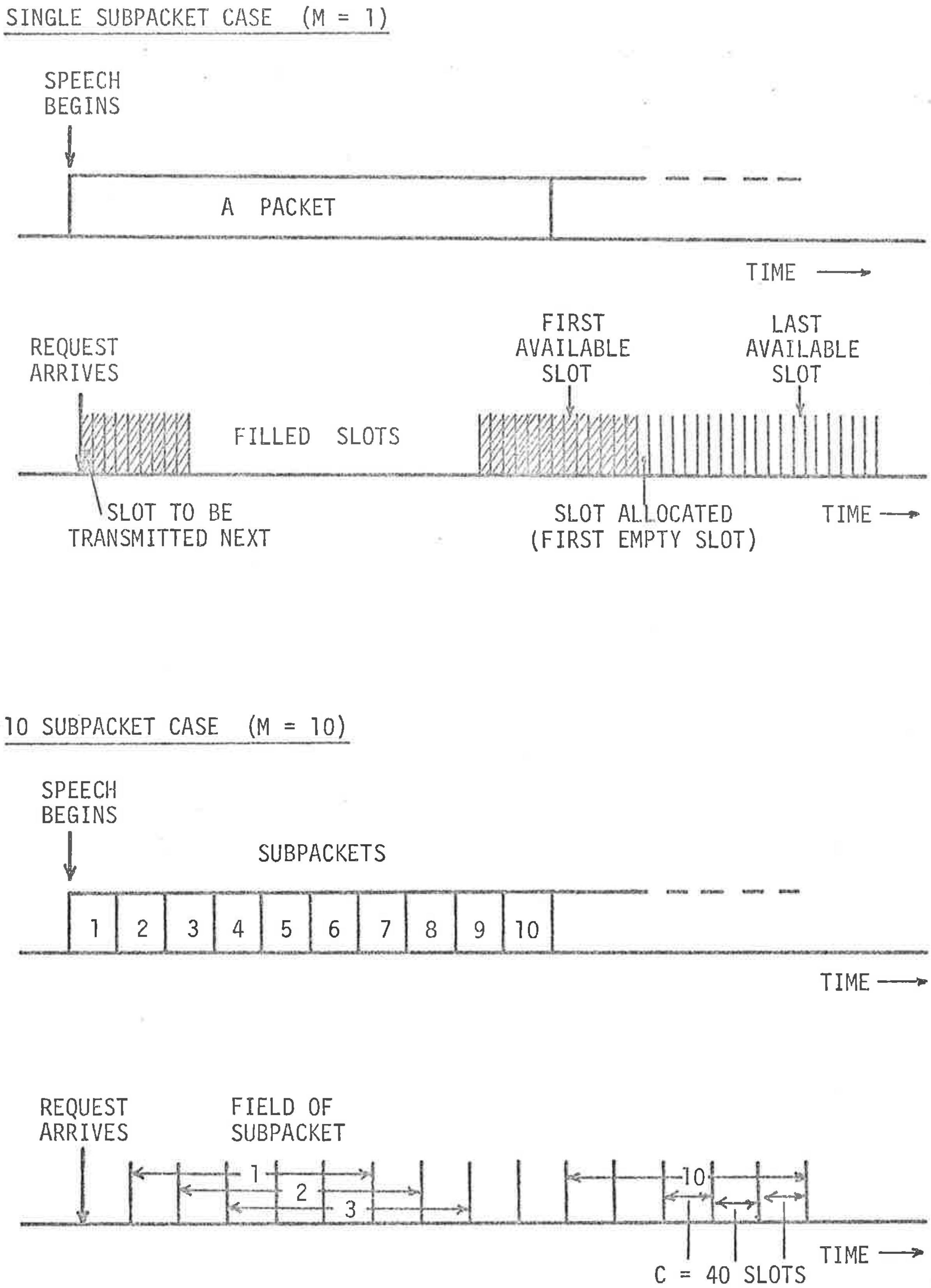


Figure 3.12 Illustration of packet transmission procedure

If a slot number is transmitted for each subpacket, M transmissions are required for each packet destined for a mobile telephone. Thus the total capacity required by the acknowledgement channel becomes (from (3.6))

$$W_a = \frac{(2M + 1)\ B\ W}{D\ T} + \frac{B\ W}{10\ D}$$

$$= \frac{B\ W}{D} \left(\frac{2M + 1}{T} + \frac{1}{10} \right) \tag{3.10}$$

If an ALOHA factor of 5 is used in the request channel, then, where M is greater than 2, more capacity is required by the acknowledgement channel than by the request channel. This is important for the optimum packet length calculation.

Various methods however, are available to reduce the required acknowledgement channel capacity. Firstly, subpacket transmission can be made partially or wholly periodic. For instance, separate slot assignments may be provided for only alternate subpackets. Then the subpacket following one with an assigned slot is transmitted exactly one subpacket length later. In the extreme case only one assignment is provided per packet. The slot number is for the first subpacket and others are transmitted sequentially at subpacket length intervals.

Another possible technique is to amalgamate the request and acknowledgement channels so that a single channel is used by both stations and telephones. To allow high occupancy, the slot length has to be increased so that the station transmits in a slot only after it detects no telephone requests in that slot. This has the dual advantage of doubling the ALOHA factor of the request channel and adding the almost 80% of that channel's capacity, which is unused, to the acknowledgement channel. The disadvantage is that the new channesl's capacity exceeds 100 kbit/s.

Thus using the above techniques, the acknowledgement channel capacity can be kept effectively equal to, or below, that of the request channel . This justifies the use of the request channel to determine the optimum packet length when subpackets are employed.

Subpackets therefore enable the recording delay to be reduced to a reasonable value, regardless of the packet length. However, there are still significant delays in both of the digital schemes considered. Thus, it is necessary to determine the subjective effects of such delays in telephone conversations. This should indicate an acceptable level for the delay and the digital systems can be designed accordingly.

3.6 Subjective Effects of Telephone Delays

Delays of up to 100 ms may be experienced in long distance telephone calls in a land based network. If a single or multiple hop satellite link forms part of the transmission path, the delay will generally be greater than this. A geostationary satellite introduces a delay of almost 300 ms. If both forward and reverse signals travel via such a satellite, the round trip delay is 600 ms, and if a two satellite hop is used it becomes 1200 ms. Concern as to the effect of such delays on speech quality has prompted much research. The conclusions drawn from this are described here.

Peoples response to delay changes significantly if an echo is present. The response without echo will be considered first. Klemmer [81] and others have performed subjective tests on the effects of pure delay. They found essentially that a delay of 600 ms did not cause any discomfort to the speakers, and no loss in quality was reported. Similarly, there was very little objection to delays of 1200 ms. When even longer delays of 1800 and 2400 ms were inserted, the test subjects noticed some degradation and were thereafter slightly more conscious of 1200 ms delays.

This suggests that the response to delay is far more complex than to say increased noise on the line. This was confirmed by Brady [82] who measured various characteristics of conversations with delays of 0, 600 and 1200 ms. He found that the main effect caused by delay, was a greater number of confusions between the speakers. Other effects, were an increased incidence of both people talking and both silent and a tendency for people to remain silent longer and to pay less attention to interruptions. These changes, while being statistically significant on average were relatively small so that the essential aspects of the speech were altered very little.

The subjects of the experiments were unable to identify the cause of the confusion as delay in the circuit. In addition, they apparently automatically adjusted their speaking technique to minimize the effects. It was also found that increasing the delay from 600 ms to 1200 ms produced no significant extra degradation. Thus it appears that a delay of 600 ms is sufficient to cause all of the above changes to reach their asymptotic values.

If an echo path exists, people are far less tolerant of delays [83]. Even a delay of 20 ms can result in perceived severe degradation unless the echo amplitude is sufficiently small. To achieve reasonable quality with a delay of 90 ms, the total loss in the echo path must exceed 40 dB. For a delay of 600 ms, a 60 dB loss is required [84]. Such losses are achieved in practice through the use of echo suppressors. These insert attenuation into the transmission path from a telephone when speech is detected on the incoming line. Unfortunately, echo suppressors add their own type of distortion to speech, since they tend to chop it on and off and can result in large amounts of speech being lost when both parties are talking.

Echo suppressors are generally used whenever the delay in a telephone line exceeds about 40 ms [85]. Special suppressors are used for long delay circuits. The problems that exist even with (and because of) these devices, prompted the CCITT to issue a recommendation regarding circuits having delay [86]. This states basically that round trip delays less than 300 ms are always acceptable. Round trip delays between 300 and 800 ms are acceptable, provided care is taken to minimize echo creation and, for delays exceeding 600 ms, echo suppressors designed for long delays are used. Round trip delays of greater than 800 ms are unacceptable, except in the most unusual circumstances.

In the normal telephone network, echos arise whenever there is a change in characteristic impedance in the transmission path. The most common echo source is the hybrid network, which enables both forward and reverse signals to travel on two wires. At the mobile telephone the two paths are kept entirely separate and an echo can occur only by acoustic feedback through the microphone. This path is a very high loss one and the echo amplitude is at least 40 dB below the local speakers level. Thus in the TDM scheme, when the echo reaches the original source, it is at least 50 dB below speech level and may well cause no problems.

If this loss is insufficient an echo suppressor or an echo cancellor is required. The latter removes any echo from a transmitted signal by subtracting an appropriately delayed version of the received signal. This is reasonably straightforward in a digital system, especially one with memory already present [87]. Such a device may be required at each mobile telephone in the TDM system. Echo suppressors or cancellors are certainly required at the interface with the normal telephone network, since this network undoubtably generates echos. One advantage of a digital echo cancellor in this position is that it can be time shared among a number of output lines.

In the interpolation system no echo can be transmitted unless packets are being generated. Of course it is imperative that echos do not activate the speech detector. If at the mobile telephone they are sufficiently large to do this then a more complex speech detector must be used. This can, for instance, subtract an appropriate version of the received signal from the microphone output, prior to determining whether speech is present. Thus an echo occurs here only when both parties are talking and is then returned to the original source at least 50 dB down. Echo suppressors as such are therefore not required at the mobile telephones in an interpolation system. However again, an echo suppressor or cancellor is necessary at the network interface.

It appears that a one way delay of 0.3 seconds is not unreasonable in a normal telephone conversation. Though this requires an echo removal device, a digital inplementation results in much smaller effects on the signal quality than the analogue equivalent. Hence the criterion of a maximum one way delay of 0.3 seconds will be adopted for the schemes considered here.

The TDM system has a one way delay of under 0.1 seconds, which is quite reasonable and permits long distance, and possibly even satellite calls, to be made from or to mobile telephones.

The one way delay in interpolation systems depends upon the time spent waiting for a free slot. The remaining delays, due to packet formation and transmission, can be kept to around 50 ms when an appropriate number of subpackets per packet length are chosen. If the final system is to meet the above criterion, the delay in slot assignment must be limited to around 0.25 seconds. The feasibility of this is determined in the next chapter.

3.7 Conclusions

In this chapter, the use of packet speech interpolation to increase the number of simultaneous conversations in a mobile telephone system has been considered. The possible gain is limited by the efficiency with which speech can be placed into packets. This in turn depends upon the actual structure of a conversation. A computer simulation of telephone conversation has been used to determine the efficiency, which is presented in terms of the interpolation gain in figure 3.9. This is a general result, not previously reported. It demonstrates that the interpolation of speech in packets is a worthwhile objective, since it can more than double the number of simultaneous conversations in a digital system.

Interpolation requires at least one request packet to be transmitted for each speech packet prepared. This necessitates a request channel capacity which is significantly greater than in the TDM system and which increases for smaller packet lengths. Since the interpolation gain also increases for smaller packet lengths, it is apparent that there must exist an optimum length that maximises the number of simultaneous conversations per unit total capacity.

This was found by calculating the full channel interpolation gain I_f, which takes into account the request channel capacity. Examples showing how I_f varies with packet length, and the optimum lengths for a wide range of system parameters, are shown in figures 3.10 and 3.11 respectively. From this analysis, a packet length of 0.4 seconds was chosen for the example interpolation system, with other parameters as follows

W = 1 Mbit/s , B = 120 bits , D = 25 kbit/s , A = 5 , W_r = 61.2 kbit/s.

A theoretical maximum of 95 simultaneous conversations can be handled by this system.

The most obvious problem with these digital schemes is the delay introduced into the speech path. The three main sources of delay are packet formation, transmission through the network and packet retransmission. These produce a total delay in the example TDM system near 100 ms. In the interpolation system the delay also includes a waiting period for a free transmission slot. Since in this system the chosen packet length must be reasonably large to produce a high gain, sub-packets are required to reduce the recording delay to acceptable levels. Delays in transmission and retransmission are small and result in a total one way delay, excluding slot assignment, of the order of 50 ms.

When the subjective effects of delay in telephone conversations are considered, it is found that pure delays of 600 ms or more are permissible and result mainly in minor changes in the speakers conversation techniques. With an echo path however, even small delays are highly disruptive. In this case echo suppressors or cancellors become mandatory.

From this investigation it may be concluded that the delay in the TDM scheme is acceptable and that that in the interpolation scheme should be kept below 0.3 seconds. The final figure for this system depends upon the delay in slot assignment which will now be considered in detail.

4. GLITCHES IN PACKET SPEECH INTERPOLATION

4.1 Slot Assignment Delay

There exists for every speech packet an optimum transmission slot in the speech channel. This is the first slot beginning after the packet formation is completed. If this slot is allocated to the packet, then the only delay involved is the transmission time itself. However, if this slot has already been assigned to some other telephone, then the packet must await a free slot. Such a situation will arise often because the arrival of requests at the central controller is essentially random. There will be periods when no requests arrive and some slots in the speech channel are left empty. On the other hand, when a number of requests arrive within a short period they will have to queue for a slot.

The delay that any request experiences equals the number of requests in the queue upon its arrival, multiplied by the slot length. This delay will be random since it depends upon the past history of request arrivals. Very large delays are possible, and in fact if the average number of arrivals is one per slot, the queue length will tend to infinity. Even for rates of input smaller than this, the queue length is certain to become very large at some time.

Consider the effect that such a random and possibly large delay has on the reproduced speech. There can be no guarantee that consecutive packets from any one telephone will be delayed by the same amount. If the later packets are less delayed then they are simply added, in the memory, to the end of the packet being replayed. However, if later packets are more delayed there will be a break in the reproduced speech. Alternatively, if there is some separation between the two packets due to a break in the original speech, then a smaller delay on the later packet will reduce the silence interval and a longer delay will increase it.

The only way to avoid this disruption to the speech is to incorporate a fixed delay between the original speech and its reproduction. Then if a packet arrives with less than this delay it will wait out the remaining time at the receiver. It was decided in the previous chapter that the delay in the interpolation scheme must be kept below 0.3 seconds in total. There must therefore be a somewhat smaller limit on the delay allowed in slot assignment. The effect of such a limit is, by the reasoning above, to set a maximum length for the slot assignment queue.

If a new request arrives to find the maximum number already waiting, then it has to be delayed by one slot length more than the time allowed. Such a request must be abandoned. The packet for which the request was sent cannot be transmitted and the speech it contains is lost. This results in a break in the reproduced speech which is termed a glitch.

Such glitches occur in one form or another in all interpolation schemes since they result from temporary channel overload. This was first examined in detail for the TASI transatlantic cable system. Here an overload results in no circuit being available to a speaker at the start of a talk. His speech therefore is clipped or "frozen out" until a circuit becomes free. In certain digital interpolation schemes considered in chapter 8, glitches take the form of a temporary reduction in the signal to noise ratio.

The subjective effect of glitches was first considered for TASI type freezeouts on the transatlantic cable system [37-39]. It was found that the effect on transmission quality was negligible, providing the fraction of speech lost was less than 0.5%. A loss of 2% was considered somewhat objectionable. Other studies [44,88] suggest that the length of the glitches is important subjectively and that attempts should be made to avoid long glitches.

Glitches in a digital packet system however, differ considerably from those in the TASI scheme. Speech may be lost from any part of a talk instead of just the beginning and the glitch length distribution is also changed. As a first step in the study of glitches the probability of their occurrance will be determined. This is essentially a queueing problem of overflow from a finite sized buffer and it may be tackled theoretically.

4.2 Theoretical Glitch Probability

Consider initially the single subpacket case in which each request is satisfied by one slot. Requests derived from a particular source are assumed to be uncorrelated and the sources to be independent. This results in a random input which may be modelled as a Poisson process. Thus the probability of "k" requests arriving in any one slot is

$$P(k) = \frac{U^k}{k!} e^{-U} \tag{4.1}$$

where "U" is the usage; the average number of requests arriving per transmission slot.

Now the number of requests in the queue is termed its state. This will vary from zero when the queue is empty, up to a maximum number "L", corresponding to the maximum delay. At each new slot, one request in the queue will be satisfied and k new requests will arrive with a probability given by (4.1). Thus the state may decrease by one, remain as it is, or increase toward L.

This is an example of an M/D/1 queue of finite capacity. Here the "M" refers to a Markov or Poisson input distribution, the "D" to a deterministic or constant service time distribution and the "1" to a single server. Such queues have been well studied for the infinite capacity case [89-91] which is the asymptotic form of a finite queue of large limit. The average number of entries in such an infinite queue is

$$E = \frac{U\,(2 - U)}{2\,(1 - U)} \qquad 0 \leqslant U \leqslant 1 \qquad (4.2)$$

Table 4.1 shows the values of E for various usages.

U	.8	.9	.95	.99
E	2.40	4.95	9.98	50.00

Table 4.1 Expected number of entries in an infinite M/D/1 queue.

This provides some insight into the queue length required by a finite queue, but more useful information is provided by the state probabilities. These are the long term probabilities of being in any particular state, i.e. that the queue is of a particular length. Such probabilities are easily converted into the appropriate values for a finite capacity queue. Unfortunately, no analytic solutions for the state probabilities exist.

The finite capacity queue has also been studied theoretically and again no analytic solutions have been found. Various authors [52,92] have been able to establish theoretical limits for the state probabilities but these are accurate only for large L. The sole method of obtaining the exact state probabilities is an iterative procedure based upon Markov chain theory.

Since each state can communicate with (change to over a period of one or more slots) every other state, and since there are a finite number of states, L + 1, the system forms a positive recurrent Markov chain. Thus a long time after initiation, the probability of being in a particular state "i" reaches a steady state value, denoted " Π_i". These probabilities can be found by solving the L + 1 simultaneous equations

$$\Pi_i = \sum_{j=0}^{L} \Pi_j \; Q(j,i) \qquad 0 \leqslant i \leqslant L-1$$

$$\sum_{i=0}^{L} \Pi_i = 1 \tag{4.3}$$

Here "Q" is the transition matrix, i.e. Q (i,j) is the probability of the state changing from "i" to "j" in one slot period. Some disagreement exists in the literature about certain values of Q (i,j). This arises from the choice of a particular instant in a slot when a transmission is said to occur. If transmission is assumed to take place instantaneously at the end of a slot, then a request may be transmitted in the slot in which it arrives. On the other hand, if transmission occurs at the start of a slot, a request arriving during the slot cannot be transmitted until a subsequent slot.

These two cases are examined in appendix A, where it is shown that the state probabilities generated by the two approaches are identical if a limit of L is used in the first case, and of L+1 in the second. It is also shown that the first case is appropriate for use here.

The transition matrix is therefore defined as follows.

$$Q\ (0,0) = \text{Probability [0 or 1 requests arrive]}$$

$$= P\ (0) + P\ (1)$$

$$Q\ (0,j) = \text{Probability [(j + 1) requests arrive]}$$

$$= P\ (j + 1) \qquad 1 \leqslant j \leqslant (L-1)$$

$$Q\ (i,j) = \text{Probability [(j - i + 1) requests arrive]}$$

$$= P\ (j - i + 1) \qquad 1 \leqslant i \leqslant L,\ (i-1) \leqslant j \leqslant (L-1)$$

$$Q\ (i,L) = \text{Probability [at least (L - i + 1) requests arrive]}$$

$$= 1 - \sum_{j=0}^{L-i} P\ (j) \qquad 0 \leqslant i \leqslant L$$

All other $Q\ (i,j) = 0$.

If these values are substituted into (4.3), an iterative technique may be used to obtain the Π_i and hence the glitch probability. The probability of an input to the queue is simply U, and the probability of an output from the queue is

$$1 - \text{Probability [a slot is left empty]} = 1 - \Pi_0\ P\ (0).$$

The glitch probability is the difference between these two quantities.

$$GP = U - 1 + \Pi_0\ P\ (0) \qquad (4.4)$$

A more useful quantity is the glitch rate, GR, the fraction of packets glitched.

$$GR = 1 - \frac{1 - \Pi_0\ P\ (0)}{U} \qquad (4.5)$$

This quantity was determined as a function of U for queue length limits of L = 20, 30 and 40, and is shown in figure 4.1. These curves cover the useful region of operation of practical mobile telephone systems.

Input in the system just analysed is provided by a Poisson process. However, this is an approximation to the actual case since it assumes

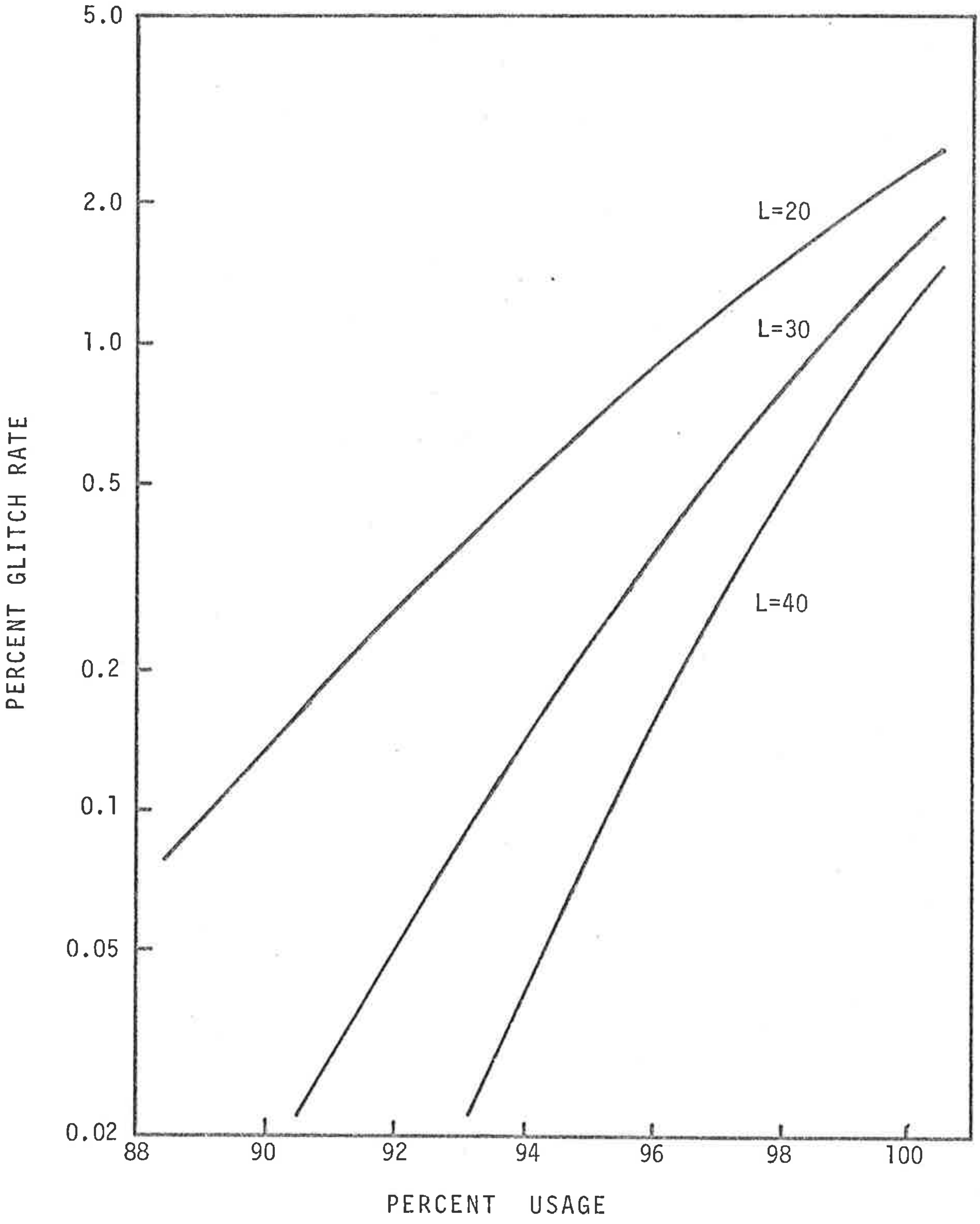


Figure 4.1 Theoretical glitch rate for various queue length limits, L

an infinite number of sources. It is more correct to use a Binomial input distribution. Then the probability of having k inputs in any one slot is

$$b\ (k\ ,\ N\ ,\ q) \quad = \quad {}_{N}C_{k}\ \ q^{k}\ \ (1 - q)^{N-k} \tag{4.6}$$

where "N" is the number of sources and "q" is the average activity, related to the usage by

$$q \quad = \quad \frac{U}{N} \tag{4.7}$$

The glitch rate was determined with this input for various values of N and the results are summarised in table 4.2 for L = 20 and U = 1.0.

Number of sources N	30	60	120
Binomial glitch rate	2.268	2.306	2.325
% Error with Poisson	3.24	1.63	0.82

Note L = 20 ; U = 1.0 ; Poisson glitch rate = 2.344

Table 4.2 Comparison of glitch rates

Clearly the Poisson input provides an upper limit on the glitch rate, but there is very little difference between the two distributions for systems of realistic size. The advantage of having one less variable in the Poisson case more than makes up for its slight loss in accuracy. Also, it should be realized that both distributions are in any case approximations to the real situation. The Poisson input will therefore continue to be used to derive theoretical quantities in mobile telephone systems.

4.3 Glitch Rate and the Use of Subpackets

It is now possible to calculate the theoretical glitch rate for any given maximum slot assignment delay. Consider again a system with C voice circuits and a packet length of T seconds. Then the length of a speech packet transmission slot in seconds is

$$Z = \frac{T}{C} \tag{4.8}$$

If the delay allowed in slot assignment is "V" seconds then the number of slots within this limit is

$$L = \frac{V}{Z} = \frac{C\,V}{T} \tag{4.9}$$

For example system with C = 40 and T = 0.4 seconds, an appropriate value for V might be 0.2 seconds. This provides L = 20 slots within which a request must be satisfied to avoid being glitched. Then from figure 4.1, if a glitch rate of say 0.5% is desired, the system must theoeretically be operated at a usage of 94%. This reduces the allowable number of simultaneous calls from 95.2 to 89.5

Now a delay of 0.2 seconds in slot assignment will only be acceptable if all other delays are kept below 0.1 seconds. Thus subpackets will be necessary to reduce the recording delay. The effects of subpackets on the theoretical glitch rate must therefore by determined.

If each packet is split into M subpackets, the slot length will be divided by M and become

$$Z' = \frac{Z}{M} \tag{4.10}$$

This results in a corresponding increase in the number of slots within the slot assignment limit to

$$L' = L\,M \tag{4.11}$$

Figure 4.1 shows how a change in the size of L by a factor of two, results in much more significant changes in the glitch rate. Hence, if the effect of using subpackets is simply to increase L as suggested by (4.11), then the glitch rate will be rendered negligible at virtually all usages. Unfortunately, the situation is not this simple because the input to the queue is no longer independent. To understand this consider the manner in which subpackets enter the queue.

If a packet begins at time slot zero, then the first subpacket is available for transmission by time slot C. This subpacket is not glitched as long as it is transmitted by slot $C + L'$. The second subpacket must be transmitted between slots 2 C and $2C + L'$ and so on. Thus, each subpacket has a field of possible transmission slots L' wide and it is quite possible for the fields of adjacent subpackets to overlap. This is easily seen from figure 3.12, showing the example case above with M = 10. Here C is 40 slots and L' is 200 slots. Clearly there is considerable field overlap. In fact the 240 slots between numbers 200 and 440 can each hold any of five different subpackets.

The best way of visualizing this queue system is shown in figure 4.2. When a request arrives, the first subpacket is assigned to the next free slot in the first queue, and the remaining subpackets are added to the end of their respective queues. These latter queues are simply delaying devices which ensure that any input becomes available at the output C slots later. At each time slot every queue is shifted down by one slot, with the output of the first queue being transmitted and the outputs of other queues being fed into the queue above.

This arrangement is necessary to ensure that the priority of the various subpackets remain correct in the final queue. The priority may be defined as the first slot number in which the subpacket can be transmitted.

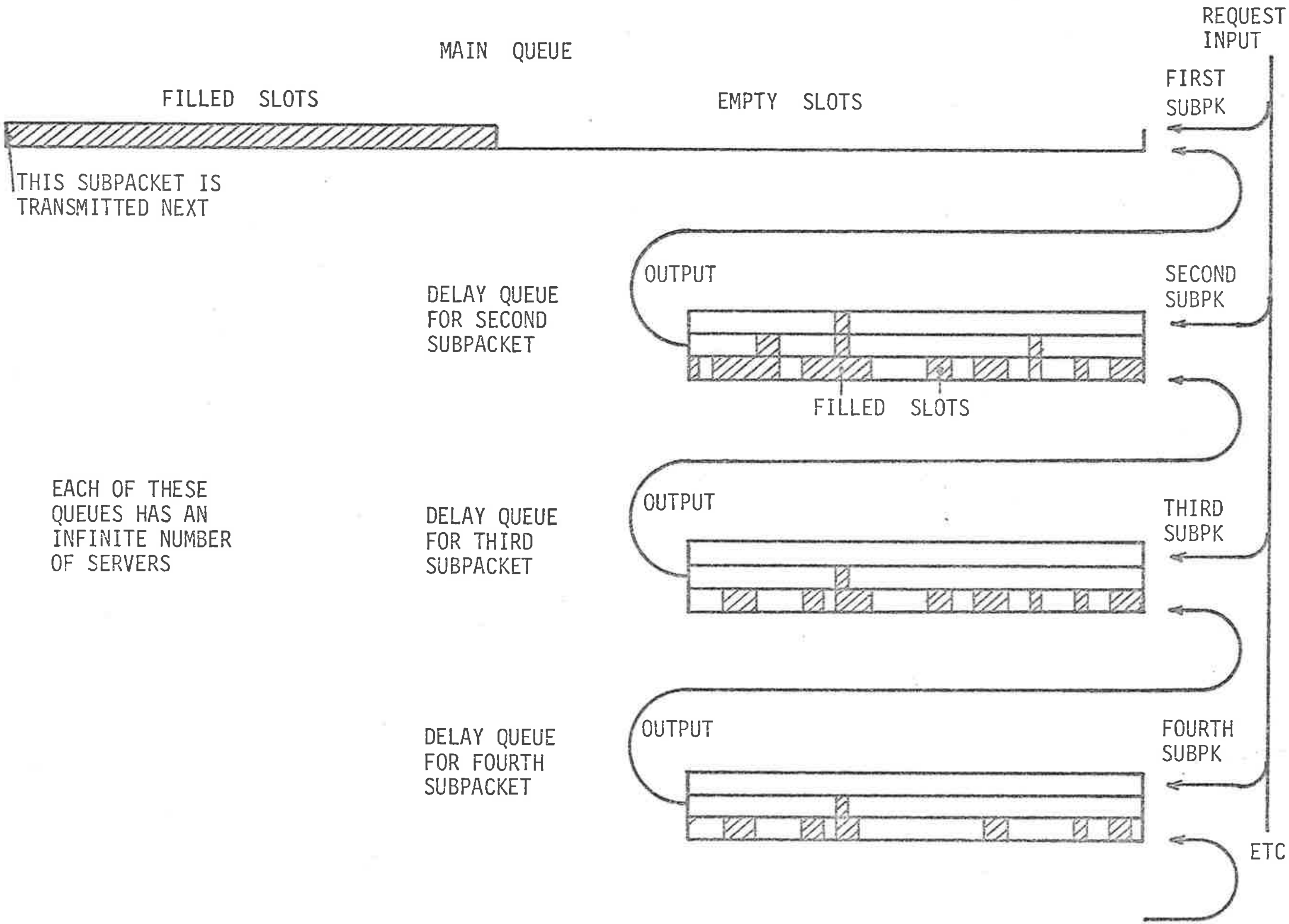


Figure 4.2 Equivalent multiqueue arrangement of system with subpackets

Correct priority is maintained in the queue system of figure 4.2 because the X'th subpacket, after being delayed by C slots, has the same priority as the (X - 1)'th subpacket of a request just arriving. With this system it is quite easy to see that if the number of requests arriving at time slot i is f_i, then the total input to the first or actual queue is

$$h_i = f_i + f_{i-C} + f_{i-2C} + \dots + f_{i-(M-1)C} \tag{4.12}$$

The input to the queue is therefore not independent since the present input is related to that in the past. Queue systems with dependent inputs have been studied by Gopinath and Morrison [53]. They propose a multidimensional state approach in which the number of dimensions equals the number of slots over which the input is correlated. In the queue with subpackets this is

$$y = (M - 1)\ C \tag{4.13}$$

Thus each state is a y-tuple of values and there are of the order of $(L + 1)^{y}$ different states. For the example system above this number is 10^{476}. Clearly it is impractical to calculate the state probabilities for any but the smallest systems.

It is possible to calculate the marginal state probabilities of which there are (L + 1) y. These are equivalent to the state probabilities in the one dimensional case. Unfortunately these alone are not sufficient to calculate the glitch rate in a queue of limited length. It is shown in appendix B that the individual state probabilities are required in this case. Hence the theoretical problem remains intractable.

Bounds on the glitch rate of systems using subpackets may be found from the theory for a single transmission per request (termed the single subpacket case). Consider for simplicity, the two subpacket case (M = 2) in which the slot length is $Z' = \frac{Z}{2}$ and the queue limit is $L' = 2\ L$. A lower bound on the glitch rate may be obtained by assuming that the input of the two subpackets is described by two independent, identical Poisson distributions, each with an average of $\frac{U}{2}$ arrivals per slot.

Probability theory states that independent Poisson processes combine to give a single Poisson process, with an average equal to the sum of those of the constituent processes. In this case the resultant input process has an average arrival rate of U per slot. This is identical to the input in the single subpacket queue. However, since the maximum allowed length of the queue is $L' = 2 \cdot L$, the glitch rate is that of the single subpacket queue with twice the normal queue limit.

In general, if the M subpackets from one request are considered to be independent, the glitch rate is given by the above theory with a queue limit of M L. Since the subpackets are highly correlated, the glitch rate has to be greater than this and it forms a lower bound.

The alternative extreme treatment of the two subpacket case involves placing the subpackets in adjacent slots. The total average input rate is again U subpackets per slot. Since two slots are occupied for each request, the slot length is effectively $2\ Z' = Z$ and hence the number of requests that can be placed within the time limit is $\frac{L'}{2} = L$. This arrangement is therefore exactly equivalent to the single subpacket case with a queue limit of L.

Thus as the correlation between subpackets varies from total to zero, the effective queue limit for calculating glitch rates varies from L to L M. This range however, permits a variation in the glitch rate by a factor of at least M. In addition, the actual glitch rate for any given number of subpackets may well change with other system parameters.

Any further theoretical work on the effect of subpackets on the glitch rate cannot be supported by rigorous theory. It is therefore necessary, considering the great uncertainty involved, to determine the effects by simulation methods. This will also provide a verification of the theory developed for the single subpacket case. When results are available from the simulation, an attempt will be made to justify them by a heuristic approach.

4.4 Computer Simulation Programme

The purpose in simulating the queue system with subpackets is primarily to find at least approximate solutions to the above theoretical queue problem. However the basic structure is sufficiently flexible to incorporate request inputs from simulated speech.

A discrete time system of unit equal to one slot length is employed. At each slot time, a random number of requests is generated according to a Poisson process with average U. These requests are each assigned M places in the queue. To maintain correct priority, every subpacket possesses a priority number equal to the first time slot in which it can be transmitted.

Queue slots are numbered sequentially, and a new subpacket can be assigned only to a slot of number greater than or equal to its priority. If the optimum slot for any subpacket is already occupied, a search is done to find the first empty slot or the first containing a subpacket of lower priority. The new subpacket is inserted into this slot, and if necessary, later entries in the queue are shifted back one position.

The advantages of this method are that each subpacket is delayed as little as possible and that no unnecessary spaces are left in the queue. Whenever a subpacket is to be added, its priority is compared to the number of the slot it is to occupy, and if the difference is greater then L M, the subpacket is glitched. Similarly, a comparison is performed after a successful addition to the queue, for all shifted subpackets, and if the difference is too large the subpacket in question is glitched and the shifting process stops.

At every slot time the first entry in the queue is transmitted and the entire queue is shifted one position. The difference between the transmitted subpacket's priority and its slot number is a measure of the delay in the system. In the programme, this delay is averaged over a period of several slots and then printed to indicate the current state of the queue. Figure 4.3 shows an example output for the case with 10 subpackets per request and a usage of 1.0. Each of the 10 delay figures in every row is averaged over 100 slots. At the end of the row are the current slot number (representing time) and the total number of subpackets transmitted to that point. A flowchart of this programme and the computer listing are provided in appendix C.

At the beginning of the computer run there is little delay because the queue is initially empty. Over a period of several thousand slots, the delay builds up and declines, and eventually reaches the limit, which in this case is 200 slots. When this occurs glitches begin, as

```
SYSTEM CAPACITY C = 40    QUEUE LIMIT L = 20    USAGE U = 1.00

NUMBER OF SUBPACKETS PER PACKET = 10

NUMBER OF SLOTS WITHIN SLOT ASSIGNMENT LIMIT = 200

NUMBER OF SLOTS OVER WHICH DELAY IS AVERAGED = 100

                         AVERAGE DELAY                          CURRENT   NO.SUBPK
                                                                 SLOT     TRANSMTD

   0.    0.    0.    2.   11.   22.   29.   37.   41.   41.     1000        787
  42.   40.   20.    1.    1.    1.    4.    4.    1.    1.     2000       1690
   1.    2.    4.   12.   24.   38.   36.   21.    1.    0.     3000       2591
   0.    2.   12.   29.   51.   69.   76.   69.   59.   60.     4000       3561
  66.   69.   75.   75.   67.   63.   61.   61.   69.   82.     5000       4561
  91.  100.  108.  104.   86.   59.   43.   32.   26.   11.     6000       5561
   2.    2.    2.    9.   21.   36.   57.   75.   73.   62.     7000       6535
  46.   45.   48.   46.   29.    2.    2.    3.    9.   17.     8000       7491
  27.   26.   22.   26.   33.   49.   62.   70.   72.   66.     9000       8491
  71.   77.   74.   73.   63.   65.   71.   72.   64.   37.    10000       9491
  26.   22.   22.   25.   27.   34.   49.   68.   86.  101.    11000      10491

** GLITCHES **   24     TOTAL  24     USAGE 97.54%    GLITCH RATE .224%
119. 137. 151. 170. 191. 198. 196. 195. 194. 195.    12000      11491
198. 197. 195. 197. 192. 172. 150. 141. 153. 169.    13000      12491
180. 179. 167. 152. 145. 142. 140. 142. 142. 141.    14000      13491
142. 151. 163. 178. 192. 189. 161. 128. 107. 106.    15000      14491

** GLITCHES **  120     TOTAL 144     USAGE 98.99%    GLITCH RATE .970%
119. 138. 161. 186. 198. 199. 195. 194. 198. 199.    16000      15491

** GLITCHES **    7     TOTAL 151     USAGE 99.10%    GLITCH RATE .952%
199. 199. 198. 195. 190. 181. 171. 162. 164. 165.    17000      16491
168. 157. 122.  99.  93.  95. 105. 113. 117. 130.    18000      17491
142. 153. 161. 170. 167. 160. 151. 146. 146. 146.    19000      18491
139. 124. 121. 131. 142. 150. 158. 157. 157. 167.    20000      19491


AVERAGE USAGE = 99.24%    GLITCH RATE = .799%

AVERAGE DELAY = 93.7 SLOTS
```

Figure 4.3 Example output from computer simulation can be seen.

To determine whether the initial conditions have any long term effects on the simulation, a run was performed with the queue initially full. It was found however that after a few thousand slots the delay, and hence the queue, returned to exactly the state it had been in when initially empty. This occurs because the queue entries at the start, and later entries delayed because of them, are transmitted in what would have been empty slots. As soon as the appropriate number of such slots have passed, the queue behaves as if the initial entries did not exist.

When glitches occur, their number is printed as are the average glitch rate and average usage. The glitch rate is simply the number of glitches expressed as a fraction of the number of subpackets offered. Similarly, the usage is the number of subpackets offered expressed as a fraction of the number of transmission slots passed. It is necessary to measure the usage, because even over relatively long runs the average input rate can differ by a small but significant amount from the nominal chosen value.

These statistics are accumulated throughout the run, and at the end an average value for the glitch rate and for the usage are calculated. The first few thousand slots are omitted from these calculations to remove any bias due to the initial emptiness of the queue. In practice, far longer runs than that of figure 4.3 are used. The output is not considered reliable until several thousand glitches have occurred. At low glitch rates however, this is often difficult to arrange, even with simulations extending over a million slots.

The programme was used firstly to verify the theoretical glitch rate derivation of section 4.2. This was done by simulating the single subpacket case ($M = 1$) at queue limits of $L = 20$ and 56 for various usages. The points obtained are shown in figure 4.4 together with the theoretical curves for these values of L. Clearly there is very good agreement. Only at low glitch rates do the simulation points differ by more than a few percent from the theoretical values. This provides ample proof, both that the solutions from theory are correct, and that the programme is functioning without error.

Next, the 10 subpacket case for $L = 20$ was simulated and the results are shown in figure 4.5. Comparison with the theoretical curve shows that the use of subpackets has produced only a small reduction in the glitch rate. This is unexpected since the theoretical investigation of subpackets promised a significant reduction. To understand this, further investigations were undertaken on different numbers of subpackets. However, before these are described it is worth noting some of the practical limitations discovered in the $M = 10$ simulation, and the implications they hold for the accuracy of the results obtained.

Much greater variation in the glitch rate was found for $M = 10$ than for $M = 1$. This arises from the greater variance of the input in the former case. For a usage of U in the single subpacket case, the average number of inputs to the queue per slot is U. Since this input is a Poisson process, its variance is equal to its mean, and is also U.

If there are M subpackets per packet length, the slot length is divided by M and the average number of requests per slot is $\frac{U}{M}$. However, since each request inserts M subpackets into the queue, the average rate

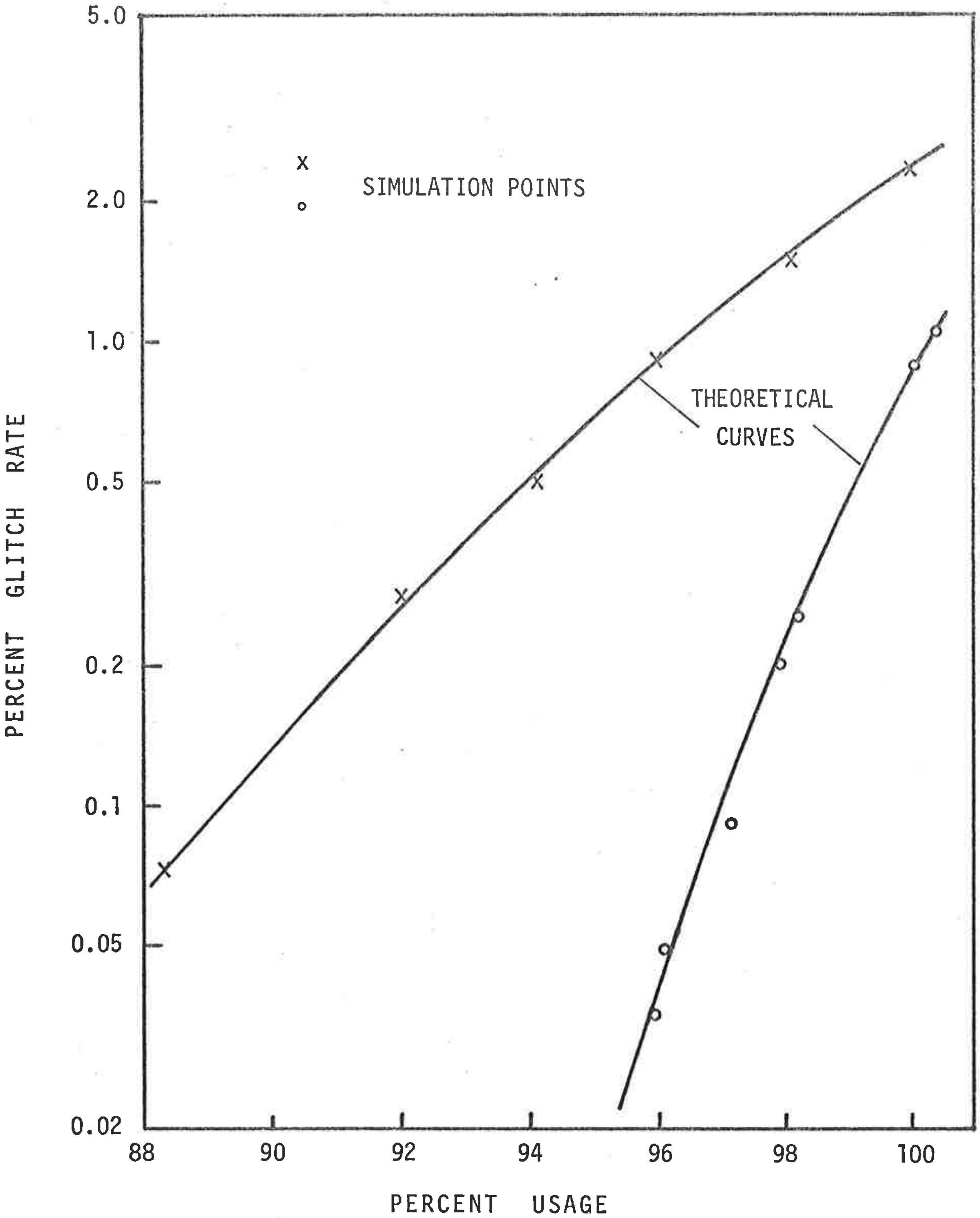
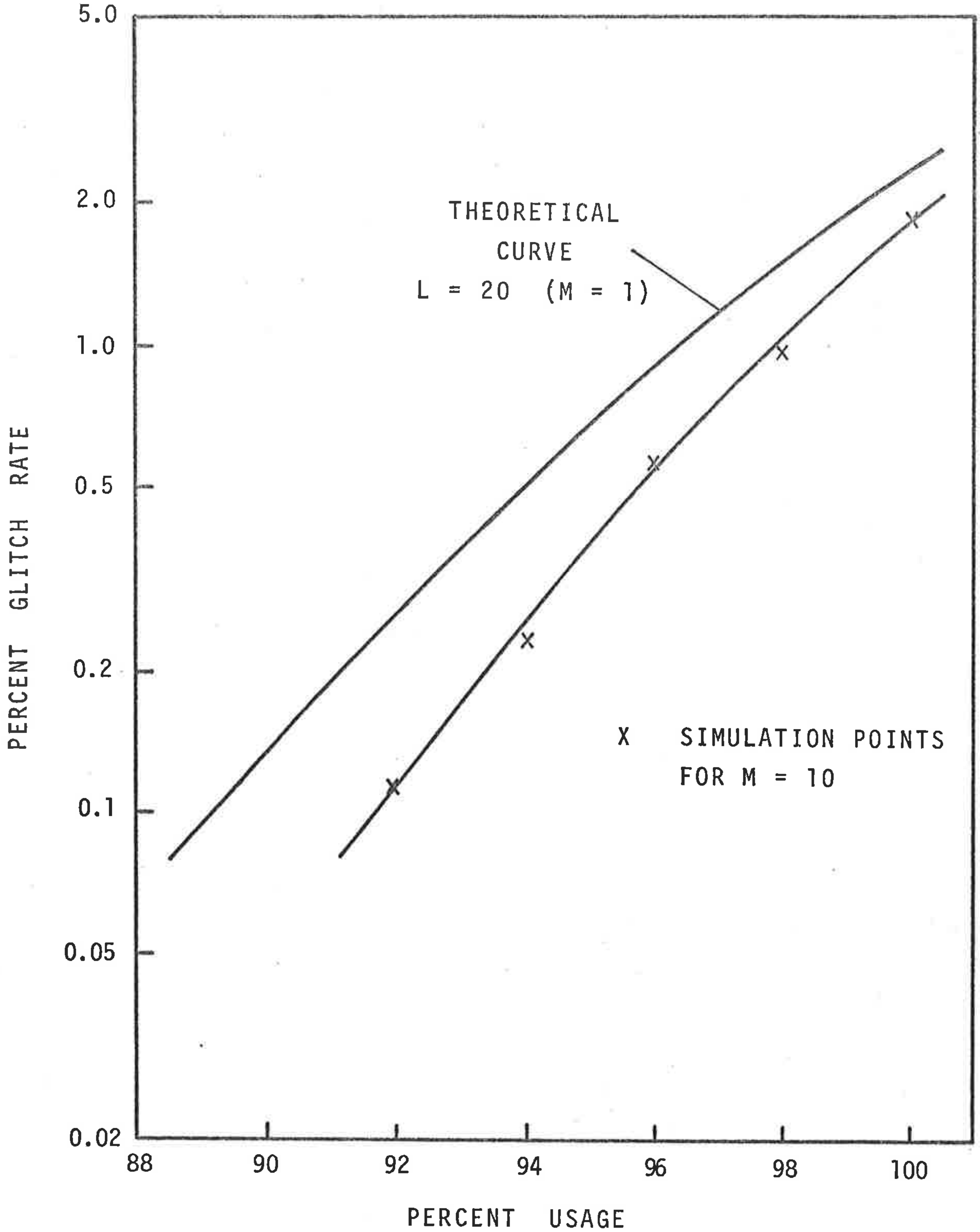


Figure 4.4 Confirmation of glitch theory by simulation



<u>Figure 4.5</u> Glitch rate from simulation of 10 subpacket case

of queue input is unchanged at U. Now the request arrival still forms a Poisson process and hence its variance is equal to its mean and is given by

$$\text{Var [request input]} = \sum_{k=0}^{\infty} (k - \frac{U}{M})^2 \; P(k) = \frac{U}{M} \qquad (4.14)$$

where P (k) is the probability that k requests arrive in one slot.

The variance of the number of subpackets added to the queue is

$$\begin{aligned} \text{Var [queue input]} &= \sum_{k=0}^{\infty} (M\,k - U)^2 \; P(k) \\ &= M^2 \sum_{k=0}^{\infty} (k - \frac{U}{M})^2 \; P(k) \\ &= M\,U \qquad (4.15) \end{aligned}$$

Thus as the number of subpackets increases, so does the variance of the queue input and hence the variance of the glitch rate.

To achieve the same variance of the final mean glitch rate with M subpackets as with one, the former would have to be run M times as long [93]. Thus for example, if 2000 glitches were sufficient to give an accurate average glitch rate in the single subpacket case, then 20,000 would be required with M = 10. This is especially difficult to achieve at low glitch rates. For instance, at a rate of 0.2% the programme would have to run for over 10^7 slots. On the CDC Cyber 173, which was used for the simulations, this would require 90 minutes of central processor time - for just one point.

In general runs were restricted to 10^6 slots, although up to 3×10^6 slots were used on a few occasions. The resultant inaccuracies at low usage rates and high numbers of subpackets had to be accepted, and care was required in assessing the results.

4.5 Optimization of Subpackets

Simulations were performed with a queue limit of 20 slots and various numbers of subpackets, for usages between 0.9 and 1.0. The resulting curves appear in figure 4.6. The curve for any particular number of subpackets possesses the same shape as the theoretical curve, but is shifted by a small amount towards lower glitch rates. At any fixed usage the glitch rate firstly decreases as M is increased from one, but reaches a minimum and thereafter increases with M. This is more clearly shown in figure 4.7 for usages of 1.0, 0.96 and 0.92.

The decrease in glitch rate from the single subpacket case is greater at lower usages. This is consistent with the changes observed when increasing the queue limit L (see figure 4.1). If at each usage the number of subpackets is chosen to minimize the glitch rate, the optimum curve formed lies very close to the L = 26 curve. Figure 4.7 shows that for usages less than 1.0 this optimum number of subpackets is quite close to 4.

To understand why there is an optimum number of subpackets, it is necessary to examine the correlation between elements in the queue. Obviously, with a Poisson process request generation, there can be correlation only between subpackets from the same request. One measure of this correlation is the number of possible transmission slots that adjacent subpackets have in common. The distance between the start of successive subpackets is C slots and the total number of slots available to each is L' = L M. Thus the fraction by which the field of one subpacket overlaps that of the subsequent one is

$$\text{Overlap} = 1 - \frac{C}{L\,M} \tag{4.16}$$

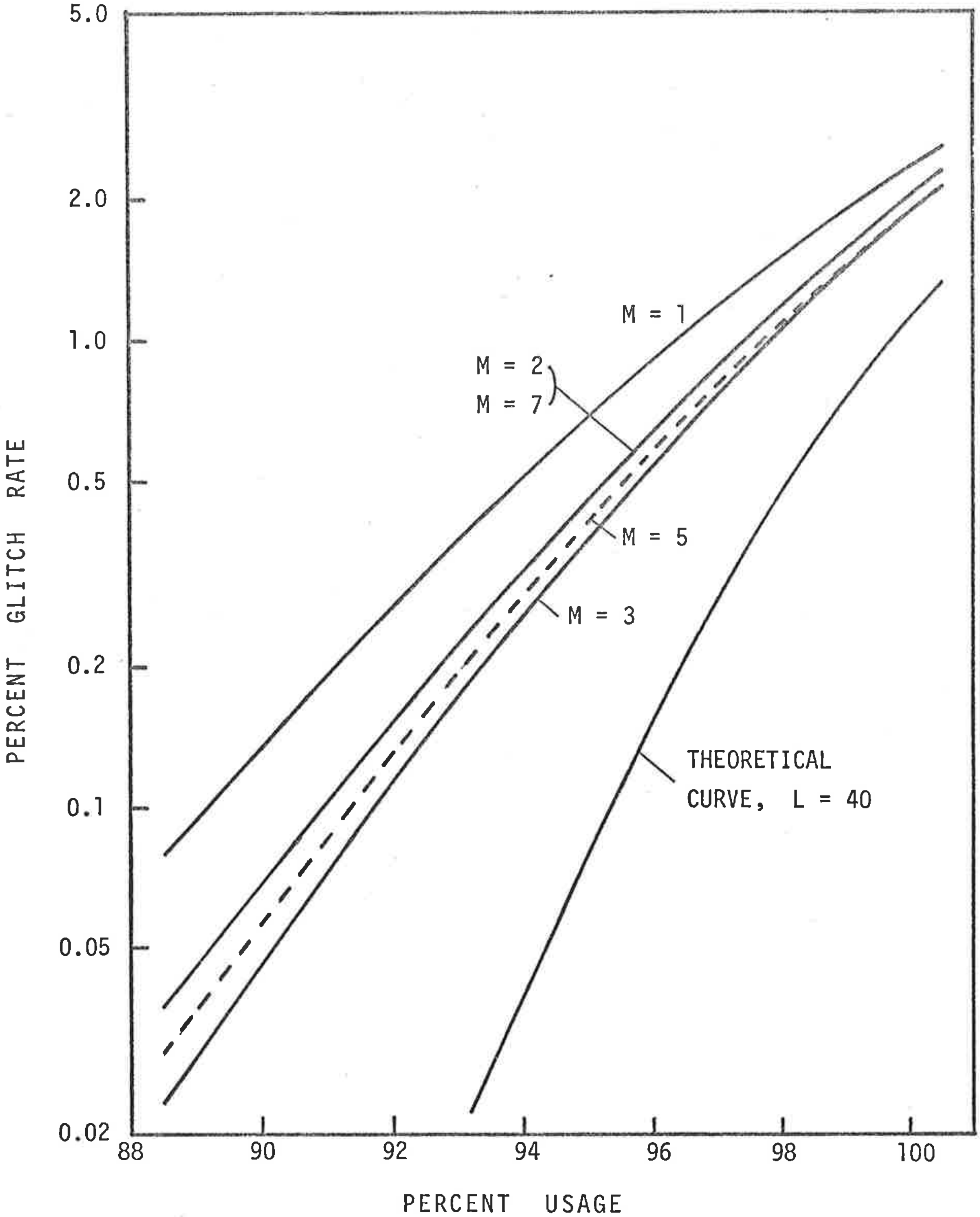


Figure 4.6 Glitch rate curves for various numbers of subpackets

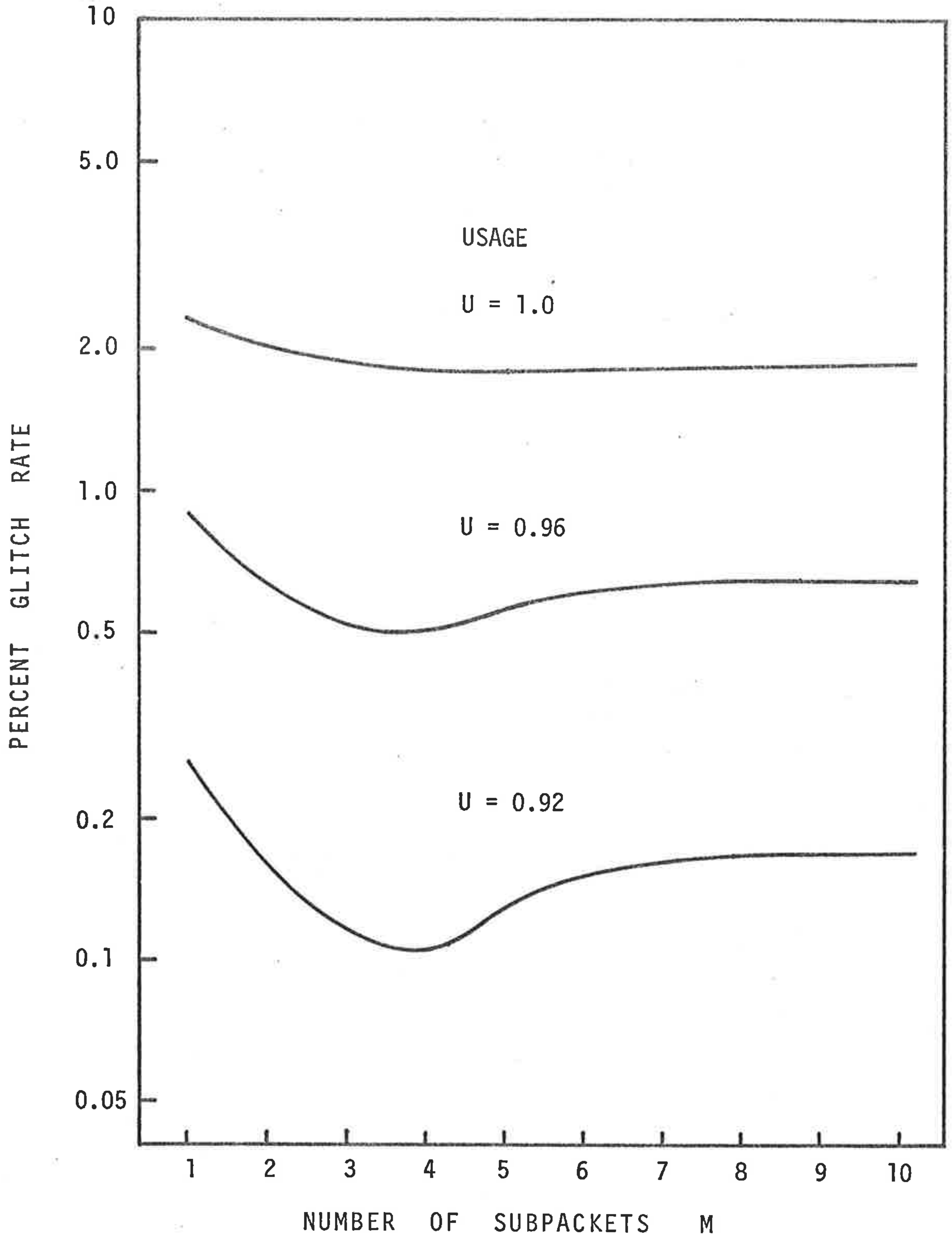


Figure 4.7 Variation in glitch rate with number of subpackets for different usages

Clearly this fraction and therefore the correlation increase with M. At the same time the number of subpackets with which any one can interfere also increases. Therefore, if the queue limit L' is held constant the glitch rate will rise quickly with the number of subpackets per packet.

However the queue limit also increases with M and, by itself, would result in a smaller glitch rate. Thus it is not surprising that there is an optimum value of M. At small M the correlation is at its lowest, while the effective queue length is significantly increased, and therefore a reduction in glitch rate occurs. Yet as M is increased the correlation quickly becomes great enough to reduce the effective queue limit from L M toward L, as discussed earlier.

Equation (4.6) implies that the correlation between subpackets will decrease if C is increased. This is physically reasonable since the effect of a larger C is to spread the subpackets out. The glitch rate should therefore decrease with increasing C and this is confirmed in figure 4.8. Note that again the optimum number of subpackets is 4 in all cases.

Because of the difficulty experienced in obtaining accurate points at low usage rates, with a large number of subpackets per packet, the behaviour of the system is somewhat uncertain in this region. To provide further information simulations were performed for M up to 20 in a small system with L = 5 and C = 10. Unfortunately, this proved to be inconclusive, as can be seen from figure 4.9. Therefore, to tidy up the behaviour at very large numbers of subpackets the case of an infinite number is now considered theoretically.

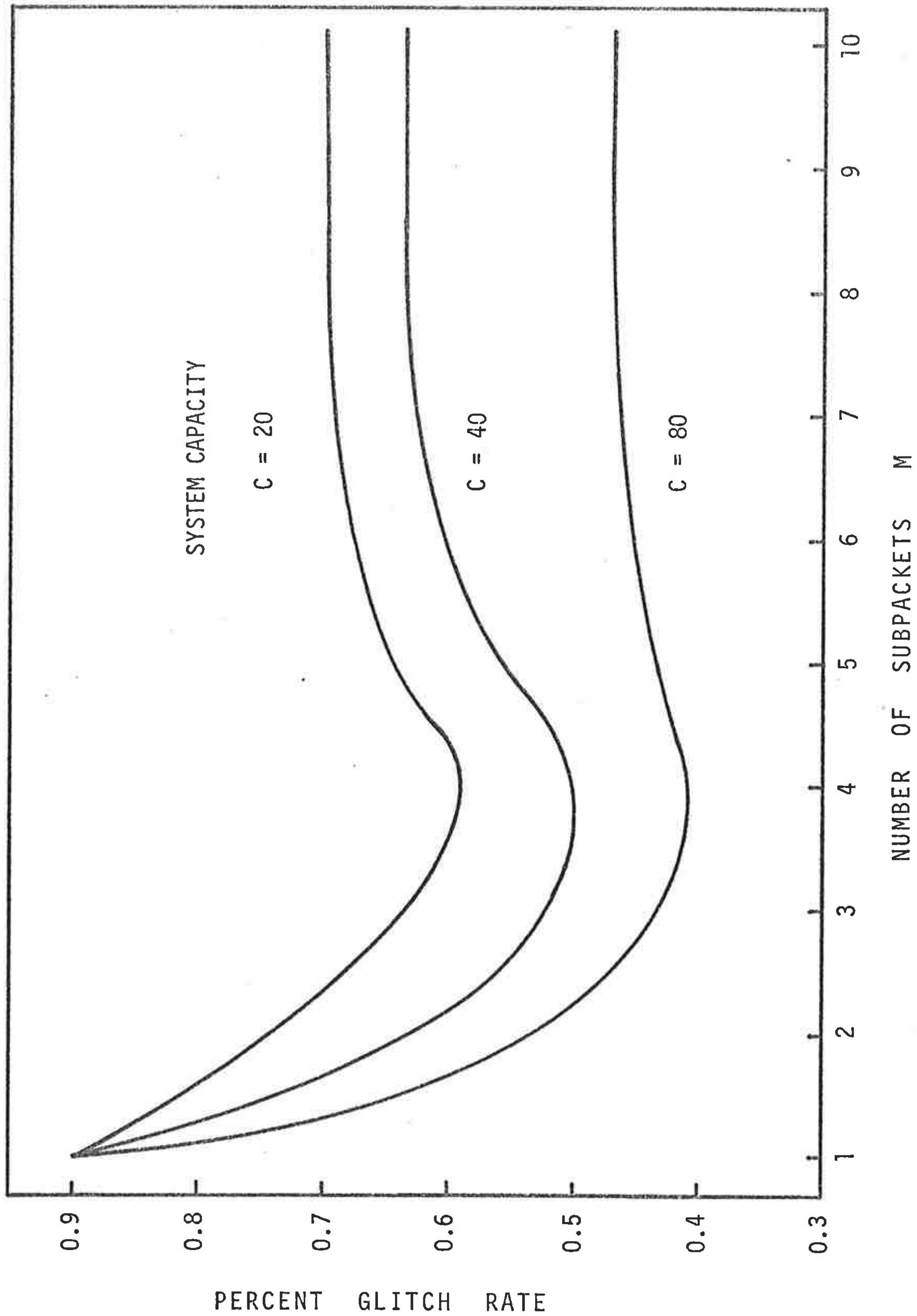


Figure 4.8 Variation in glitch rate with number of subpackets for different system capacities

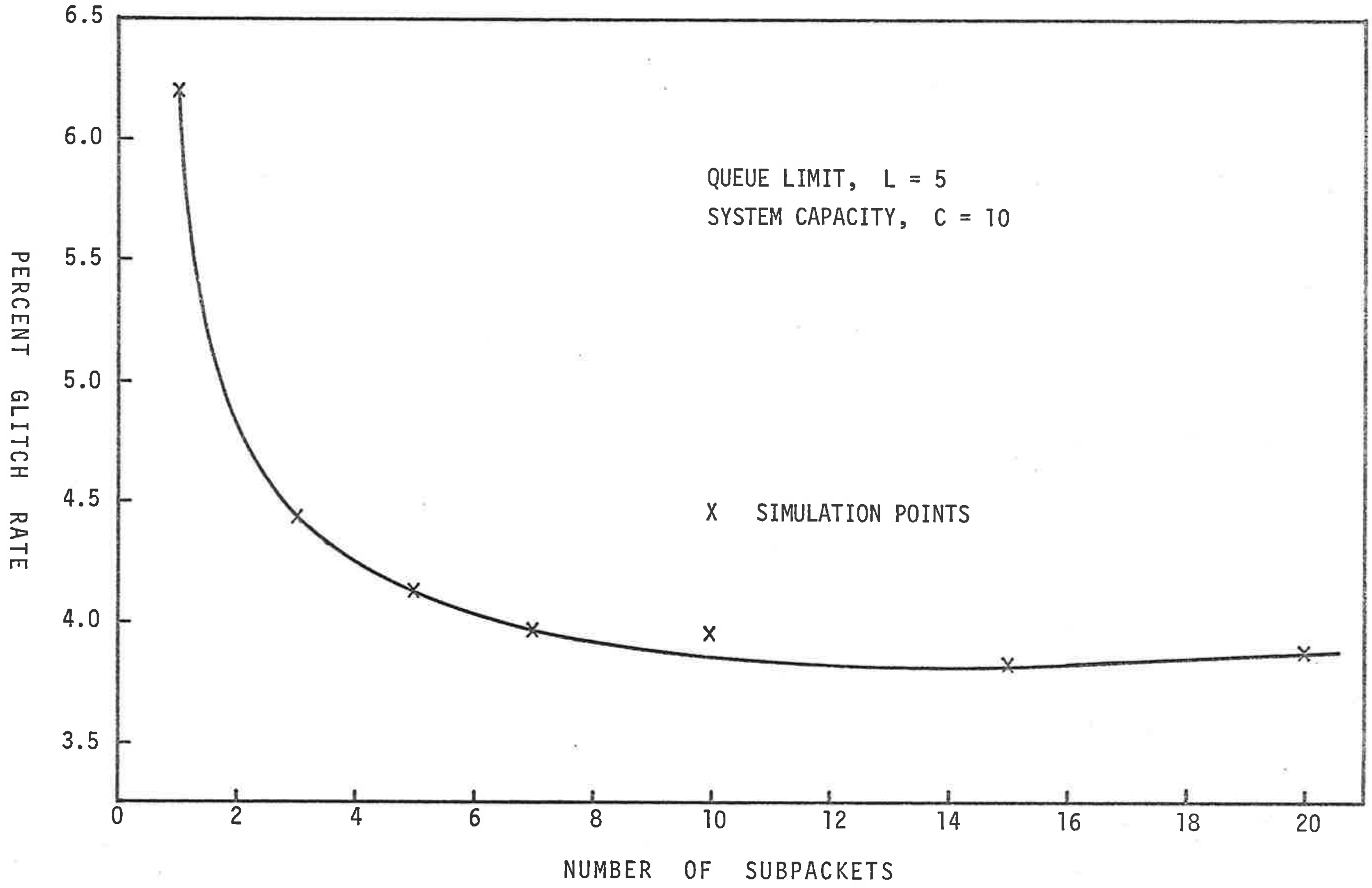


Figure 4.9 Variation in glitch rate with large numbers of subpackets

As the number of subpackets per packet M, tends to infinity, the slot length Z', tends to zero (from (4.10)) while the time for slot assignment V, remains fixed. The number of slots during a subpacket recording time, still equals the nominal number of voice circuits, C. This is equivalent to a continuous transmission system in which packets from different sources are transmitted in parallel, with each packet requiring $\frac{1}{C}$ of the channel capacity for T seconds.

The recording delay is zero and for simplicity it is assumed that other pre-transmission delays are also zero. Thus a packet is transmitted as it is prepared but it may be delayed at any point or any number of points up to a total of V seconds, without glitching. Such a delay will occur only when more than C packets are transmitted simulaneously. This causes the packets to be spread out in time so that they occupy less than $\frac{1}{C}$ of the channel capacity and take longer than T seconds to transmit. At any point in time, the delay on each packet is identical, and is increasing if more than C packets are being transmitted and is zero, or decreasing toward zero, if less than C packets are being transmitted.

Now the total delay and channel usage are unchanged if instead, packets use the entire channel capacity for a time of $\frac{T}{C}$ seconds. Then packets from different sources are sent sequentially rather than in parallel. Any packet that arrives while the channel is in use has to wait for the current transmission to finish, and for any other packets ahead of it to be sent. The delay on any packet may again be up to V seconds before glitches occur.

Packet generation still forms a Poisson process with an average rate of U starting every $\frac{T}{C}$ seconds. Thus this sequential arrangement has exactly the same input process as the single subpacket system where the slot length is $\frac{T}{C}$ seconds. The only difference between the two schemes is the continuous nature of the starting times in the infinite subpacket case. If here transmission is constrained to begin only at the start of a slot, then the queue systems are identical and they produce identical glitch rates.

Quantizing the starting time in fact makes no difference to the average glitch rate, for the delay in slot assignment is unchanged. This was confirmed by simulation of the random starting time sequential system. It must therefore be concluded that as M becomes large the glitch rate will increase asymptotically toward the single subpacket case.

A final comment should be made concerning the generality of the results obtained in this section. The problem considered is that of a limited length queue with random input of fixed length packets. Each packet is split into subpackets which enter the queue at regular intervals. It is found that for a given limit on time (or memory) the split packets produce less overload than those that are transmitted whole. The magnitude of the overload reduction depends upon various system parameters and in particular is greater for lower input rates.

It is also noticeable that for a wide range of conditions the optimum number of subpackets is 4. This optimum however, may well depend upon the queue limit L, which was largely fixed at 20 for these tests. Usually, the overload is quite close to its minimum for a range of numbers of subpackets and particularly for numbers greater than the optimum.

These results are obviously applicable to many areas involving packet transmission. The essential point is that overload in limited sized buffers may be reduced by adding a deterministic component to a previously random input.

4.6 Conclusions

In this chapter the consequences of dynamic slot assignment in the interpolation system have been investigated. To avoid possibly very large delays, a time limit for slot assignment has to be set. This inevitably means that some packets cannot be transmitted and these are said to be glitched. Such overload phenomena are present in all interpolation schemes, and work done on TASI systems suggests that the resulting speech loss should be no higher than 0.5% for acceptable quality.

The glitch rate in a packet system was determined theoretically for the simple case of one transmission per packet and an assumed Poisson process for packet generation. This glitch rate depends upon the allowed queue limit L, given by (4.9), and upon the usage as shown in figure 4.1. It was noted that the example mobile telephone system considered in chapter 3 with a slot assignment delay of 0.2 seconds and a glitch rate of 0.5%, can theoretically be run at 94% of its capacity.

Next the effect upon the glitch rate of multiple subpackets was considered. This was found to be an intractable problem theoretically, though rather loose bounds for the effects were set. The queue system was therefore simulated and the glitch rate determined as a function of the number of subpackets, for various system parameters. The glitch rate decreased from the single subpacket value, at times by factors greater than two (see figures 4.7, 4.8).

A well defined optimum number of subpackets which minimizes the glitch rate has been shown to exist in most cases. For a wide range of system parameters this number is 4. An approximate theory of the effect of subpackets was developed, which supported these simulation results, and another analysis showed that for a very large number of subpackets the glitch rate rises toward that of the single subpacket case. Thus it appears that the use of subpackets will reduce the glitch rate, as well as the recording delay.

Finally it must be noted that the solutions presented for this queue system are quite general. They are applicable to any M/D/1 queue of limited size and prove that the overflow from such a queue can be reduced by splitting the input into parts which take correspondingly shorter times to serve.

5. THE COMPLETE MOBILE TELEPHONE SYSTEM

5.1 System Simulation

Theoretical and simulation investigations in the preceding chapters indicate that a packet speech interpolation mobile telephone system is feasible. The two main problems of delay and glitching appear to have acceptable solutions which are achievable simultaneously. To date, the glitch rate determination has involved only a Poisson input and a simplified structure, not truly representative of the actual system. The only manner of taking other factors into consideration is to incorporate them into the computer simulation. For this reason the entire mobile telephone system was simulated.

All of the parameters of the mobile telephone system have been made computer variables so that their effects can be determined individually (see appendix C). Up to 100 simultaneous conversations (termed calls) are generated by the speech programme described in chapter 3. This is used as a subroutine to produce a talkspurt and a silence and to assign packets to the various talks in the proper manner. The resulting packet sequences for each call are stored with the starting time of the first packet and the finishing time of the silence. When the last packet for any call has been transmitted a new talkspurt and silence are generated, starting from the end of the previous pair. Thus the conversations are continuous and independent.

At initiation, the conversations must be in different stages if interpolation is to function properly. Their starting times are therefore distributed randomly over the first 20 seconds simulated. This figure results in a fairly rapid transition to full operation while not overloading the system at the very beginning. The starting times are then sorted into ascending order and the first talkspurt and silence interval are generated for each conversation.

The length of a request slot is taken as $\frac{T}{A\ C}$ seconds. Here the requests due to call set up and termination are ignored and hence the request channel capacity is given by (3.3). The first request slot for each conversation is calculated and stored in order in an array. At the start of a transmission cycle, the first two slot numbers are compared to determine if a collision has occurred. If the numbers are the same, subsequent entries are tested until a different number is found. Then all of the collided requests are resequenced and the cycle is begun again.

Resequencing involves adding up to R + K slots to each request slot number. Here R is a constant for any particular system and represents the time-out delay for an acknowledgement. In the programme, the value used for R is 2 slots plus 12 ms. This provides time for the transmission of a request and an acknowledgement, and a further time to cover calculation and other delays. The random component of added delay is uniformly distributed between zero and K, which is usually set at 5. Once the new request number is calculated it is inserted into the array at the appropriate position.

If the initial requests do not collide then the first is transmitted. One speech packet is removed from the call concerned and if necessary a new talkspurt and silence are generated. From the starting time of the next packet a new request slot number is found and inserted at the appropriate point in the request array. The speech packet which has just been removed is then added to the speech queue in the manner described in chapter 4.

Before the packet is actually inserted, the speech queue is brought up to date by removing those slots which have by then been transmitted. The length of a speech slot is $\frac{T}{C\ M}$ seconds but time in the programme is measured in terms of request slots. Thus the number of the last speech slot transmitted is calculated from the request slot just used.

Four extra request slots are added to the present number before this calculation is done to keep the speech queue slightly ahead in time. This is the simplest way of ensuring that there is time to transmit a speech slot number to the mobile telephone after assignment. It is necessary to delay slot assignment, and hence its notification, as long as possible, to accomodate packets that arrive late due to multiple collisions.

Each subpacket has a priority number which dictates where it may be placed in the queue. To calculate this number the starting time of the packet rather than the arrival time must be used due to the problem of collisions. The priority of the m'th subpacket is given by

$$\text{Priority} = (\text{packet starting time}) \ \frac{C\ M}{T} + m\ C \tag{5.1}$$

A subpacket may be placed in any slot in the queue numbered between that subpacket's priority and its priority plus L'.

Usage in this system is controlled by the number of calls. However the actual usage, as measured by the average number of subpackets transmitted per speech slot, often differs by more than 1% from the usage predicted. Also measured is the glitch rate given by the ratio of glitched subpackets to the number offered. The cumulative glitch rate and usage are printed periodically throughout a computer run, so that the degree of stability can be ascertained. From this and the considerations in section 4.4, the run length required by any particular system can be estimated.

For comparison with the theory developed in the preceding chapter, a simulation was done of the example interpolation system. This has a nominal circuit capacity, C, of 40 and a queue limit, L, of 20. Runs were done for both the single subpacket and the 10 subpacket systems. The resulting glitch rate curves appear in figure 5.1. Recall that the theoretical curve for single subpackets coincided with the Markov chain theory curve, also shown in figure 5.1. Obviously the glitch rate in the complete system simulation is somewhat greater. The glitch rate with 10 subpackets per request is however lower than that with 1. Thus it appears that the use of multiple subpackets still reduces the glitch rate.

Possible causes of the higher glitch rate in the full simulation are the delays involved in request transmission and the non-random nature of the speech input. The possible effects of the request channel, and in particular of collisions, will be considered in detail in the next section.

Although the input from different calls is completely uncorrelated, the sequence of requests from any single call is far from random. With a packet length of 0.4 seconds, the probability that a packet will be followed, without a space, by another from the same source is 0.81.

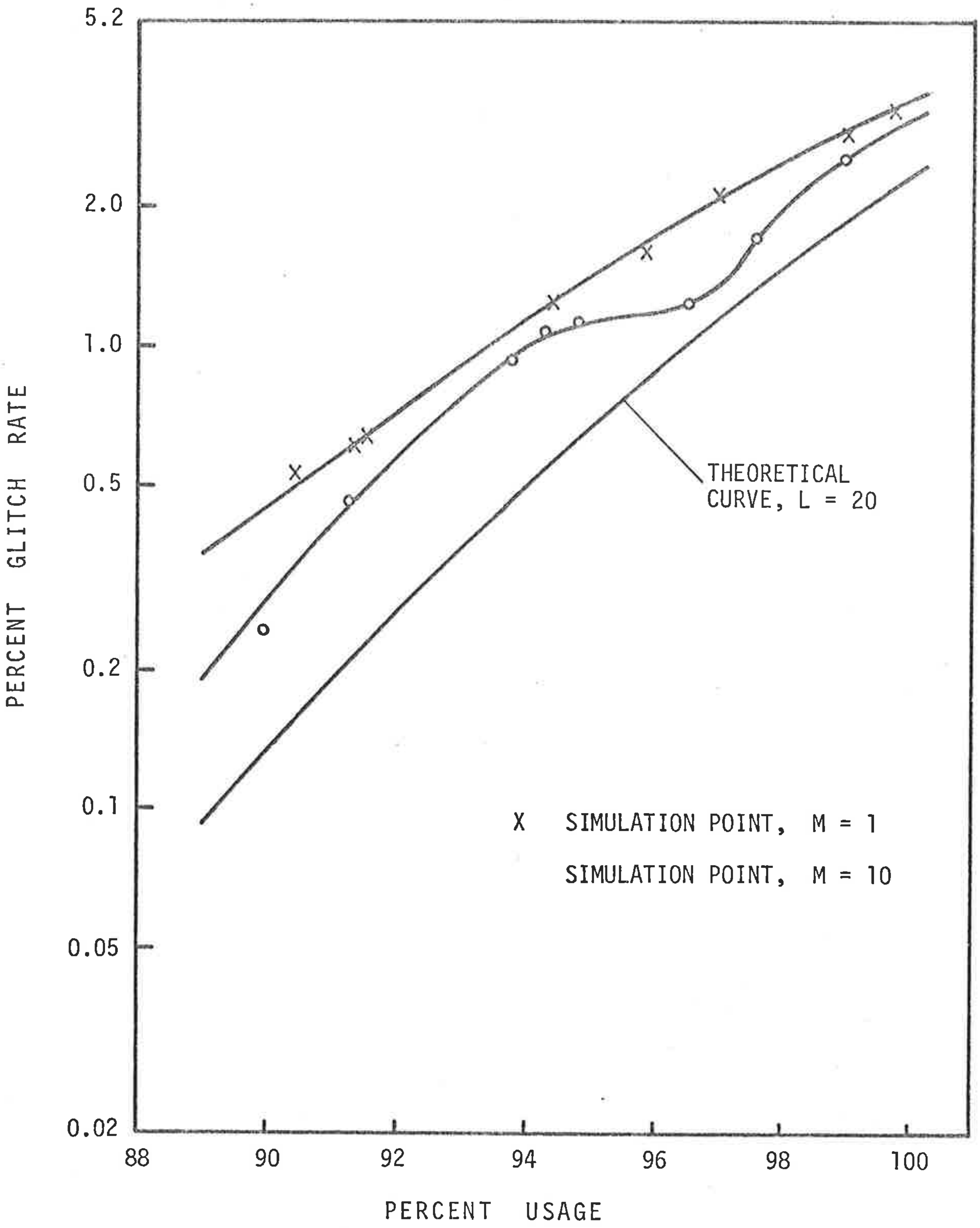


Figure 5.1 Full simulation of standard mobile telephone system

If packet generation was entirely random the probability of successive packets would be $\frac{U}{N}$ which for practical systems is around 0.01. In fact the correlation in the input extends over several packets. Figure 5.2 shows the probability of sequences of various numbers of packets with a 0.4 second packet length. The average number in a sequence is 5.2.

This problem originates in the bursty nature of speech. Packet generation follows the speech, with individual sources producing a number of packets separated by exactly one packet length, and then a large space until the next sequence. This must increase the variance of the total request input and hence the glitch rate. The effect is greater for smaller packet lengths since these result in longer sequences of adjacent packets. This will be demonstrated later when the variation of system parameters is considered.

5.2 Effect of the Request Channel

If a request experiences few or no collisions, it will arrive at the central controller before the subpacket has finished recording. This will leave the full field of slots available for assignment. If sufficient collisions occur however, the request will not arrive until some of its possible transmission slots have passed. This obviously reduces the field available to that subpacket and may cause it, or another subpacket, to glitch unnecessarily. In the worst case the request will collide so often that it does not arrive until the slot assignment limit for the subpacket has passed.

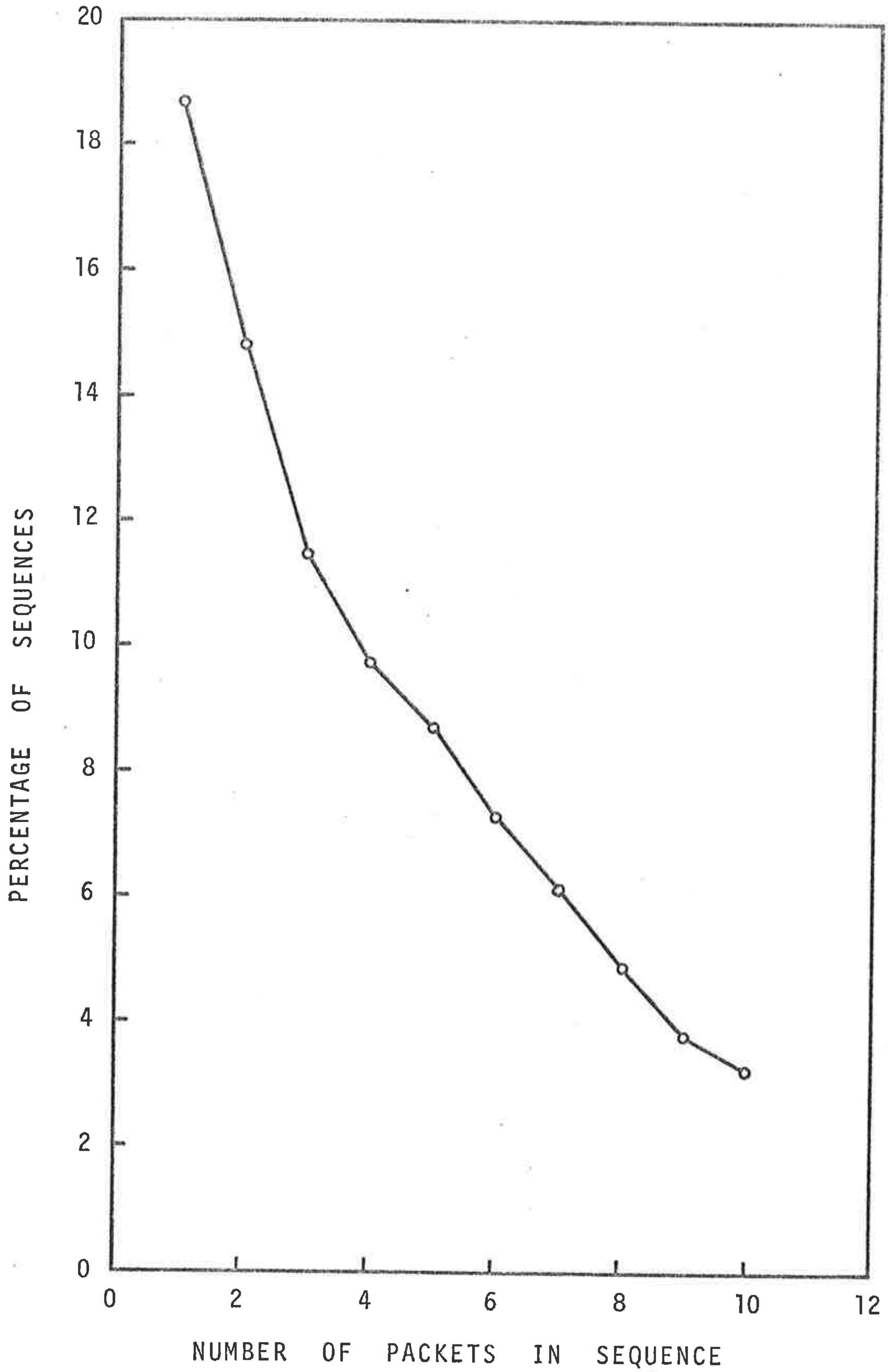


Figure 5.2 Distribution of the number of packets in a sequence

To determine the probabilities of each of the three cases mentioned, the times involved must be expressed in terms of request channel slots. The time of a request slot in the programme is

$$Z_r = \frac{T}{A\ C} \tag{5.2}$$

The time taken for resequencing is on average $R + \frac{K}{2}$ slots i.e.

$$L_r = 2 + \frac{.012\ A\ C}{T} + \frac{K}{2} \tag{5.3}$$

The number of slots until the first possible subpacket transmission slot is

$$L_f = \frac{A\ C}{M} - 4 \tag{5.4}$$

Finally, the number of request slots until the expiry of the slot assignment time limit is

$$L_t = A\ C\left(\frac{1}{M} + \frac{V}{T}\right) - 4 \tag{5.5}$$

Next the probabilities of various numbers of collisions must be determined. If all requests are assumed to arrive randomly, according to a Poisson distribution, then the simplified theory embodied in (2.2) may be used and the probability of a collision is

$$P_{col} = 1 - e^{-G} \tag{5.6}$$

In practice, the input is only random for requests that have not collided. Thus, this relation cannot be expected to accurately predict the collision probability for retransmitted packets. However, the probability of no collisions should be predicted fairly well. This was checked by simulation for some typical systems and the results are shown in table 5.1.

Prob. of no collisions	A=5, K=5	A=5, K=10	A=10, K=5
Theory	0.7717	0.7717	0.8942
Measured	0.7642	0.7714	0.8923

Note C = 40, M = 4, T = 0.4 seconds, U = 1.0, V = 0.2 seconds

Table 5.1 Probability of no request collision

Clearly the agreement is very good, especially as the actual degree of randomness in the queue improves (with larger K). The simulations also measured the probability of a request colliding a number of times, as shown in figure 5.3. Since these curves tend toward straight lines (on log paper) the probability of a collision for retransmitted requests tends toward a constant value. These values can be determined from the slope of the lines and are compared in table 5.2 with the theoretical collision probabilities calculated from (5.6)

Prob. of a collision	A=5, K=5	A=5, K=10	A=10, K=5
Theory	0.228	0.228	0.106
Measured	0.437	0.356	0.290

Note C = 40, M = 4, T = 0.4 seconds, U = 1.0, V = 0.2 seconds

Table 5.2 Comparison of collision probabilities for retransmitted packets

The simple theory obviously does not apply for retransmitted packets. The probability of a collision is much higher in this case than for new packets, because of the relatively significant probability of multiple collisions between the same two (or more) requests.

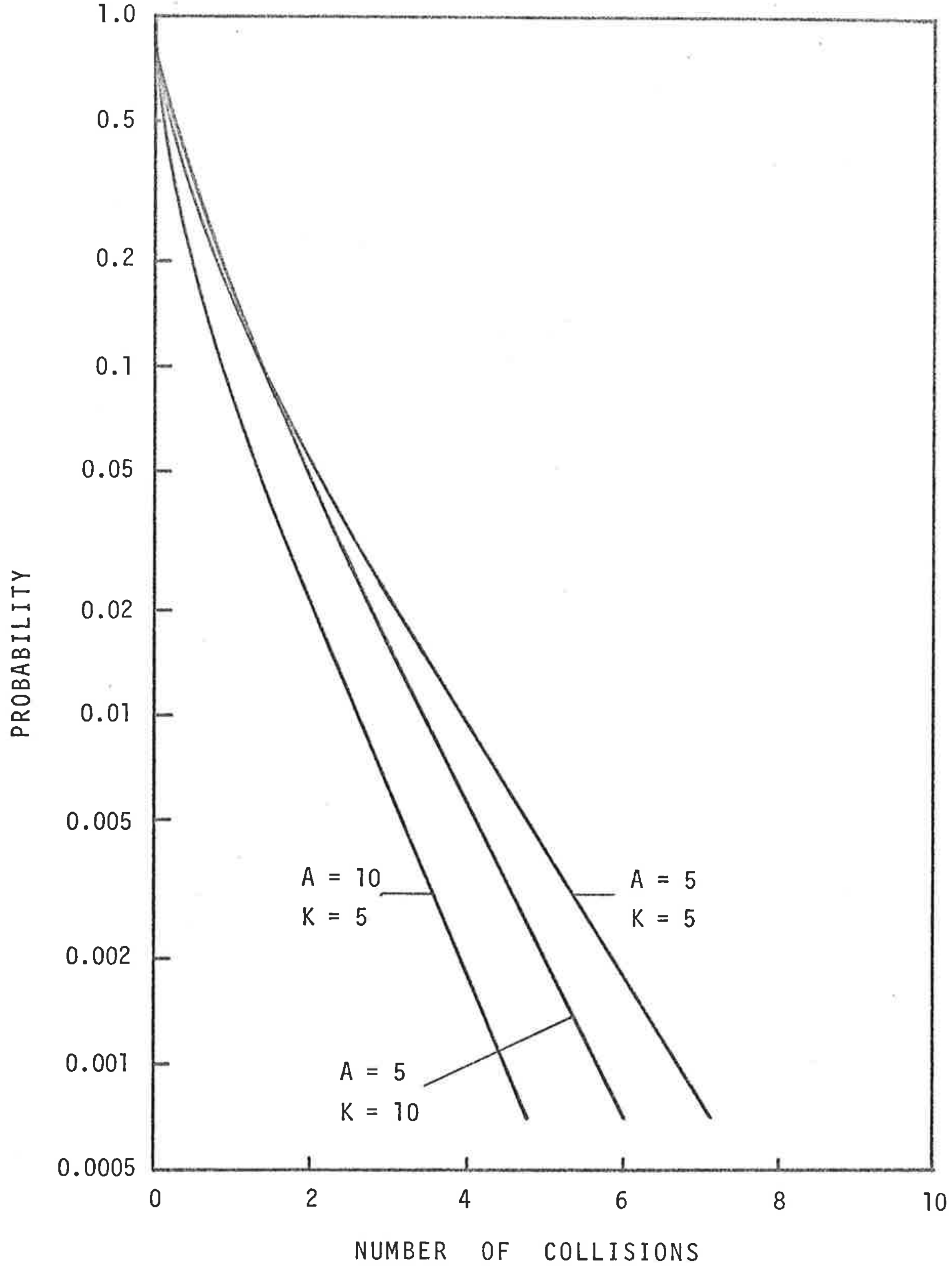


Figure 5.3 Collision distribution for various request channel parameters

The simulations support this reasoning since increasing the maximum random addition upon resequencing, K, from 5 to 10 substantially reduces the measured probability of a collision among retransmitted packets. In the case of uncollided packets, (table 5.1), the measured effect of K is much smaller.

Increasing the ALOHA factor, A, directly reduces the probability of a transmission in any slot and hence the probability of a collision. The theoretical and measured values for both new and retransmitted packets all show a substantial reduction when A is increased from 5 to 10. This is equivalent to doubling the request channel capacity.

The measured probabilities of requests colliding a given number of times can now be used to calculate the delay probabilities. These are shown for the three ranges of delay length of interest, for several example systems in table 5.3.

The probability of glitches resulting form delays exceeding the slot assignment limit is so small that it can be ignored. The only occasions upon which request collisions may cause glitches are those in which delays fall into category 2. The glitch rate due to these events cannot be obtained easily because of the dynamic assignment and reassignment that occurs in the speech packet queue.

Parameters			Slot Lengths			Probabilities of delay in Interval		
A	K	M	L_r	L_f	L_t	1	2	3
5	5	10	10.5	16	116	0.907	0.093	2×10^{-5}
5	5	4	10.5	46	146	0.983	0.017	2×10^{-6}
5	10	4	13	46	146	0.992	0.008	10^{-6}
10	5	4	16.5	96	296	0.999	0.001	10^{-10}

Notes C = 40, T = 0.4 seconds, U = 1.0, V = 0.2 seconds

Delay interval definitions :

1 delay is less than the time till the first slot

2 delay is between the time of the first slot and the assignment limit

3 delay is greater than the assignment limit

Table 5.3 Probability that a collision delay falls in various intervals

When a request of this type finally arrives it is in fact unlikely to cause any damage. For if the delay is to make any difference, the queue must have been at zero length some time between the original collision and the final arrival, so that a slot the request could have filled, was left empty. Even given this, if the queue then returns to zero length before approaching the limit, the extra subpacket will be transmitted and the status quo regained.

Thus the maximum possible damage that such late arrivals can do is to reduce the effective queue length L' by a few percent for a small fraction of the time. The resultant effect on the glitch rate is likely to be negligible compared to the effect of the slot assignment limit itself. This was confirmed by simulation.

The effect of no collisions was simulated by removing the delay and retransmission procedures. In this version of the programme, when two or more requests were scheduled for the same slot, all were considered to be successfully transmitted and their subpackets were added to the queue in sequence. The remainder of the programme worked as before and the glitch rate and usage were determined.

Many runs were done over a wide variety of system parameters but no detectable decrease in glitch rate occurred. Exact comparisons between the normal and non resequencing programmes was difficult because the usage changed slightly even with identical parameters. The observed variation in glitch rate was entirely explicable in terms of the changes in usage alone.

Another possible problem caused by collisions is an ALOHA type overload in the request channel. This will occur whenever the request input rate exceeds 0.368 of the channel capacity for a sufficient length of time. The actual mechanism for this phenomenon is as follows.

When a sudden spurt of input causes collisions to occur in a number of adjacent slots, each collision produces at least two retransmissions between R and R + K slots of its occurrence. If the number of requests involved is large enough, there will be a collision in most of the K or more slots. Hence, any new requests are very likely to collide and simply add to the number of colliding requests. This wave of collisions quickly spreads out to cover every slot and sweeps along all new requests. Since the probability of any request actually getting through becomes smaller and smaller while the input rate is fixed, there is a build up of requests and eventually the throughput becomes zero.

The probability of such a breakdown is dependent upon A, K and R. The average number of packets offered per slot (and hence the collision probability) is inversely proportional to A and the number of collisions required to start a breakdown is inversely proportional to K + R. In all of the simulations run a breakdown occurred only once. This was for a system with A = 5, K = 5 and a packet length of 1.6 seconds. The reason for the collapse was the large packet length which resulted in a value for R of 3.5 request slots.

In this case the problem was solved by increasing the value of K from 5 to 10. No breakdown occurred with a packet length of 0.8 seconds for which R is 5 request slots. Thus it would appear that for safety the sum of R and K should be kept above 10.

If a breakdown does occur in a mobile telephone system, then a special recovery procedure has to be instituted. This could involve the broadcast, over the acknowledgement channel, of an order for each telephone to drop its current packet. This will quickly reduce the request traffic to zero and normal operation should resume since it is very unlikely that the conditions which caused the original overload would still exist. If problems persisted over a period of seconds the traffic could be reduced by not replacing calls which finished. However, because of the extreme rarity of this event, the quality of speech reproduction rather than the possibility of a breakdown will restrict the number of connected calls.

5.3 Dependence of Glitch Rate on System Parameters

Simulation of the full mobile telephone system showed that the actual glitch rate exceeded that predicted by theory. This was tentatively explained in terms of delays introduced in the request channel, and of correlation in the input. However, the request channel has been shown to have little or no effect because other delays in the system render request delays unimportant. The non-random speech input must therefore be the source of the extra glitches. There is no way in which the speech can be altered to reduce this effect, however there are several system parameters which may be varied. These will be considered in turn to determine how they may be best set to minimize the glitch rate.

Consider firstly the number of subpackets per packet, M. It was shown in chapter 4 that with a Poisson input the glitch rate is reduced by increasing the number of subpackets. Figure 5.1 confirms that this is also true with a speech input. The curve for 10 subpackets however shows large variations due to the difficulty of obtaining accurate simulation results in this case. Simulations performed with 4 subpackets produce a glitch rate versus usage curve which lies virtually on top of the 10 subpacket curve but is rather less variable. This prompted simulations at other numbers of subpackets with the results as shown in figure 5.4.

For more than 2 subpackets per packet, the glitch rate is almost constant. There is no distinguishable optimum number of subpackets as there is with a Poisson input and the reduction in glitch rate at any point is smaller. Thus from a glitch rate point of view, the number of subpackets chosen is unimportant as long as it exceeds 2.

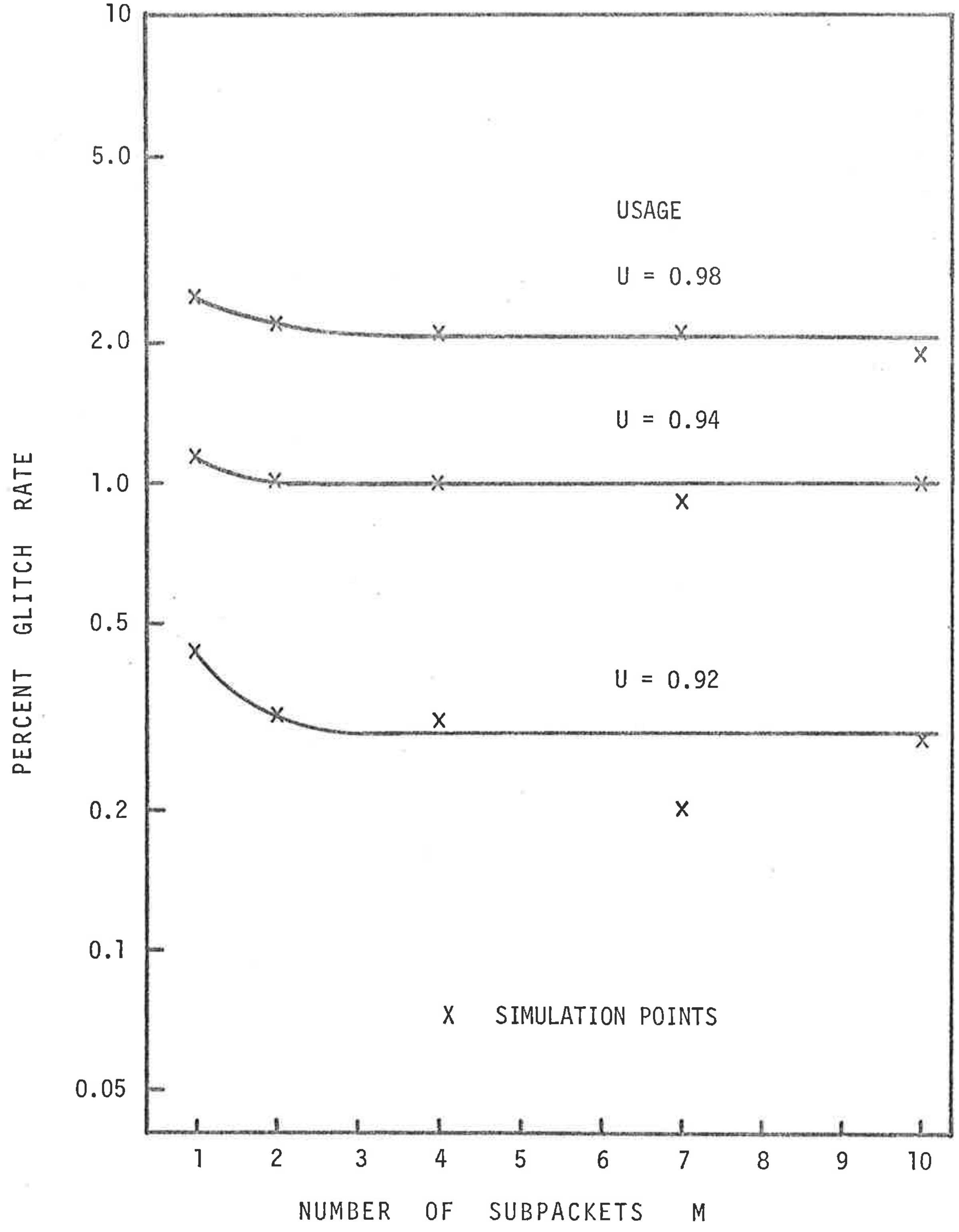


Figure 5.4 Variation in glitch rate with number of subpackets

An upper limit on M will be set by physical restraints such as the reduction in effeciency of formation and transmission with smaller subpackets. Also as M is increased, the number of slot assignment transmissions on the acknowledgement channel may increase. This was considered in section 3.5 where various techniques were found for increasing M without enlarging the acknowledgement channel. The simplest technique involves transmitting more than one subpacket per slot assignment. This retains the small recording delay that the actual value of M produces while providing a lower effective value of M for slot assignment and glitch rate determination.

Consequently there is little restriction on the value of M and reasonable recording delays can be obtained with virtually any packet length. This aspect can therefore be ignored when a packet length is chosen to minimize the glitch rate.

Up to the present time, the packet length has been held constant at 0.4 seconds. When it is changed the glitch rate versus usage curves resulting are similar in shape to those obtained previously, but are somewhat displaced. Glitch rates for various packet lengths are shown in figure 5.5 for the 4 subpacket case at usages of 0.90, 0.94 and 0.98. The importance of the packet length is clearly demonstrated by the significant increase in the glitch rate with smaller lengths.

The reason for this can be traced to the speech input. Although the most common talk length is 0.25 seconds, 50% of talks exceed 1.36 seconds and the average length is 1.79 seconds. It is therefore very common to have a number of packets transmitted in sequence. This results in a significant correlation in the speech queue and therefore a larger glitch rate than with a random input. With a smaller packet

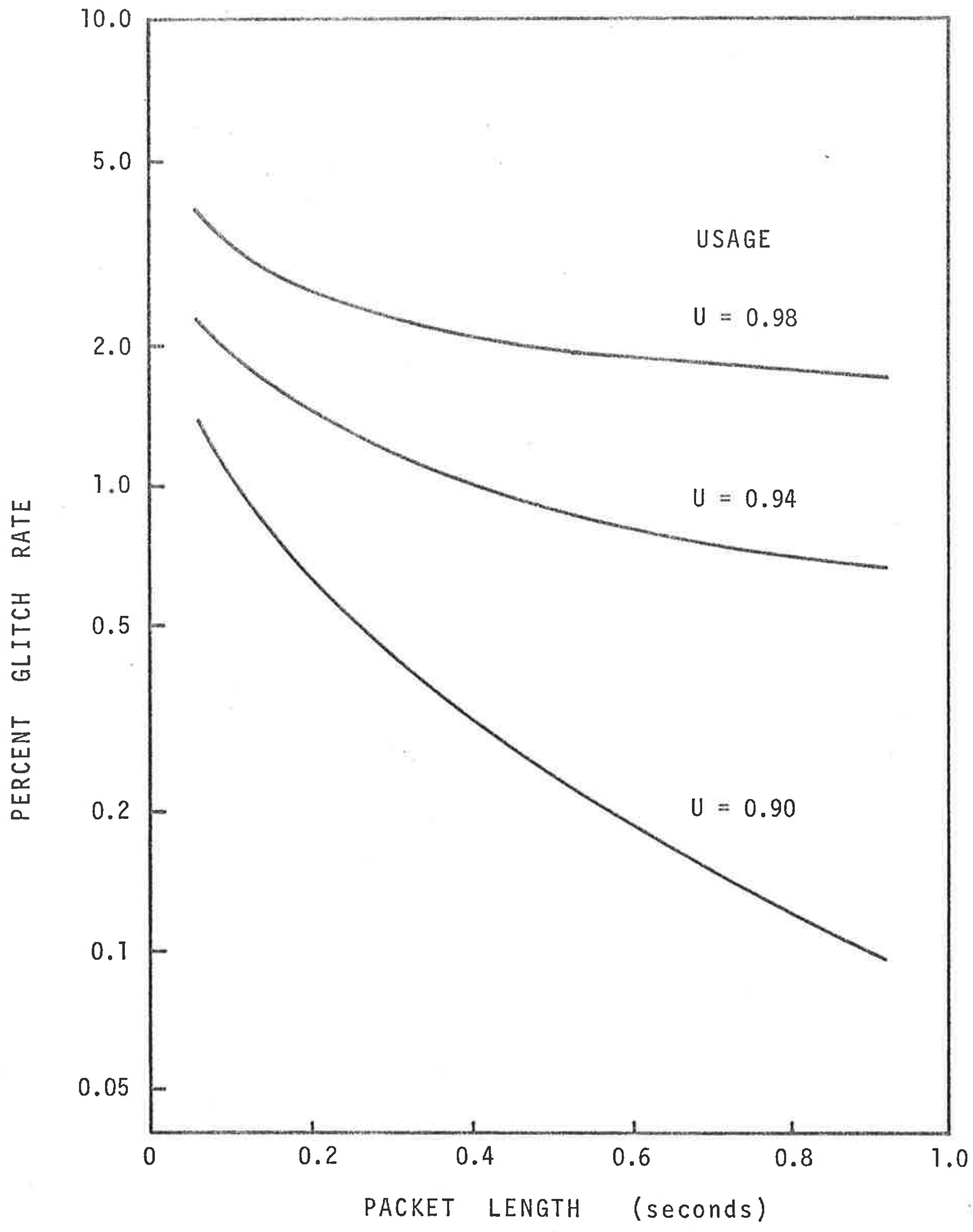


Figure 5.5 Variation in glitch rate with packet length

length the number of packets required to convey a given length talk increases. This in turn increases the correlation in the queue and hence the glitch rate.

The theoretical manner in which the glitch rate changes with the queue limit was determined in chapter 4. This is shown in figure 5.6 where the glitch rate drops dramatically with increasing queue limit. For usages below 0.95 the decrease is virtually exponential.

Unfortunately, the behaviour with a speech input is quite different. Simulation results for packet lengths between 0.2 and 1.6 seconds are shown in figure 5.7. Clearly in no case does the glitch rate decrease as quickly as in the theoretical curve at the same usage (shown dashed). The slopes of the curves at large L do however increase with the packet length. This again reflects the tendency toward more random input. The curve for T = 1.6 differs from the others because it was measured for the single subpacket case. Four subpackets were used in the remaining simulations.

Another interesting aspect of this figure is that the simulation curves actually cross the theoretical one. This is not surprising for the 4 subpacket curves since they have lower glitch rates than the single subpacket case on which the theoretical model is based. However the single subpacket, T = 1.6 curve also crosses the theoretical one and further simulations have shown that this happens at shorter packet lengths as well. The queue limit at which crossover takes place increases with the packet length.

Previously it has been assumed that a more random input will produce a lower correlation in the queue and hence a lower glitch rate.

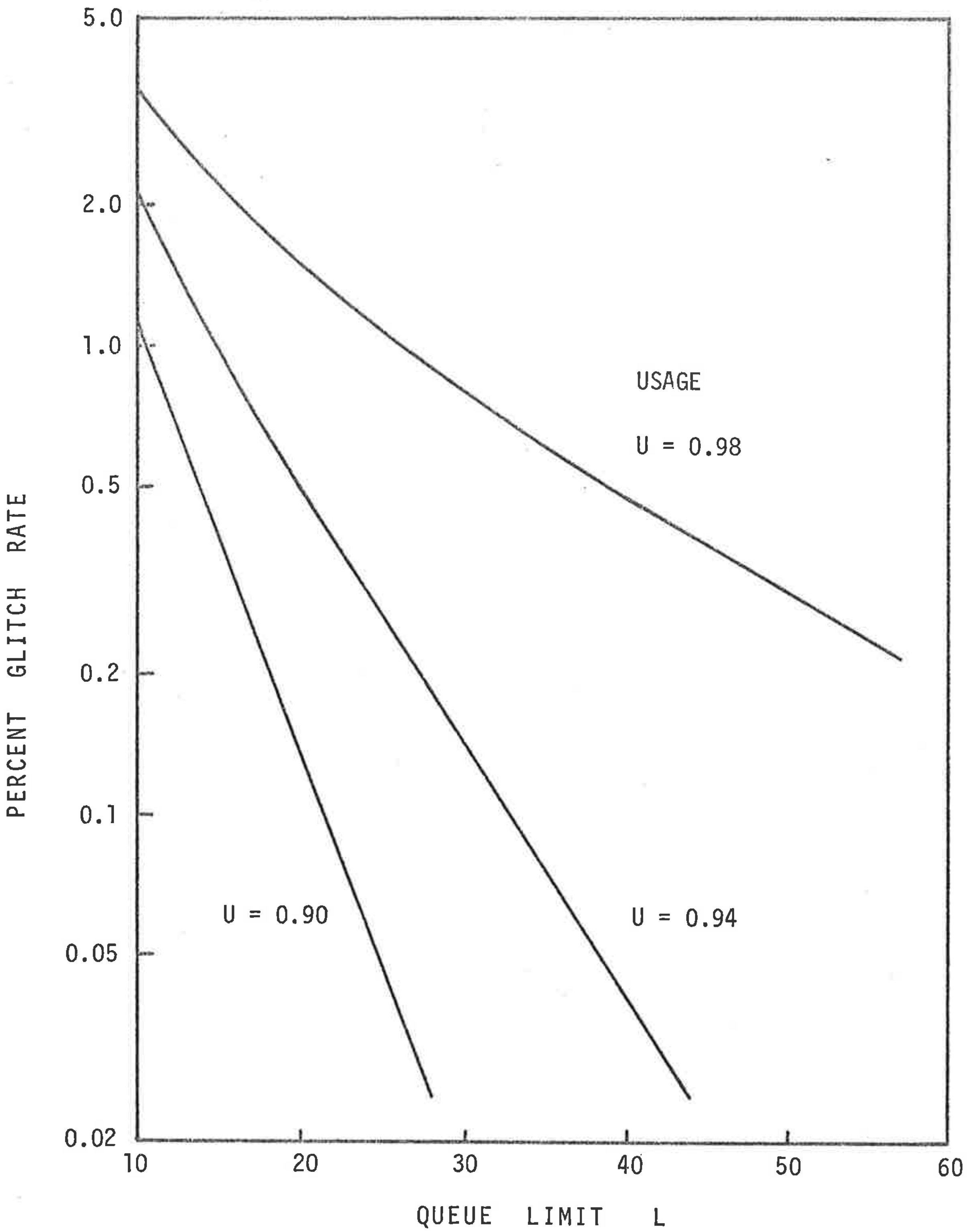


Figure 5.6 Theoretical variation in glitch rate with queue limit at different usages

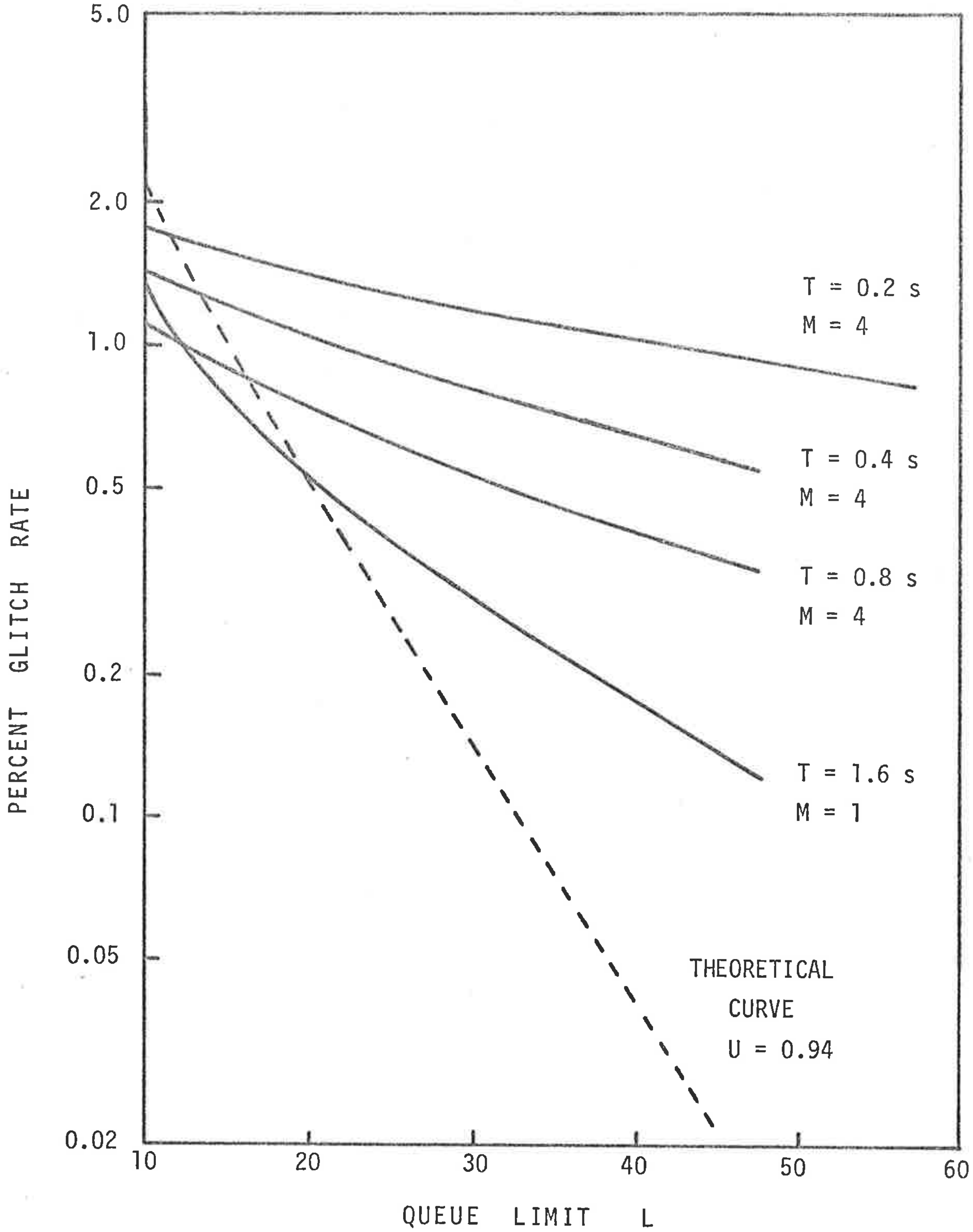


Figure 5.7 Actual glitch rate variation with L for usage near 0.94

This still holds but there is another aspect which results in glitch rates below those of truly random inputs in certain circumstances. The probability of one packet directly following another is quite high for speech though it decreases with larger packet lengths. However over a time interval less than the packet length the probability of a packet following another from the same source is zero. Thus if the queue limit is substantially less than the packet length, the input will be smoother than random because of the enforced periodicity. This effect is greater for larger packet lengths and relies upon the finite number of sources. It does not arise in the Poisson theoretical case since this assumes an infinite number of sources.

5.4 Optimization of the System Parameters

Now that the effects of the various parameters have been determined individually, it is possible to combine them and optimize the entire system. The goal of this exercise is to permit the maximum number of simultaneous conversations within a given total channel capacity and subject to a set maximum delay and glitch rate. These conditions fix the values of certain parameters and optimization involves selecting the remaining parameters to maximize the system efficiency. The fixed parameters are C, the nominal number of circuits and V, the delay in slot assignment.

Consider firstly the request channel. The capacity required by this channel is (from (3.5))

$$W_r = A\ B\ C\ \left(\frac{1}{T} + \frac{1}{20}\right) \qquad (5.7)$$

To optimize the system this must be minimized. The smallest possible value of the ALOHA factor A is 2.72. However, in practice, a larger value must be used to avoid constant overload in the request channel. A safe value of A which results in a very small overload probability is 5. This has been used throughout the simulations and will be taken as the optimum value.

The number of bits in a request packet, B, depends largely upon the synchronization requirements. A maximum value of 120 bits was suggested in chapter 3 and will be used here.

Now (5.7) suggests that the packet length should be made as large as possible. However the efficiency of packet speech interpolation is reduced by an increase in the packet length. The combined effect of these two factors is defined by the full channel interpolation gain which, for the values of A and B just set, is shown as example (b) of figure 3.10. This gain has a maximum at a certain value of packet length corresponding to the optimum ratio of request channel to speech channel capacities.

In deriving this curve a usage of 1.0 was assumed. However, as has been shown, if reasonable delay and speech quality are required the usage must be reduced somewhat. The final or system interpolation gain, I_s, is therefore given by the product of the full channel interpolation gain and the usage required to produce a given glitch rate at a fixed delay. This usage however, will vary with the packet length in a manner which will now be determined. When this is done the optimum packet length for the entire system can be found as that which maximizes the system interpolation gain.

With a fixed system capacity, C, and allowed slot assignment delay, V, the relationship between the queue limit and the packet length is given by (4.9) as

$$L = \frac{V\,C}{T} \tag{5.8}$$

The discussion in section 5.3 showed that the packet length should be increased to reduce the correlation in the speech queue and hence the glitch rate. However (5.8) indicates that another consequence of increasing the packet length is a reduction in the queue limit. The latter, by itself, results in an increase in the glitch rate. Thus there are conflicting influences on the glitch rate when the packet length is changed. The final result for any particular combination of parameter values can be determined only by simulation.

For the purpose of the simulation, and in line with previous conclusions, the values of C and V were chosen at 40 and 0.2 seconds respectively. This provides a queue limit of (from (5.8))

$$L = \frac{8}{T} \tag{5.9}$$

Simulations were performed at various packet lengths, using this relation to set the queue limit. For each combination the usage which resulted in a glitch rate of 0.25%, 0.5% and 1.0% was measured.

Because of the conflicting effects mentioned above, the usage was found to vary very little with the packet length. However there was a small peak for each glitch rate at lengths between 0.3 and 0.4 seconds. At both larger and smaller packet lengths the usage was reduced. The system interpolation gain was then calculated by multiplying these usages by the full channel interpolation gain. The resulting curves are shown in figure 5.8.

These curves are the ultimate description of the system performance. They show how the number of simultaneous calls possible

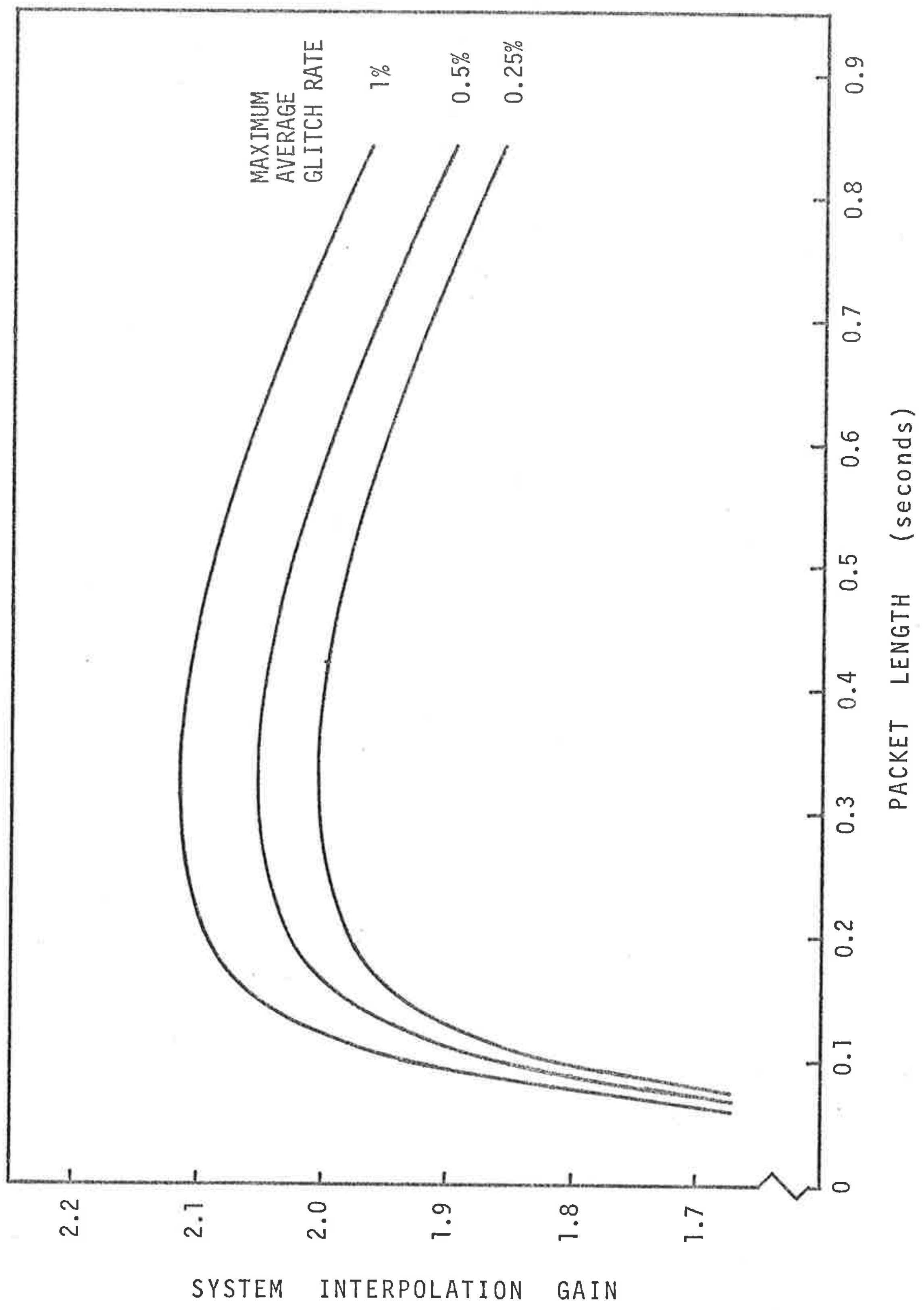


Figure 5.8 System interpolation gain with 0.2 second assignment delay

($\frac{W_t}{D} I_s$) varies with the packet length for a given glitch rate and assignment delay. There is an optimum packet length of about 0.32 seconds in all the cases shown. However, the curves are remarkably flat near their maximum point. A choice of packet length anywhere between 0.2 and 0.5 seconds results in at most a 1% reduction from the optimum efficiency. This is not surprising, given the flatness of the full channel interpolation gain curve and the conflicting effects on the glitch rate of varying the packet length.

If the delay in slot assignment is reduced to 0.1 seconds the system interpolation curves are as shown in figure 5.9. There is a drop in efficiency, in the region of interest, of less than 2% from the 0.2 second delay case. The optimum packet length is again around 0.32 seconds and the shapes of the curves are also little changed. Such similarity between systems with delays differing by a factor of two is perhaps surprising at first glance but it arises from the rapid reduction in glitch rate with decreasing usage. The three interpolation gain curves, I, I_f and I_s are shown together for comparison in figure 5.10. The I_s curve represents the V = 0.1 seconds, GR = 0.5% case.

The only system variable which has not yet been changed is the nominal number of circuits, C. This is set by the capacity of the speech radio channel and the data rate of digitized speech. If a speech channel capacity of 1 Mbit/s cannot be achieved, or a data rate of greater than 25 kbit/s must be used for speech, then C will be less than the value of 40 used previously. This must result in a greater glitch rate even if no other parameters are allowed to vary. Fewer calls can be handled and the correlation in the speech queue rises accordingly. Figure 5.11 shows how the glitch rate varies with C for several usages.

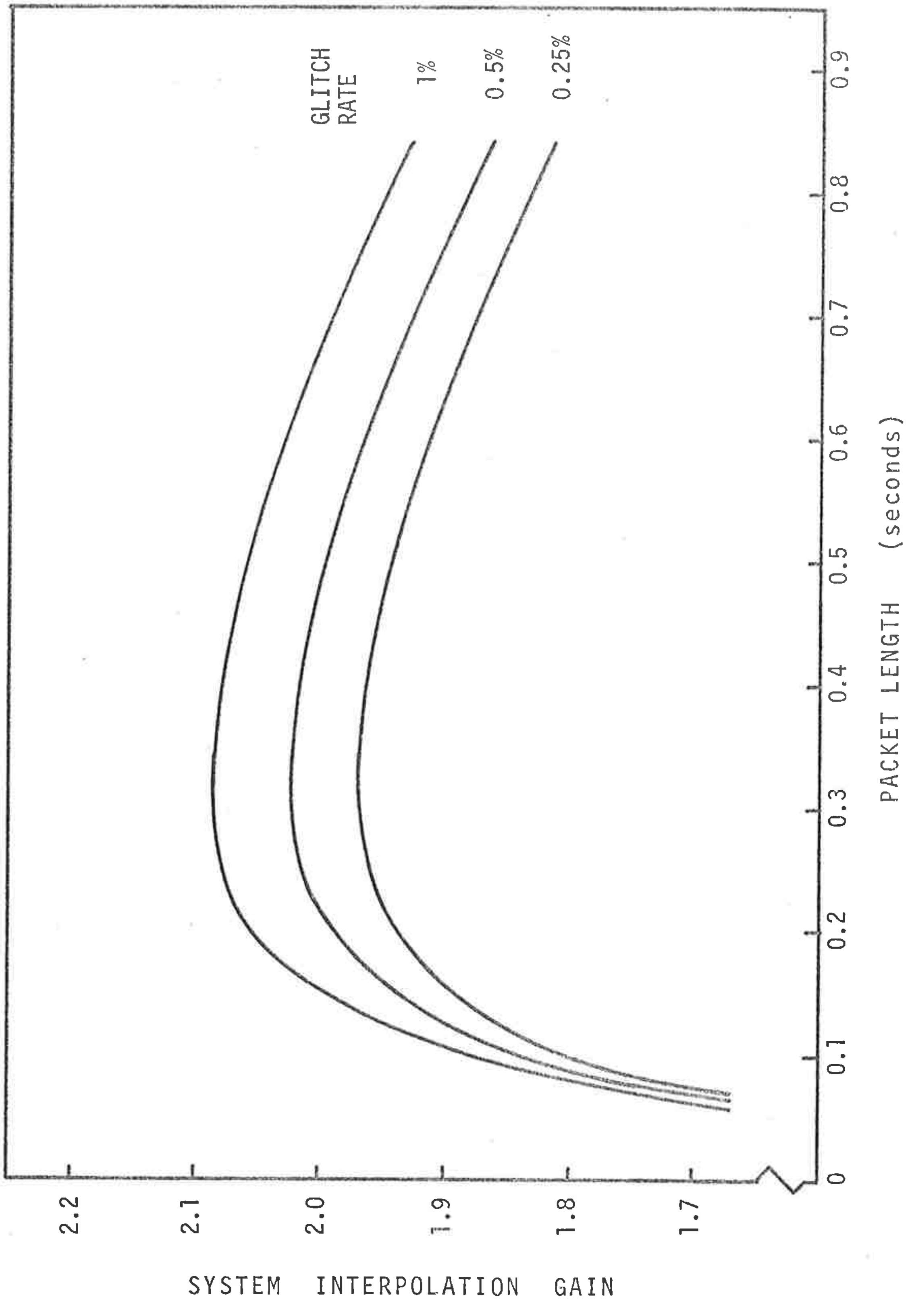


Figure 5.9 System interpolation gain with 0.1 second assignment delay

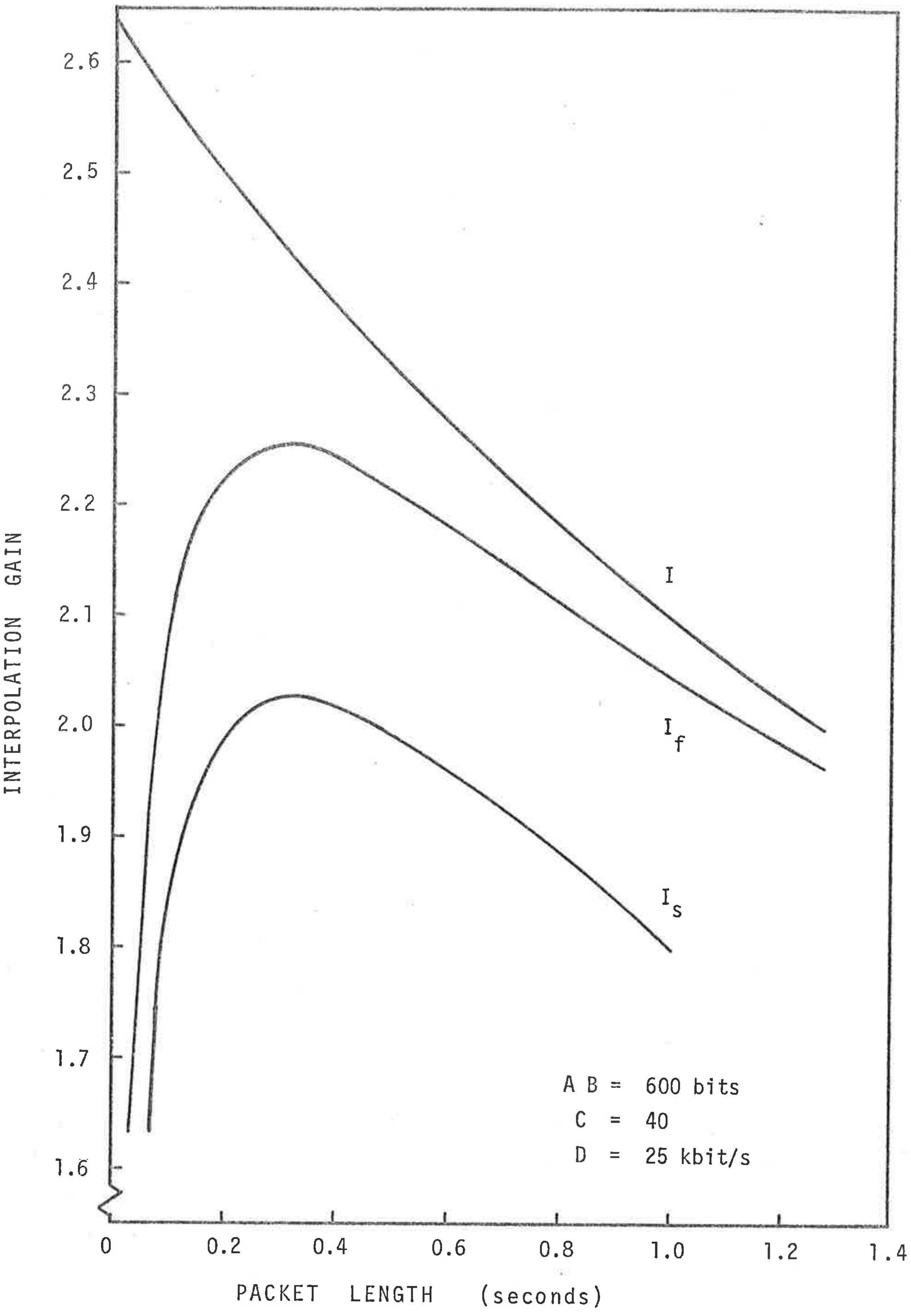


Figure 5.10 Comparison of interpolation gains

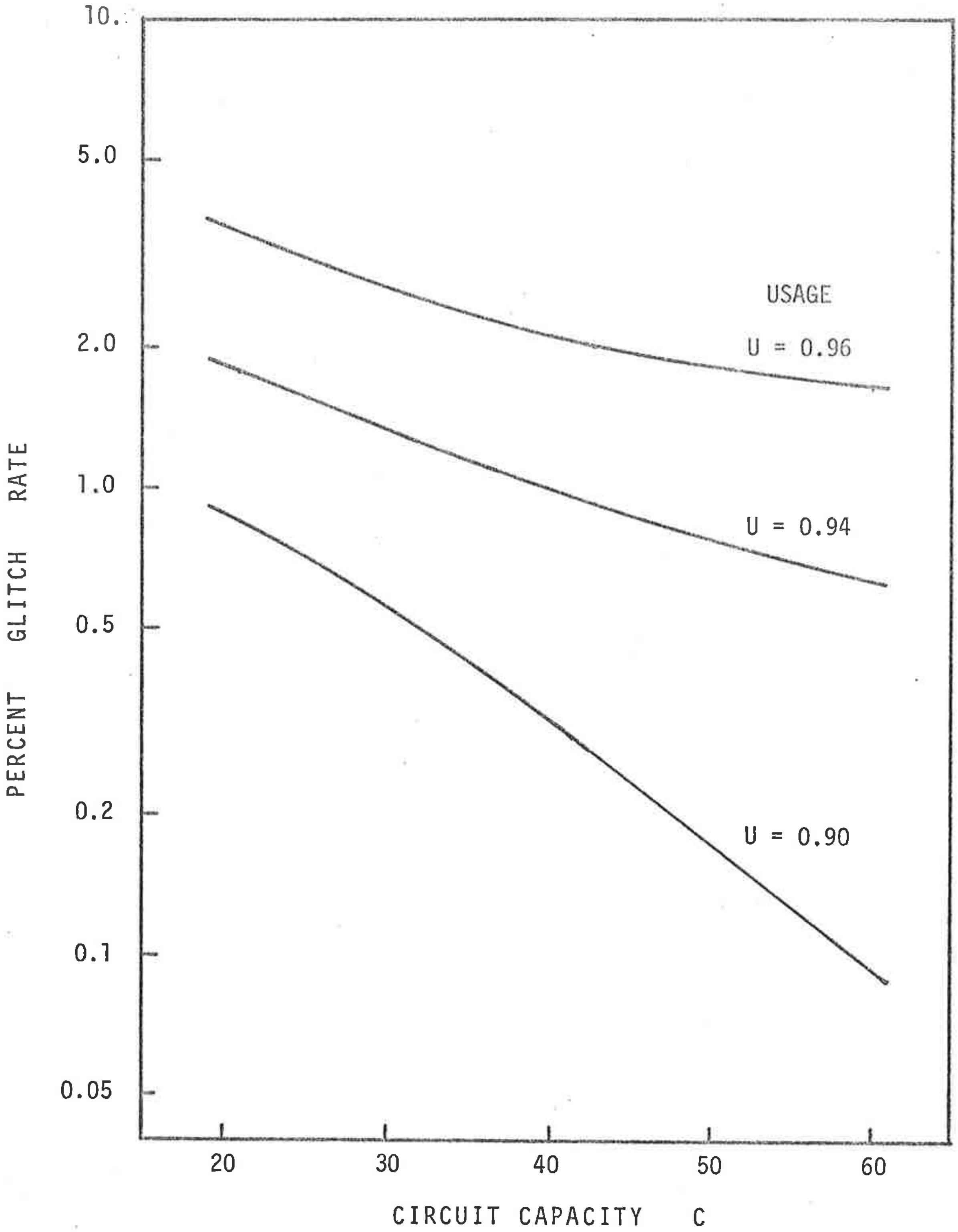


Figure 5.11 Variation in glitch rate with system capacity

A reduction in capacity has the same sort of effect as a decrease in the slot assignment delay time. Again the natural compensating factors result in little change in the system operating characteristics and optimum packet length. Simulations of systems with an assignment delay of 0.1 seconds, show that reducing C from 40 to 20 results in only an 8% drop in the system interpolation gain. Of course in practice, C should be kept as large as possible to ensure good averaging of the speech input.

It has been found that the peak of the system interpolation gain curve is quite broad. Thus although a packet length exists which maximizes this gain, a range of packet lengths can be selected with little overall loss in efficiency. It is therefore possible to look toward other characteristics to determine how best to optimize the system. The most important of these characteristics is the subjective effect of glitches which will now be considered.

5.5 Distribution of Glitches

The extent to which a given glitch rate is subjectively important depends upon the manner in which glitches occur. An advantage of the simulation programme is the ease with which realistic statistics on glitches can be obtained. The first of these to be measured was the probability of various numbers of glitches within an interval of 200 slots. If the glitches occurred randomly, this quantity should have a Poisson distribution with an average of 200 times the glitch rate. Figure 5.12 shows the probability density function measured, together with that of a Poisson source.

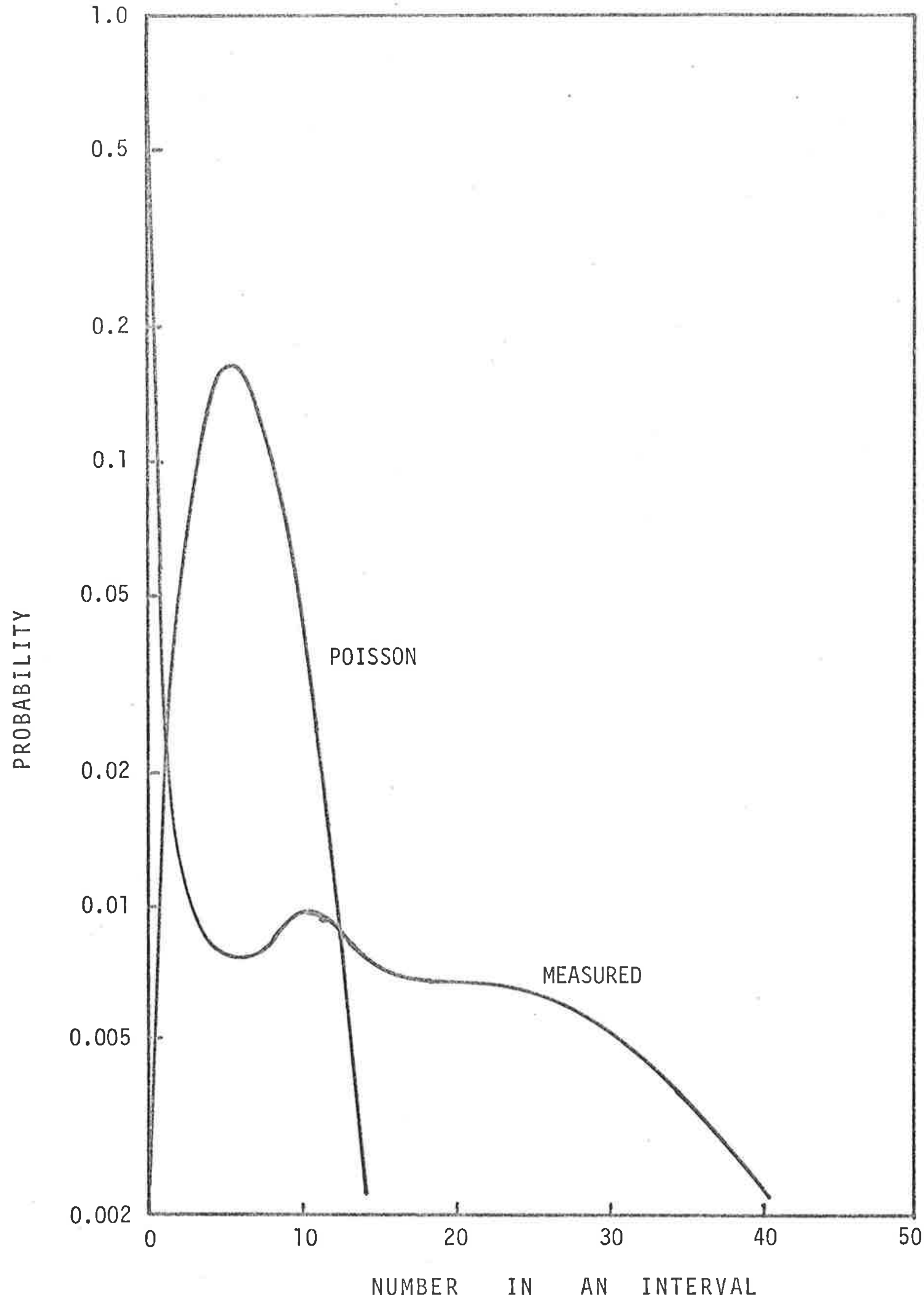


Figure 5.12 Probability of various numbers of glitches in an interval

In the Poisson case there are few intervals without glitches, but in reality, the majority contain none. Also, the measured probabilities of many glitches in an interval are far greater than the Poisson values. Thus it appears that glitches occur in bursts with relatively long glitch free spaces between the bursts.

Now the glitches in a burst could have come from any number of conversations, but the subjective effect of glitches will depend upon the pattern for an individual user. It is necessary therefore to determine the space between glitches in a single conversation and whether adjacent subpackets are often glitched. To this end the programme was modified to produce statistics for each of the simultaneous conversations. It was found that the number of conversations contributing to a glitch burst was fairly small and that successive subpackets were glitched quite often.

The amount of speech lost in one glitch is simply the packet length divided by the number of subpackets, i.e. $\frac{T}{M}$ seconds. The actual length of a sequence of glitches, termed a glitch interval, is this length multiplied by the number of glitches in a sequence. Glitch intervals were measured for various numbers of subpackets with a packet length of 0.4 seconds, and the results are shown in figure 5.13. The first thing to notice is that for $M \geqslant 4$ the probability of a sequence containing more than one glitch is greater than 0.5. Even for $M = 1$ the probability is greater than 0.3. In every case the average glitch interval length, as shown in table 5.4 is considerably greater than a single subpacket length.

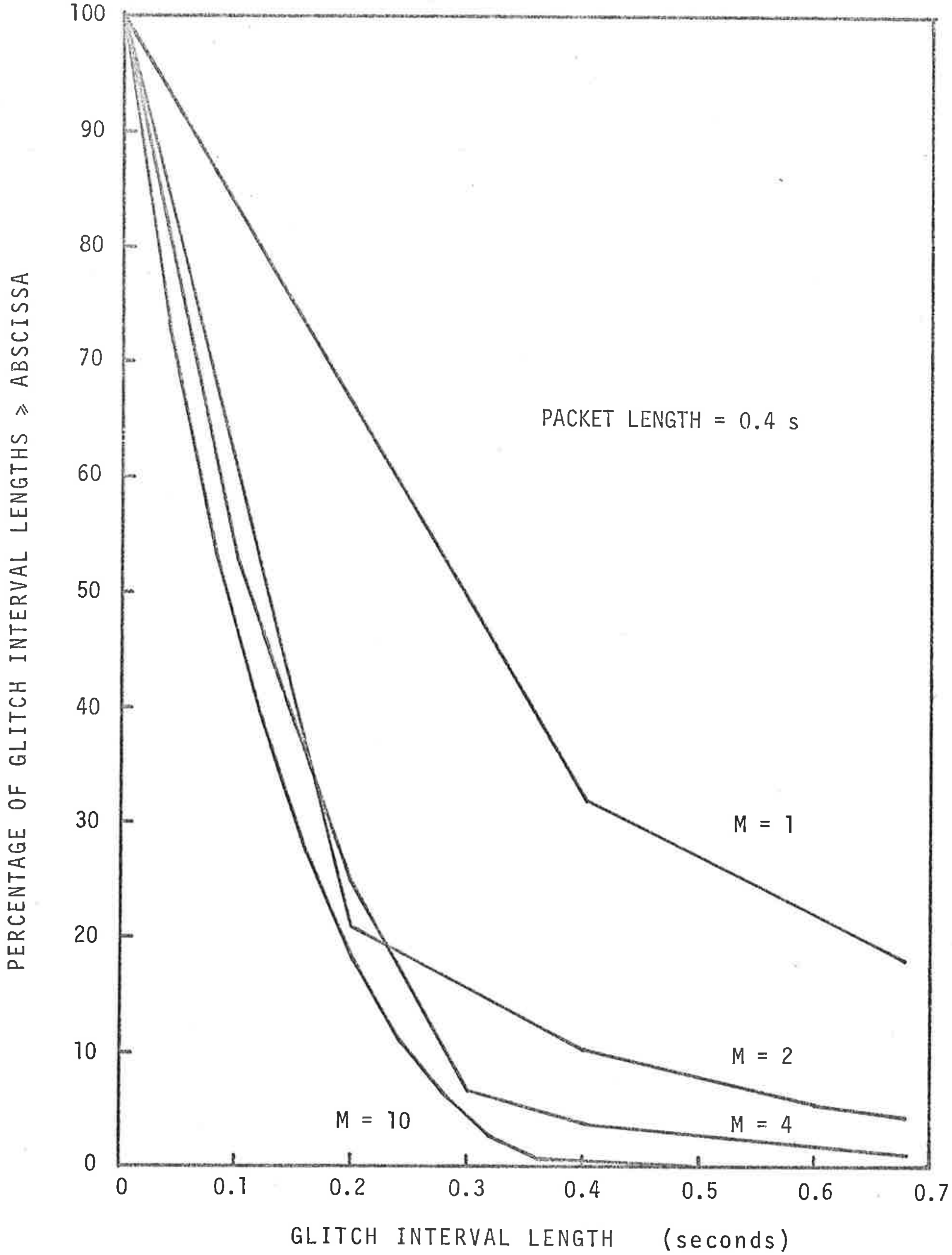


Figure 5.13 Distribution of glitch interval lengths with various numbers of subpackets

Number of Subpackets, M	1	2	4	7	10
Average glitch length (s)	0.607	0.286	0.198	0.172	0.134

Note C = 40, GR = 3%, L = 20, T = 0.4 seconds

Table 5.4 Average length of a glitch interval

Although the average glitch interval length diminishes consistently with increasing M, figure 5.13 shows that the distributions for M ⩾ 2 are very close. This implies that it is the time of a glitch interval, rather than the number of subpackets involved, which is relevant to the system. The spread in average length simply reflects the fact that the unit quantity of glitch is coarser with smaller M.

Figure 5.14 shows how the length of a glitch interval varies with packet length when four subpackets are used. In all cases less than 10% of the glitch intervals exceed the packet length, and there is a sudden change in the distributions at a length of three subpackets. The same change is noticable in figure 5.13, always at a length of M - 1 subpackets. Thus the probability of an entire packet being glitched appears to be somewhat depressed.

It would not be surprising to find a discontinuity in the distribution for lengths greater than the packet length because the probability of a following packet is less than one. However, for some reason, the change occurs at one subpacket less than this. In an effort to understand this the probabilities of each subpacket in a packet being glitched were determined. The results appear in figure 5.15 plotted against the subpacket number, where the first

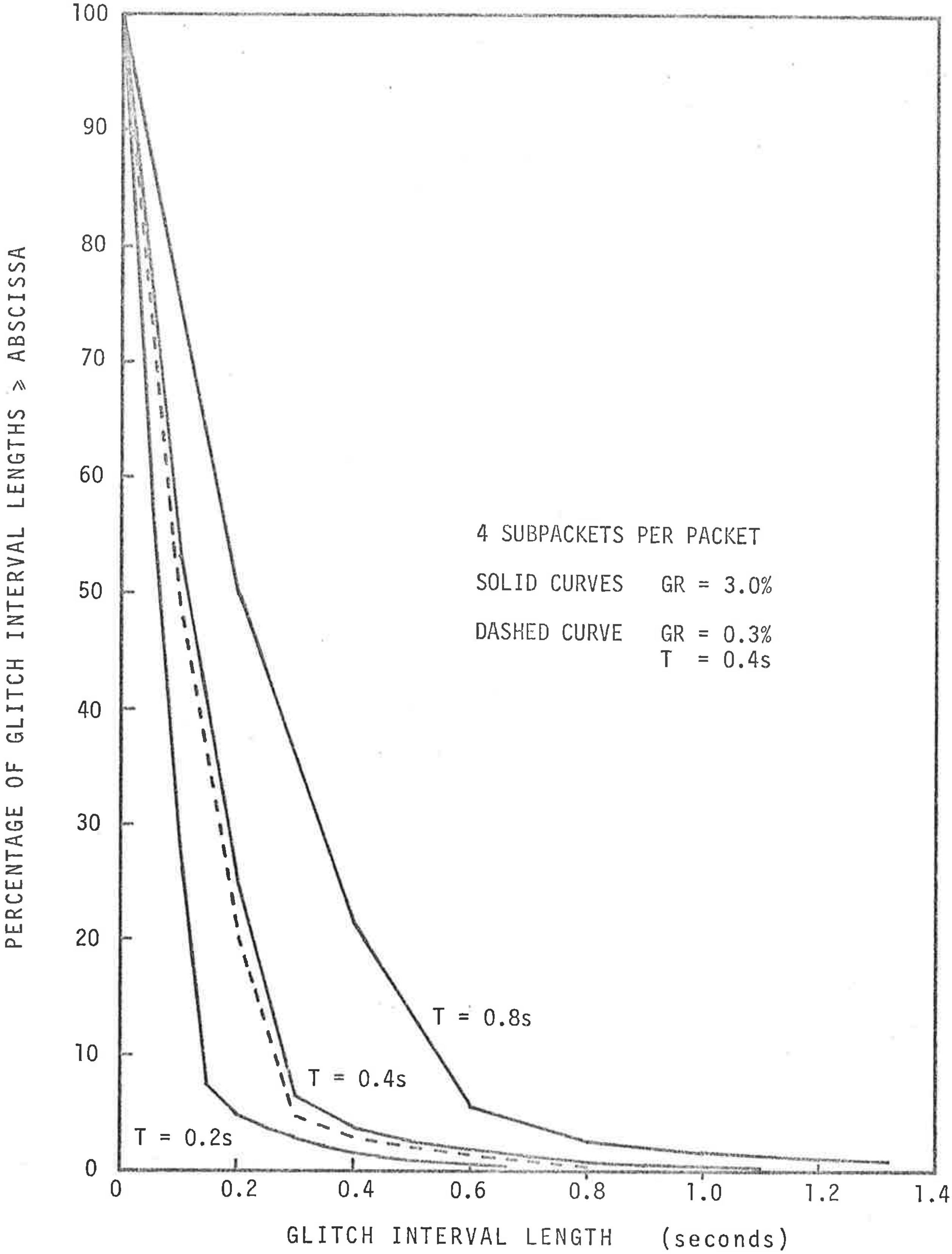


Figure 5.14 Distribution of glitch interval lengths for various packet lengths

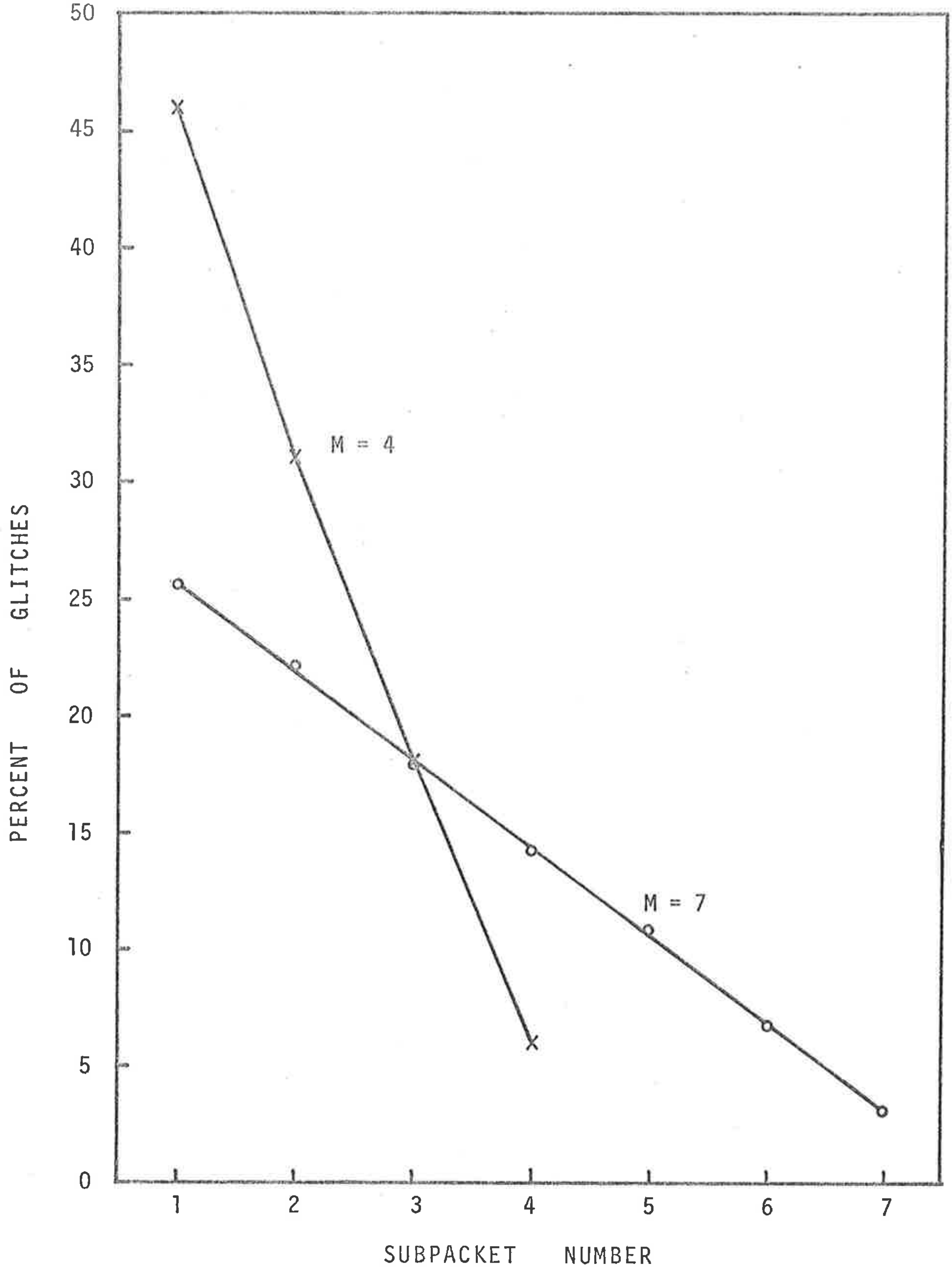


Figure 5.15 Probability of various subpackets being glitched

subpacket in a packet is numbered 1.

In both cases shown, the glitch probability decreases in a remarkably linear fashion from the first to the last subpacket. Thus the probability of glitch intervals exceeding the packet length is low because it requires the last subpacket to be glitched. The fact that early subpackets form the majority of glitches however will influence the subjective effects and hence the system design. Thus it is important to understand why this occurs and whether it can be changed if necessary.

The probability of particular subpackets being glitched was found to be independent of both the glitch rate and the packet length. Therefore the probabilities must be determined entirely by the nature of the queue and the method of adding subpackets to it. Consider the queue when it is near its limit and a packet is added. The first slot in which the first subpacket can be placed is almost L' slots from the start of the queue because of the assumed state. Thus the initial part of the queue is not altered before transmission, except for insertions of subpackets delayed by multiple request collisions. This section of the queue therefore contains an equal mixture of subpackets from all parts of a packet.

The next section of the queue is the active part where the early subpackets are added and where glitches are occurring. The last section contains only the final subpackets from recent requests and has a small delay and consequently experiences no glitches. In the glitch region therefore there is an excess of early subpackets and they are glitched correspondingly more often.

Action which could be taken to alter the distribution would involve biasing the queue to position early subpackets ahead of late subpackets with the same priority. This would result in a shift in the glitch distribution toward later subpackets and is advantageous because these are less likely than the early subpackets to contain speech. However, the subjective improvement from this change is limited by the significant likelihood of one packet being followed by another.

The space between glitch intervals was also determined for a number of systems. The distributions, shown in figure 5.16, indicate an extreme non-uniformity in the space lengths. For instance with a packet length of 0.4 seconds, 4 subpackets per packet and a glitch rate of 3%, some 50% of all spaces are less than 2 seconds long. At the same time the largest 10% exceed 40 seconds. The most common space is one subpacket length. This indicates that even for individual calls the glitches tend to occur in bursts. Each burst consists of a number of glitch intervals separated by only a few subpacket lengths and again the spaces between bursts are relatively long.

The importance of the packet length is again emphasised by its obvious effect on the distributions at small lengths. Spaces of length equal to or greater than the packet length have similar probabilities but are much less likely than spaces equal to a fraction of the packet length. This means that the glitches in a burst will consist of the early subpackets in adjacent packets. For instance, a very likely occurrence in a system of four subpackets is the glitching of the first three subpackets in one packet and the first or more in the subsequent packet. If the queue biasing is changed as

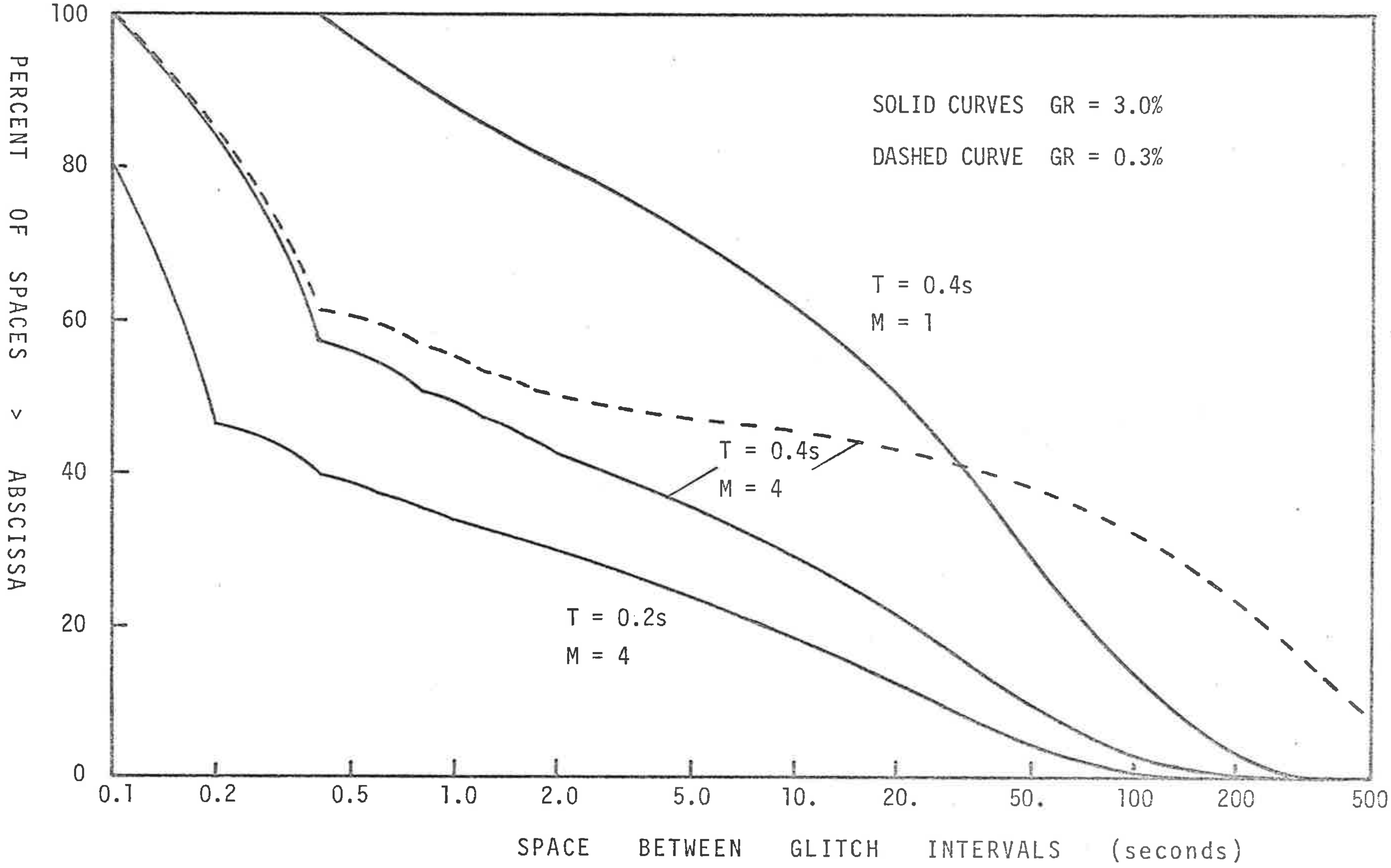


Figure 5.16 Distribution of spaces between glitch intervals

suggested earlier these small breaks within a burst may well be eliminated.

Figure 5.16 also shows that the actual length of a glitch is a crucial factor. As this length is made shorter either by decreasing the packet length or by increasing the number of subpackets there is a shift toward smaller spaces. This results in a greater number of bursts and more glitch intervals within a burst. Thus the queue is adjusted by glitches more frequently but in finer steps.

Space length distribution at large lengths is mainly determined by the glitch rate. The dashed curves in figures 5.14 and 5.16 show the 0.4 second packet length, 4 subpacket case as above but with a greatly reduced glitch rate. Large spaces are much more likely here but the space distribution at short lengths, and the probability of adjacent subpackets glitching, are virtually unchanged. Thus the glitches still occur in bursts of the sort described above but there are much larger spaces between the bursts.

In summary, it has been found that glitches occur in bursts, with the nature of the burst being determined mainly by the packet length, and the space between bursts, mainly by the glitch rate. This situation is sufficiently different from that occurring in the TASI undersea cable system to render the subjective effects of glitches completely different. To obtain more information on this aspect, listening tests were done on speech with glitches. This is discussed in the next section.

5.6 Subjective Speech Trials

The effects that glitches cause in speech are quite different in nature to the results of more conventional degradations. For the majority of the time an interpolation system provides the best speech quality possible, while at times there is no signal at all. The nearest phenomenon in other telephone networks is impulsive noise, but the complex distribution of glitches and their possibly long length make the two degradations quite dissimilar. There is therefore very little available information relevant to the subjective effect of glitches. Hence it was necessary to perform speech tests.

The aim of these tests was to determine what glitch rate people found to be acceptable and how this depended upon the nature of the glitches. The very complexity of the glitch pattern however provided significant difficulties. The only method of determining the true subjective effect would be to use speech subjected to a realistic pattern of glitches, perhaps as provided by a simulation. This was not done because of the difficulty in arranging such recordings and because of the time required in testing to ensure that the true long term effects were measured. Instead, tests were done on recorded speech with random glitches of constant length and constant average rate. The values of the length and rate were varied independently over a wide range.

A pseudo-random shift register of 11 stages, shown in figure 5.17, was used to generate these glitches. The glitch length was set by the clock period and a number of output bits were ANDed together to form the glitch signal. The number of bits chosen determined the rate according to table 5.5.

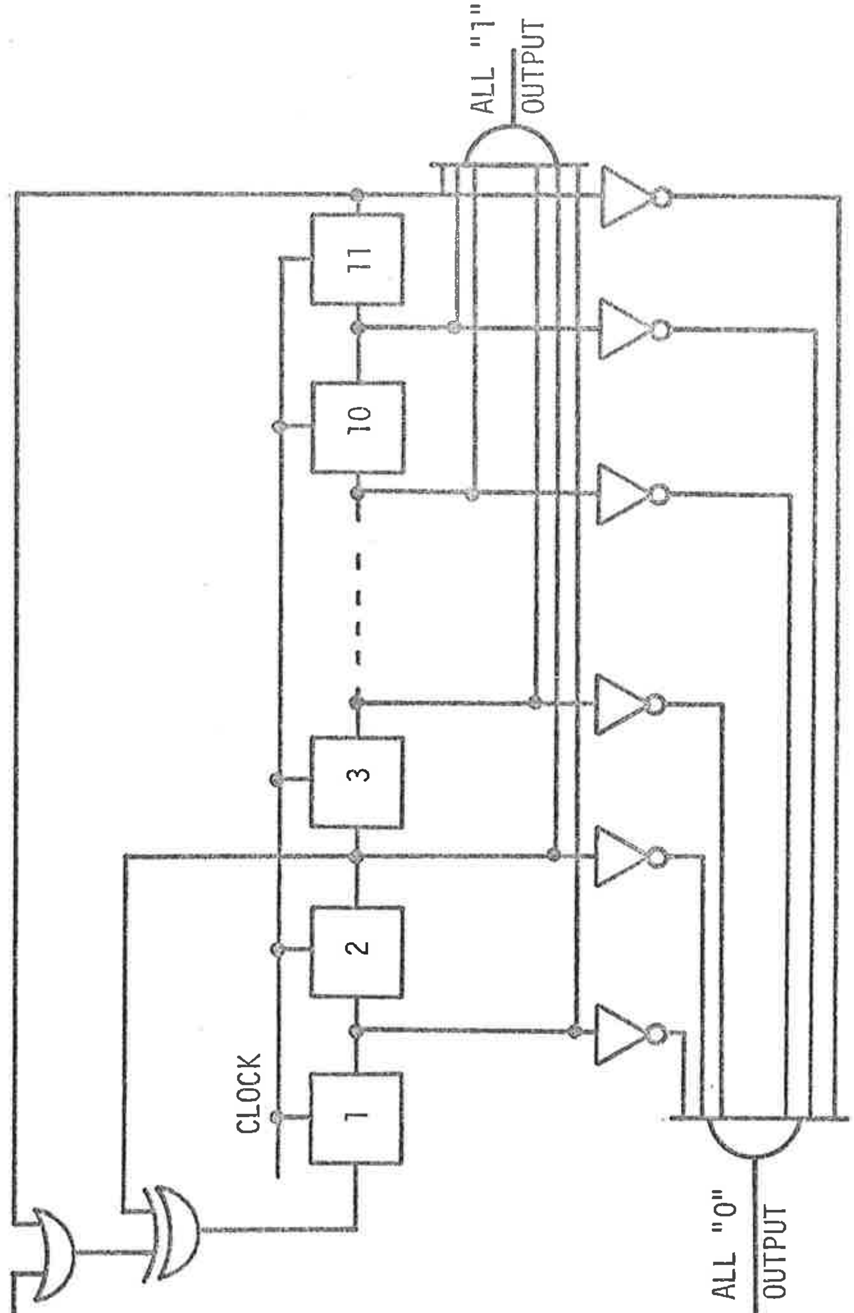


Figure 5.17 Block diagram of random glitch producer

Number of Bits	11	10	9	8	7	6	5	4
Glitch Rate %	.049	.098	.195	.391	.781	1.56	3.13	6.25

Table 5.5 Glitch rate for each number of ANDed bits

Subjects listened on a telephone to speech sent through a central console [94]. This console possessed digital attenuators which were used to reduce the speech volume by 32 dB during glitches. It also possessed a white noise source which could be substituted for the speech instead of leaving the glitch intervals completely silent.

A minicomputer controlled the system during the tests. It set the glitch length and rate on the glitch producer and fixed the length of speech offered. The glitch length was set at one of six values between 6.25 and 200 ms, with each step being a factor of two larger than the previous one. Similarly, there were six possible glitch rates between 0.1% and 3.1% with again a factor of two between adjacent steps. Each test was 40 seconds long. Such a length was necessary to obtain a reasonable probability of having several glitches in the test in all cases.

The speech used consisted of recorded news broadcasts. This type of speech was selected in an effort to closely parallel normal conversation. The alternative of some of random speech would have been more applicable if word intelligibility had been at issue. Preliminary tests had however, already shown that intelligibility was very high. This is not surprising as even for very long glitches the percentage of words lost completely will be less than the glitch rate. On the other hand, at small lengths the intelligibility is even better, as such glitches rarely destroy entire words. The interpolation system will obviously be used in a conversation environment where fill in from context is available and therefore real speech is the most appropriate.

After listening to each trial the subject assessed the quality according to an opinion scale of 7 grades. To ensure that the full scale was used, a best and a worst case were presented before the tests commenced. The scores for each combination of glitch rate and length were averaged over all the subjects and curves of equal subjective quality were drawn on the glitch rate-frequency plane. Figure 5.18 shows how these lines split the plane into regions of good, above average, below average and poor qualities.

It is apparent that the perceived effect of any glitch rate is worst at a glitch length of around 25 ms. This can be explained in terms of the frequency of glitches, which is inversely proportional to their length. Glitches with long lengths occur very infrequently and this reduces their subjective importance. At very short lengths the frequency is high but the damage done by each glitch is negligible and again the effects are not as significant. Thus a medium length is, subjectively, the worst choice.

In the above tests white noise was used, instead of silence, during the glitches. This was done because the preliminary trials showed that noise made the glitches less noticable. Unfortuantely the proper tests failed to confirm or deny this observation. In the case of small glitches the speech appears to be broken up without the noise and yet at longer glitch lengths the noise has a degrading quality of its own. It may well be that different levels of noise

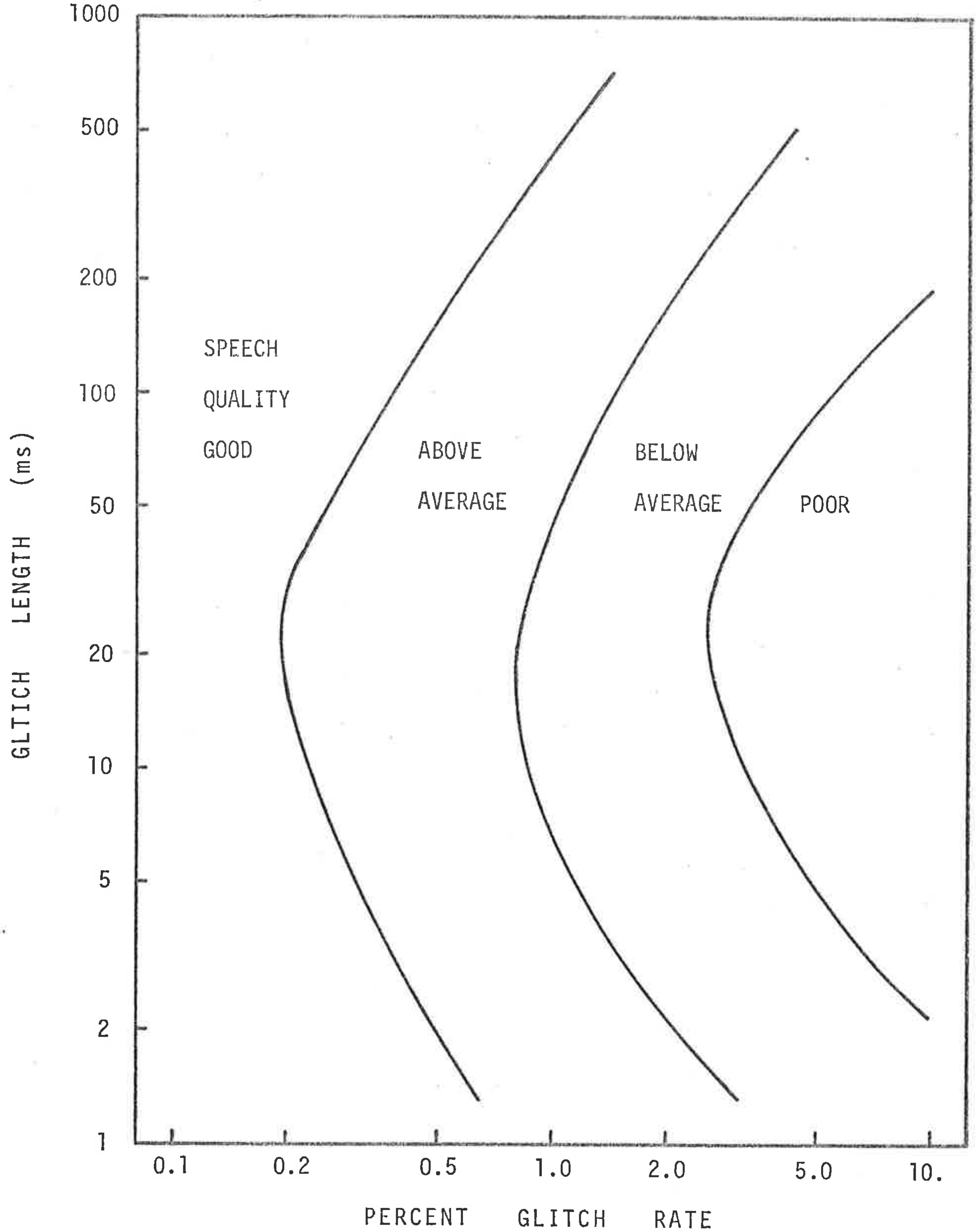


Figure 5.18 Results of subjective speech tests

are appropriate in different areas of the glitch length-rate plane.

There are several points to bear in mind when considering the results of figure 5.18. Firstly, the individual tests were judged very much in comparison with each other. Thus, the results give only a weak indication of the absolute acceptability of any point. It would be better to compare glitches with other types of distortion and in particular to determine the signal to noise ratio of a channel without glitches, judged to have equivalent quality. The extreme difference between the two types of distortion could however cause problems.

In the tests performed, the nature of the distortion was made apparent at the beginning, with the worst case demonstration using a 6% glitch rate and a 25 ms glitch length. Thus, the subjects were in a sense waiting for glitches to occur and judging the quality according to their frequency and length, rather than assessing the overall situation. This concentration on the degradation rather than the speech might be removed if a two way conversation were used instead of recorded speech.

A constant glitch length was used in these tests for simplicity. However, it was shown that glitches in reality occur in sequences of varying number. This of course is an advantage if the glitch length exceeds 25 ms, since larger glitches are less objectionable.

Also, in the tests, glitches were randomly distributed in time which is somewhat different to the grouping that occurs in practice. Again however, this means that the tests are rather conservative since a burst of glitches may well sound like one very large glitch. Lastly, it should be noted that since glitches do not occur during silence intervals their frequency in an actual two way conversation would be half that observed here.

The glitch rate of 0.5%, which has to date been the required goal, falls at all glitch lengths within the above average region in figure 5.18. In view of this and the comments above it appears reasonable to leave the goal at 0.5%. It should also be remembered that this figure is to apply when the system is running at its design load. At periods of lighter traffic the glitch rate will drop as shown in figure 5.1. When the rate is below 0.2% glitches can for all practical purposes be ignored.

5.7 Conclusions

In this chapter the entire mobile telephone system has been investigated through a computer simulation. The programme is essentially that described in chapter 4 except that requests are generated by random conversation rather than according to a Poisson process. Also, the effects of the request channel and various delays are included. The glitch rate is higher than that with a Poisson input due almost entirely to the correlated nature of the input from speech.

The possible effects of request collisions were examined theoretically and by simulation. Both confirmed that with the parameters chosen earlier the request process adds negligibly to the glitch rate.

An investigation of subpackets showed that their use reduces the glitch rate, but by less than with a Poisson input. Also, there is no discernable optimum number of subpackets and hence any number between 2 and 10 is equally as effective. The final choice can be made simply on the basis of reducing the recording delay to a reasonable figure.

The glitch rate decreases when either the packet length or the queue limit are increased. In the latter case the reduction is not as great as theoretically predicted due to the increasing correlation within the queue, as its maximum length is increased. This again results from the nature of the speech input.

Since the queue limit is inversely proportional to the packet length, the last two effects tend to cancel. To optimize the entire system the usage producing various glitch rates has to be determined as a function of the packet length for a fixed slot assignment delay. This usage, when multiplied by the full channel interpolation gain at the same packet length, gives the system interpolation gain. This is the gain, over the dedicated circuit case, in the number of simultaneous calls possible at a given rate and slot assignment delay. It incorporates the efficiency of packet formation and the capacity required by the request channel. The system interpolation gain is the final figure of merit for the entire system.

An optimum packet length exists which maximizes the system interpolation gain. In fact, because of the virtual cancellation of glitch effects in the packet length region near the peak, the optimum value is almost unchanged from that of the full channel interpolation gain curve. The actual gain that can be achieved at this point depends primarily upon the average activity level of the speech.

The peak of the system interpolation gain curve was found to be fairly broad. Thus, any packet length between 0.2 and 0.5 seconds can be used with little loss in efficiency. This results from the numerous conflicting effects of a variation in packet length and shows how the system accomodates and even compensates for any changes in parameters. Another example of this is the drop in efficiency of merely 2% when the slot assignment delay is reduced from 0.2 to 0.1 seconds.

Because of the design freedom that the above observations imply, it is possible to base the choice of a packet length on secondary factors. The most significant of these, the subjective effect of glitches, was therefore investigated. It was found that glitches often occur in sequences, even within an individual conversation. These sequences are further grouped into bursts, with spaces of a few sub-packet lengths between the sequences in a burst, and possibly many tens of seconds between bursts.

Rather elementary speech tests showed that the perceived speech quality at any glitch rate varies significantly with the glitch length. In particular, a length of 25 ms results in the worst effects. A glitch rate of 0.5% was judged at all glitch lengths to be of above average quality, and a rate of 0.2% was considered to be good. Since the glitches actually occur in bursts covering several times the glitch length, the quality rating given by figure 5.18 at any length is a conservative estimate. Thus, this investigation shows that a glitch rate of 0.5% is an appropriate figure for a fully loaded system and that the glitch length should be kept above 25 ms.

From the information now available a complete mobile telephone system can be designed. This is done in the next chapter.

6. A COMPARISON OF MOBILE TELEPHONE SCHEMES

6.1 The Complete Packet Interpolation System

A packet speech interpolation, mobile telephone system, utilizes two digital radio transmission channels in each direction between the mobile telephones and the receiving stations. Space in the main or speech channel is allocated by a central controller to any telephone which transmits a request on the second or information channel. This system has been thoroughly analysed and various tradeoffs in system performance illuminated. It is now possible to present a design which, given two fundamental assumptions, is an optimum one.

The first assumption is that a digital data rate of 1 Mbit/s is achievable in the speech channel. The second is that speech may be digitally encoded, at an appropriate standard for mobile telephone services at a bit rate of 25 kbit/s. These two combined mean that the nominal number of voice circuits within the speech channel is 40. The consequences of changing this value are studied later.

After consideration of all aspects of the system a packet length of 0.32 seconds was found to be near the optimum for a wide range of conditions. This will then be accepted as the design packet length. Transmission in 8 subpackets per packet should be used to reduce the recording delay to 40 ms. With these values each subpacket will contain 1000 bits and will take 1 ms to transmit.

A further reduction in the recording delay by increasing the number of subpackets would not be beneficial for three reasons. Firstly, the subjective effects of glitches were found to become more significant with lower subpacket length in this range. Secondly, the complexity of the system and the amount of control required rises with the number of subpackets, while the transmission efficiency falls. And finally, diminishing the recording delay too much will increase the effect of collisions in the request channel.

The important parameters of the request channel are the ALOHA factor, A, and the number of request packet bits, B. After close study it was decided to adopt rather conservative values of 5 for A and 120 bits for B. Subsequent simulations proved that these choices were appropriate. The resulting request channel capacity required is 76.2 kbit/s. In practice, a value of 80 kbit/s would be used. This effectively increases A slightly and therefore reduces the request collision rate and the probability of overload. It also results in a request slot time of 1.5 ms which will very conveniently establish a time base for speech transmissions. The reduction in system efficiency from this change is negligible.

The return information or acknowledgement channel will also have a capacity of 80 kbit/s. In section 3.6 it is shown that with A = 5 (and equal capacity for the two channels) no more than two slot assignments can be transmitted per packet under normal circumstances. It is possible to increase this number by amalgamating the two information channels or by incorporating addresses within the return speech packets. However, there is no need for this as an effective number of subpackets of 2 is adequate to achieve virtually the maximum glitch rate advantage. Very small delays are guaranteed in the acknowledgement channel by the low overall usage of 88% and the regular nature of slot assignments, which constitute three quarters of the traffic.

The most significant delay in the entire system, that in slot assignment, can be set at 0.1 seconds. The total one-way delay including the contributions from recording, transmission and coding should then not exceed 0.15 seconds. In the reverse path between the stations and the mobile telephones the same delays exist and the same total is found. There is of course no request channel in the reverse direction as all the packets are generated at the control centre. The round trip delay with this system is therefore 0.3 seconds. This is equivalent to the delay on a land based long distance telephone call of over 1000 km. It is half the round trip delay on a synchronous satellite circuit.

If a mobile to mobile telephone connection is made, the delay need be no larger than on a mobile to land based call since any packet has to be recorded only once, and slot assignment can take place in both paths simultaneously. It is also possible to transmit mobile calls over long distances and even via satellite links. This simply requires that any packets in such a call be given priority in the slot assignment queue so that the delay here is made negligible. Then the additional round trip delay added by the mobile connection is only 0.1 seconds. A slightly higher glitch rate results from this procedure but as long as the proportion of such calls is small there will be no problem.

With a glitch rate of 0.5% in this arrangement the system interpolation gain is 2.02 (see figure 5.9). Thus a total of 87 simultaneous conversations can be handled within the total capacity of 2.16 Mbit/s. Assuming a busy hour traffic per subscriber of 0.02 Erlangs, the number of subscribers who can be serviced with a 0.05 probability of blocking is 4100 [95].

It should be noted that there is no need to have a fixed delay before reproducing the speech. This was originally proposed to prevent sudden and possible large delay changes. However the simulations show that in all cases the change in delay is quite gentle, with a change from low to high values or vice versa requiring at least four seconds. Thus, the change from one subpacket to the next is only of the order of two milliseconds. Having no fixed delay allows the smallest possible delay at all times and also avoids the need for special storage to ensure that the full delay is used.

In times other than the busy hour, when the number of simulataneous calls is less than 87, the average glitch rate and delay will fall. Thus for the majority of the time the total round trip delay will be near 0.1 seconds and the glitch rate will be negligible.

An aspect of this system which is worthy of another mention is the possibility of packet repetition whenever a slot assignment is lost or a subpacket is destroyed. A new slot can be assigned by the controller and transmitted to the mobile telephone, resulting in a repetition delay of only 10 to 20 ms. Thus as long as the system is not close to the slot assignment limit, no speech need be lost through radio fading. When the probability of a lost packet is below 1% there should be little change in the glitch rate. However, because of the added delay it may be necessary to fix a 20 ms delay to the reproduced speech to avoid breaks.

The hardware requirements of this system will now be considered. Most important and probably most complex of all the hardware is the radio equipment. In order to provide the necessary 1 Mbit/s capacity, an arrangement of ten parallel channels was suggested in chapter 2.

This essentially requires ten separate transmitters in each mobile telephone, though the antenna and final power amplifier stage can be common. Similarly, ten detectors are needed to receive speech.

In addition to speech, the mobile telephone must transmit and receive packets on the information channels. Fortunately, in no case does the mobile have to both receive and transmit at the same time. This is obvious for the speech channel where all packets can be appropriately scheduled by the controller. Also, most acknowledgement channel packets are transmitted in response to and therefore after request channel packets. The only conflict of any consequence is a slot assignment packet coinciding with a request for a subsequent packet. The glitch rate resulting from this is of the order of 10^{-5} in the system described above.

Each mobile telephone requires some digital control and a memory to store the speech. The latter has to hold 0.15 seconds of digitized speech for the transmitter and up to 0.1 seconds from the receiver. This will involve a total of 6250 bits of memory, which is quite trivial by current standards. When recording or replaying speech, the memory has to be accessed at a rate of 25 kbit/s. During packet transmission or reception the rate jumps to 1 Mbit/s. Again this is well within the capabilities of current integrated circuits. A smaller memory is required to store one or two information packets both prior to transmission and for decoding after reception.

A microprocessor can perform the control and packet formation functions. It may even be possible to employ software error correction, for the speech required is not great. Encoding and decoding of the speech can be done by a dedicated integrated circuit. One of the major advantages of such a digital system is the compatability of the control and speech sections. This allows very simple switching and arrangement of the speech into packets.

The hardware at the receiving stations is very similar to that in a mobile. Undoubtedly, the radio equipment must be more sophisticated and of higher power but only limited control intelligence is required. At the central controller a computer performs all the normal exchange functions and also special ones for the mobile system. The same 6250 bits of memory are required for each connection, however this poses no problem for a computer. Algorithms for controlling the transmission queues are a little complex but should not overtax a computer of reasonable size (this was proven by the simulations on a Cyber 173 which entailed far more calculation and still ran at twice real speed).

All the hardware described above is within the range of current technology at reasonable cost. The major difficulty still lies with the radio equipment which must perform in a very hostile environment and with quite a large bandwidth at 100 kbit/s. To ensure maximum efficiency, modulation must be at a rate of 1 bit/s/Hz. Finally, the volume of this equipment and its cost must both be as small as possible in a practical system.

Before alternative mobile telephone schemes are considered and compared with this one, the original dedicated channel digital scheme will be reviewed. This operates in the same way as the scheme above except that interpolation is not used. As a result only C simultaneous conversations can be connected. Since there are no requests for slots, the request and acknowledgement channel capacities can be much smaller. It was shown in chapter 3 that the capacity was finally determined by the need to reduce the request slot time to 10 ms. This results in a figure of 12 kbit/s.

In terms of hardware requirements there is little difference between the two digital schemes. Certainly the radio requirements are virtually identical. The only real savings are a reduction in the memory requirements and perhaps a lessening of the computer power needed at the control centre. Given this, the interpolation scheme is obviously to be preferred because of its higher conversation capacity.

6.2 Alternative Mobile Telephone Systems

Mobile telephone services are presently provided in many countries [2]. Traditionally these services have employed a small number of common channels which are assigned to individual telephones for the duration of a call. Channels are provided by frequency division multiplexing with a channel spacing of 20 to 30 kHz. Fairly high power transmitters of 100 to 200 Watts are used to provide a range of up to 20 km. The total coverage area extends over 100 or more square kilometers and within this area the number of simultaneous calls is restricted to the number of channels available. In many cases the systems are not automatic but require an operator to make connections.

To improve the quantity and quality of mobile telephone services a small cell approach has been proposed [54,96,97]. This involves the use of a number of transmitters each of which covers a small area or "cell" of radius between 1 and 10 km. A large number of channels are made available and a subset is allocated to each cell. The same subset is also used in several other cells, separated by distances sufficient to avoid interference. This technique of frequency reuse increases the effective number of channels within a single metropolitan area by a factor of five or more.

First to propose and implement a cellular mobile telephone system in the USA was the American Telephone and Telegraph Company (AT & T) [3,55,56,98]. This system uses 6 km radius cells with omnidirectional antennas at the cell centre (or site). Each cell site is connected via landline to a mobile telephone switching office interfacing the system to the local telephone network. The switching office is the control centre of the system. It supervises the functions of locating the mobile, of call set up and termination, and of channel assignment and change over.

To initiate a call, special set up channels present in each cell are used. Continuous digital radio signals are broadcast on these channels and if a mobile detects its call number it responds upon the associated return set up channel. Alternatively, if the mobile initiates a call it transmits its identity number and awaits a reply. To complete the call set up the switching office assigns a vacant channel by transmitting its frequency on the call set up channel.

Because a channel may be used only within its own cell, when a mobile crosses a cell boundary it must change channels. A boundary crossover is detected by signal strength measurements at the original, and other cell sites. The new channel's frequency is transmitted by temporarily blocking the conversation and sending the data in a burst at a rate of 10 kbit/s. The mobile then retunes to the new frequency and conversation proceeds.

The AT & T system operates in the 800 MHz band and will eventually have up to 666 radio channels. Frequency modulation is employed with a channel spacing of 30 kHz. Cells are of roughly hexagonal shape and are arranged in groups of seven. No channel is used more than once within any group of seven adjacent cells.

Another small cell mobile telephone system is operated in the USA by the American Radio-Telephone Service Inc. (ARTS) [4,58]. This is basically very similar to the AT & T system but differs in the cell layout. A four cell group is used with six directional antennas at the centre of each hexagonal cell. This requires twenty four separate sets of channels but the directionality reduces interference and increases the antenna gain, so that a one Watt mobile transmitter can be used. This system like AT & T's uses 30 kHz channel spacing and will eventually possess 666 channels.

A small cell mobile telephone system is also being built in Japan. [2,30,59,99,100]. Here the cell sites are connected to mobile control stations and then to a mobile telephone switching centre. Again hexagonal shaped cells are used with fifteen in a group. The optimum cell radius was found to be 5 km for urban areas and 10 km for rural areas. The system occupies the 800 MHz frequency band with a channel spacing of 25 kHz. It is designed to ultimately serve 100,000 subscribers.

The common features of small cell systems are the use of frequency division multiplexing to provide voice channels and the restriction of the channels to only certain cells. The total bandwidth used is 10 to 20 MHz and some 1 to 5% of this is required by control channels.

Compared to the digital schemes there are essentially three areas of difference. Firstly, in the radio frequency channel arrangement, a small cell system has channels of around 30 kHz bandwidth while 80 and 100 kHz are used in the digital scheme. Secondly, analogue (FM) speech transmission is employed in small cell systems rather than digital techniques. Finally, the small cell system does not use interpolation to increase the number of simultaneous conversations but obtains an even greater increase through frequency reuse. In the next section each of these differences will be investigated to illuminate the advantages and disadvantages of both systems.

6.3 Comparison of Digital and Small Cell Systems

Frequency division multiplexing is the standard technique for providing mobile telephone services. Speech transmission in bandwidths of 20 to 30 kHz is well understood and reasonably easy to implement, even in the urban environment. Transmission of digital information at rates of 100 kbit/s is however virtually untried. The greatest bit rate used commercially is the 10 kbit/s used in the AT & T scheme and this is modulated at 0.5 bit/s/Hz.

There are however no theoretical restrictions on transmission at higher rates. The difficulties are basically due to the multipath propagation effects and to the noise in an urban environment. With proper design and diversity reception basic error rates of 10^{-3} should be achievable at data rates of 80 to 100 kbit/s. This can be reduced by appropriate error correction where necessary. It should be noted that small cell systems also use diversity.

The other difficulty with the digital mobile scheme is the need to transmit ten radio signals in parallel. While this is certainly possible it can be complex and expensive, especially at bandwidths of 100 kHz. By comparison, the mobile telephones must be able to switch between up to 700 different frequencies. This however is relatively straightforward with current techniques and can be done cheaply. One disadvantage of the small cell system is that because of the continuous nature of the connection a mobile must transmit and receive signals simultaneously. This is not necessary in the digital scheme. Overall however, the small cell systems have a clear advantage in the actual radio hardware.

Now consider the relative merits of the analogue and digital techniques employed by the two schemes. Transmission of speech signals in analogue form is a big disadvantage for small cell systems. In the urban environment several types of audio impairment occur in FM transmissions [32,72]. The most important are the clicks arising from multipath fading. The nature of these clicks and their frequency depends upon the rate at which the mobile moves through fades and therefore upon the vehicle speed. Another impairment is the Gaussian noise arising from various sources, including thermal, man made and receiver noise. Interference from other users on the same or a different frequency also results in various types of distortion including whistles, clicks and occasionally bursts of someone else's speech.

Diversity reception reduces the frequency of fades and hence of clicks, and companding can improve the signal to noise ratio. However, none of the above impairments can be removed completely and they will always degrade the speech quality.

In the digital system the main effect of signal fading is the destruction of occasional subpackets. It has been shown that as long as this does not happen too frequently, the majority of such subpackets can be retransmitted with no ill effects. Noise of course still occurs, but error correction coding can be incorporated with the speech to maintain the signal to noise ratio near that set by digitization. The extent of the effects of errors can be controlled by an appropriate choice of speech encoding technique. Interference too, can only result in a decrease in the signal to noise ratio and hence its effects are much reduced. Finally, since there is no transmission during silence intervals there can be no degradation in this time.

Thus the digital scheme is characterized by more noise free speech reproduction than the small cell system. The subjective effects should be much better in the former, provided the quality of digitized speech is adequate. This improvement is essentially derived from the digital rather than analogue nature of speech transmission in the very hostile radio environment.

Other advantages accrue to the interpolation scheme from its digital nature. The compatability of the speech and control data has already been noted. As has the relative simplicity with which speech can be handled within the mobile telephone circuitry. Another aspect of this is that the control centre telephone exchange can employ digital switching. It is a relatively simple exercise to convert the packet speech to a PCM format for transmission over digital links in the normal network. This compatability will become more important as more and more of the telephone network is changed over to digital.

A problem with radio communication is that it may be easily overheard. This problem of privacy is an important one in FM systems, where the reception of any desired voice channel is relatively easy. However, imagine the difficulty in trying to intercept a conversation carried in packets. Even if this were possible a digital format makes speech encryption a simple process. Complete privacy can therefore be assured with the digital system.

Now consider a comparison of the number of services each system provides. In the digital scheme interpolation is used to lower the effective bandwidth of a voice circuit, whereas frequency reuse is employed in FM schemes. Interpolation creates some impairment to the speech quality, and this will be considered before the actual system efficiencies are examined.

The digital system has been designed to produce a worst case glitch rate of 0.5%. Subjectively this was found to be noticeable but not objectionable. Glitches in fact , are not dissimilar to the clicks found in FM systems. The key difference, however, is that glitches depend entirely upon the system usage rather than the nature of the channel. If the system is operated at only three to four percent less than its design value, the effects of glitches become insignificant. Because of the non uniform distribution of telephone activity, this will occur for over 90% of the time.

In fact this very aspect of the digital scheme can be an advantage, for there is no definite limit on the number of simultaneous calls possible. In the subjective speech tests of chapter 5, it was found that speech was almost perfectly understandable even at a glitch rate of 6%. If in times of emergency, such a glitch rate is acceptable, the system can be operated at least 10% above its design capacity.

Another degradation in the digital system is the delay which in the present design has a maximum round trip value of 0.3 seconds. The discussion of delays in chapter 4 indicates that this will have little effect on the subjective quality of speech as long as echo suppressors are used. Fortunately, the nature of the system operation with speech detectors, automatically incorporates this to a certain extent. Also, with the speech in a digital form, not only echo suppression, but also echo cancellation, can be achieved reasonably easily.

Simulations showed that the average delay in slot assignment is less than 20% of the maximum at the design operating point. Hence for the majority of the time the round trip delay is under 0.15 seconds, even at full load. Since the changes in delay are gradual and to a certain extent are buffered by the speech memory, it is very doubtful that the delay would be noticeable in practice.

To compare the actual capacities of the two schemes several factors must be considered. Obviously the concept of frequency reuse is the overriding one in favour of the small cell system. However, there is no fundamental reason why the same principal cannot be employed in the digital scheme, in addition to interpolation. Thus initially, the comparison will be performed excluding frequency reuse.

In the small cell system a voice circuit is allocated to a mobile as soon as it goes "off hook". The time required for dialling, exchange switching amd awaiting the called party to answer is therefore included in the total call time. The digital scheme however provides no space in the speech channel until speech actually begins. Consequently, the activity of each subscriber is lower and the number who can be serviced per voice circuit is correspondingly increased.

The benefit obtained from interpolation depends upon the assumed data rate and modulation technique in the digital system. With present assumptions these produse a nominal bandwidth per voice circuit approximately equal to that in the small cell system. Thus the use of interpolation makes the digital system as least twice as efficient.

However, when the small cell system alone employs frequency reuse it is able to service more subscribers. With a total bandwidth of 40 MHz (as proposed in the American systems) a small cell system can support around 200,000 subscribers compared to 80,000 in the digital case. In practice these two figures might well be much closer in a dense city area because of interference factors. Nevertheless, frequency reuse is obviously the key element here, and for this reason its use in the digital scheme will be considered in the next section.

Before this is done however, the original dedicated circuit digital mobile telephone scheme of chapter 3 must again be examined. This system enjoys all of the advantages of the digital scheme except that the maximum number of simultaneous conversations is halved. The use of retransmissions is also more difficult because of the fixed nature of slot assignment. However, the problem of glitches disappears and the delay is reduced somewhat.

This arrangement is very close to a digital version of the small cell approach. The main difference being that one large frequency band is used instead of a number of smaller ones. It is possible to draw the two schemes even closer if, in the digital system, individual digitized conversations are transmitted in their own channels of 30 kbit/s capacity. The advantages arising from such digital transmission, have of course been recognised before [65] and in fact a system along these lines has been suggested by Feggeler [101]. In this particular case, the signal to noise ratio of the digital technique was found to be about 20 dB poorer than that of the analogue system. This was due to the high bit rate of 48 kbit/s assumed for digitized speech.

6.4 Small Cell Packet Systems

The simplest method of expanding the digital interpolation system is to use several of them in parallel. This involves duplicating all of the radio channels at different frequencies. Savings are possible in this if some of the equipment is made common to all of the systems. In particular the receiving station sites and their antennas can be common and, providing the system does not become too large, the central computer can control all systems simultaneously. Individual mobile telephones are attached to just one system and do not need to swap frequencies to operate anywhere within the service area. This arrangement of course, incorporates no element of frequency reuse.

For frequencies to be reused they must be restricted to cells covering only a small fraction of the service area. Thus an entire digital system must operate within one cell in a group of 10 to 20 cells. Other cells then contain systems at different frequencies and the entire group is repeated as often as necessary. With this arrangement each mobile telephone must be able to operate in any of the systems and must change from one to another as it crosses a cell boundary, just as in the

FM small cell schemes.

Such a network can be implemented in stages as demand grows. Initially only two or three systems would be required, as this number is sufficient to handle around 10,000 subscribers. A truly parallel arrangement can be used, with each system covering the entire area and each telephone able to operate in only one of the systems. Eventually however, with a sufficient increase in the number of subscribers, a swap to the cellular system in proper would be necessary, and for very large systems, frequency reuse must be employed.

Now the number of circuits provided by the digital interpolation system is 87, which is far greater than the number used in each cell in the present small cell systems. If this number of circuits is used, the cells have to be very large and this will inhibit frequency reuse. In the interests of efficiency the size of the digital system must be reduced in this case.

If a nominal number of circuits, C, of 20 is used in place of the present 40, the efficiency of interpolation falls. The system interpolation gain is 1.82 with a packet length of 0.2 seconds and other parameters as in the standard system. This allows 40 simultaneous conversations within a system and is probably a better size for a small cell arrangement. Reducing the system size by a factor of 2 also has other advantages. The capacity required by the system can be halved, either by reducing the capacity of each parallel speech channel from 100 to 50 kbit/s or by halving the number of parallel speech channels.

Up to now, reuse has been considered only in terms of frequency, but it is also possible to reuse slots. Consider for instance a C = 40 system which is time shared between two adjacent cells such that each slot may be used in only one of the two. Then since a telephone can operate only in alternate slots, the system is effectively of size C = 20. Additional C = 40 systems, similarly split, can be used to complete the group of cells which is repeated to cover the area required. The advantage of such an arrangement is that a telephone need only be able to switch to half as many different frequencies as with the proper C = 20 system. In essence, frequency switching is replaced by time switching. There is of course, no frequency change at all when a telephone moves from one cell of a pair to another.

It is possible to split the C = 40 system into more parts and extend the time switching concept over wider areas. A problem arises however, for with each reduction in the effective size of C in any cell, there is a reduction in the system interpolation gain. If the effective C is much below 10 there will be no gain at all from interpolation.

To understand this more clearly, consider the queue limit L. It is this factor which is changing when the system is divided. For instance, if only alternate slots are available within a cell then there will be only half the number of slots in the time limit for slot assignment. Equation (5.8) shows that this is exactly equivalent to halving C.

The only way to divide the slots and still maintain an effectively large L, is to make the assignment of slots to particular cells partially or fully dynamic. This means in essence, that if a subpacket cannot be fitted within the slot assignment limit, using only slots available in its cell, then a slot must be borrowed from an adjacent cell. This has implications for the slot assignment in the remainder of the network. For if a slot is used in the wrong cell it is one cell length nearer to an interfering slot being used in the adjacent group of cells.

The same problem arises in FM small cell systems because of the limited number of channels available in any cell. The probability "p" of blocking a new call, in a cell with "n" available circuits and an offered traffic of "y" erlangs, is given by the Erlang formula [95] as

$$p = \frac{y^n/n!}{1 + y + y^2/2 + \ldots + y^n/n!} \tag{6.1}$$

The smaller the number of circuits available, the smaller the allowable traffic per circuit for a given blocking probability. It has been found however, that this quantity is increased significantly in small cell systems if even a few circuits can be borrowed from adjacent cells [102]. This has spurred some investigation into suitable algorithms [103].

The situation in the digital system is quite similar and the same sort of algorithms can be employed. In the extreme case, slots are allocated in an entirely dynamic manner. Then a standard C = 40 system is spread over the entire service area and every slot is used within each group of cells. Cells in adjacent groups using the same slots must be well separated to avoid interference. Thus the allocation of slots depends not only upon the priority of elements in the queue, but also upon the position of each queue entry in its group of cells. An algorithm which at the same time minimizes the delay in each queue and also prevents interference will undoubtedly be complex indeed.

If such a dynamic slot division scheme can be implemented it will quite probably require some sacrifice in the interpolation gain. The extent of the reduction will depend upon the actual algorithm used. The big advantage of such a scheme however is that a mobile telephone has to operate in only one C = 40 system, no matter where it is in the service area. Expansion is achieved by using further systems in parallel. This simply entails increasing the equipment at each cell site and connecting new subscribers at the new frequency.

Hence the digital interpolation scheme can employ frequency reuse in a number of ways. It is a little less suited to this role than the FM systems because of the block nature of circuit provision, and the minimum system size necessary to utilize interpolation efficiently. Nevertheless, the reduction in efficiency is not sufficient to overcome the advantage of the digital acheme. It can provide more and higher quality services than the FM small cell system at a cost of somewhat greater complexity and a little extra hardware.

6.5 Conclusions

In this chapter an optimal digital interpolation system has been designed. It is based upon two assumptions.

1 : that a 1 Mbit/s digital data rate can be achieved in an efficient manner over radio channels in the urban environment

2 : that speech can be digitally encoded in an error resistant manner at a bit rate of 25 kbit/s, and provide quality suitable for mobile telephone use.

The resulting optimal system has a packet length of 0.32 seconds and a recording delay of 40 ms. The maximum slot assignment delay is set at 0.1 seconds giving a total round trip delay of between 0.1 and 0.3 seconds. It was seen that where necessary, as for instance in long distance calls, the delay could be kept to the minimum figure. At a glitch rate of 0.5% this system provides 87 conversations within the total 2.16 Mbit/s capacity required and is therefore capable of serving up to 4100 subscribers.

A consideration of the hardware requirements showed that the radio equipment design is the most difficult aspect. The overall layout of the system with radio repeaters and a central control is remarkably similar to that of the small cell arrangement used in modern mobile telephone schemes. In comparing the equipment needed by these two schemes the main difference was found to be in the radio requirements which were more onerous in the digital case.

This scheme however has many advantages in other areas. It is far less effected by multipath propagation in the urban environment and will possess much less noise in the reproduced speech. In particular the effects of fading are virtually eliminated by packet repetition. Other advantages of the digital scheme include the ease of speech manipulation both at the telephone and in the exchange, and the possibility of voice scrambling to ensure privacy.

Small cell systems do not employ interpolation and therefore, with the assumptions above, the digital scheme is at least twice as efficient at providing voice circuits in a given bandwidth. The use of these circuits is also better in the digital case since call set up takes place "off air". However the power of reusing circuits five to ten times means that in large systems a small cell arrangement must be able to service more subscribers.

Digital interpolation systems employing reuse were also investigated and a number of practical approaches were found. In all of these the gain due to interpolation is somewhat reduced from that of the system described above. However to completely offset the advantage of the digital system it would be necessary to reduce the speech channel capacity to 500 kbit/s, and to increase the data rate of digitized speech to around 40 kbit/s. If in any practical system, improvements upon these figures are possible, then the digital interpolation scheme can service more subscribers per unit bandwidth than the FM small cell system and in addition has all the advantages provided by digital transmission.

7. A PURE TASI MOBILE TELEPHONE SYSTEM

7.1 System Description

One of the major advantages of the packet mobile telephone system is the ease with which interpolation can be implemented. It is also possible however, to interpolate mobile telephone speech in other ways. For instance, a voice circuit can be allocated to a mobile when speech begins and held until speech ceases. This is the method employed in the original time assigned speech interpolation (TASI) transatlantic undersea cable system. Its implementation is a little more difficult in a mobile environment, but it is viable and represents an alternative to the packet approach. In this chapter such a TASI scheme will be investigated and compared to the equivalent packet system.

The operation of a TASI system again involves time division multiple access (TDMA) in the radio channel. Therefore a reservation scheme is required to achieve maximum throughput. This entails each telephone notifying the controller, via a request channel, when speech begins. Since these requests are again random, a slotted ALOHA protocol must be used and for each request an acknowledgement must be sent on a return channel.

Once a voice circuit has been selected by the controller its identity is returned via the acknowledgement channel and speech transmission can begin. This set up process requires some time and hence a delay must be inserted into the speech path to avoid loss. When the period of speech is finished the mobile ceases transmitting and the voice circuit is freed for another user. At the start of each subsequent talk the mobile must again request a channel.

With this system, speech transmission is continuous and at the real time rate. Voice circuits exist as separate radio channels and may contain analogue or digital speech. There is therefore no possibility of speech retransmission if a portion is lost in a fade, but the provision of voice circuits is identical to that in FM small cell schemes and is therefore simpler than in the packet system. Also the TASI arrangement requires no delay either in recording the speech or in the assignment of circuits. Instead, if no circuit is available when speech begins, all the speech is lost until a circuit is freed. Such a speech loss is termed a freezeout.

Freezeouts in TASI schemes are the equivalent of packet glitches. If reasonable speech quality is to be maintained the freezeout fraction must be kept below a set level. The suggested value in other TASI schemes is 0.5% [77]. It is possible to analyse TASI schemes theoretically to derive an expression for the freezeout fraction in terms of the system parameters. This will now be done.

7.2 Theoretical Freezeout Fraction

Theoretical expressions for the freezeout fraction have been derived by several authors [37,42,79,80]. Unfortunately their results differ, though not by large amounts. The correct analysis is due to Weinstein [80] and is presented here.

Consider a TASI system with the following parameters

N : the number of speech sources

C : the number of voice circuits

P : the probability that a source is issuing speech

Assuming that the sources behave independently, the probability that k of them are active at any time is given by the binomial distribution as

$$b\ (k,N,p) = \binom{N}{k} p^k (1-p)^{N-k} \tag{7.1}$$

However, since only C circuits are available if more than this number of sources are active the excess speech must be lost. The average amount of speech lost through freezeout is

$$F_r = \sum_{k=C+1}^{N} (k - C)\ b\ (k,N,p) \tag{7.2}$$

The average amount of speech generated is simply N p. Thus the fraction of the speech lost i.e. the freezeout fraction is

$$F = \frac{1}{N\ p} \sum_{k=C+1}^{N} (k - C)\ b\ (k,N,p) \tag{7.3}$$

Note that the freezeout fraction does not depend upon the nature of the speech length distribution, and the only assumption required is that the sources are independent. A more rigorous derivation, proving these points in detail is given by Weinstein.

A computer calculation based upon (7.3) provided the freezeout fraction for various combinations of other variables. Figure 7.1 shows how the freezeout fraction changes with the interpolation gain (given by $\frac{N}{C}$) for various numbers of circuits. The freezeout at a particular interpolation gain decreases with increasing C. This is physically reasonable since better averaging of the speech input occurs in larger systems and therefore there is less time in which no circuit is available.

Similarly, figure 7.2 shows the variation in freezeout fraction with interpolation gain for different speaker activities. The freezeout

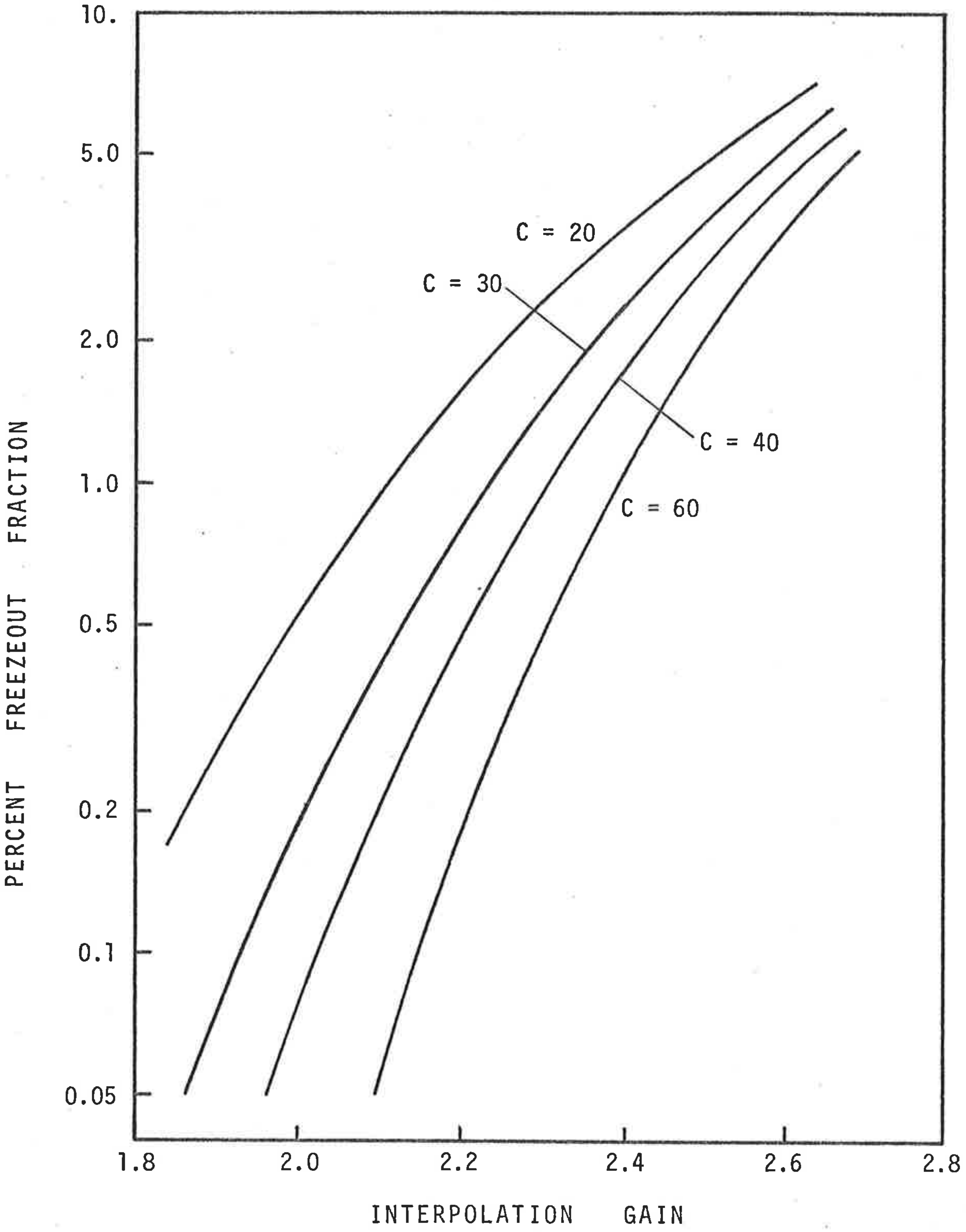


Figure 7.1 Theoretical TASI freezeout fraction at 0.38 speech activity

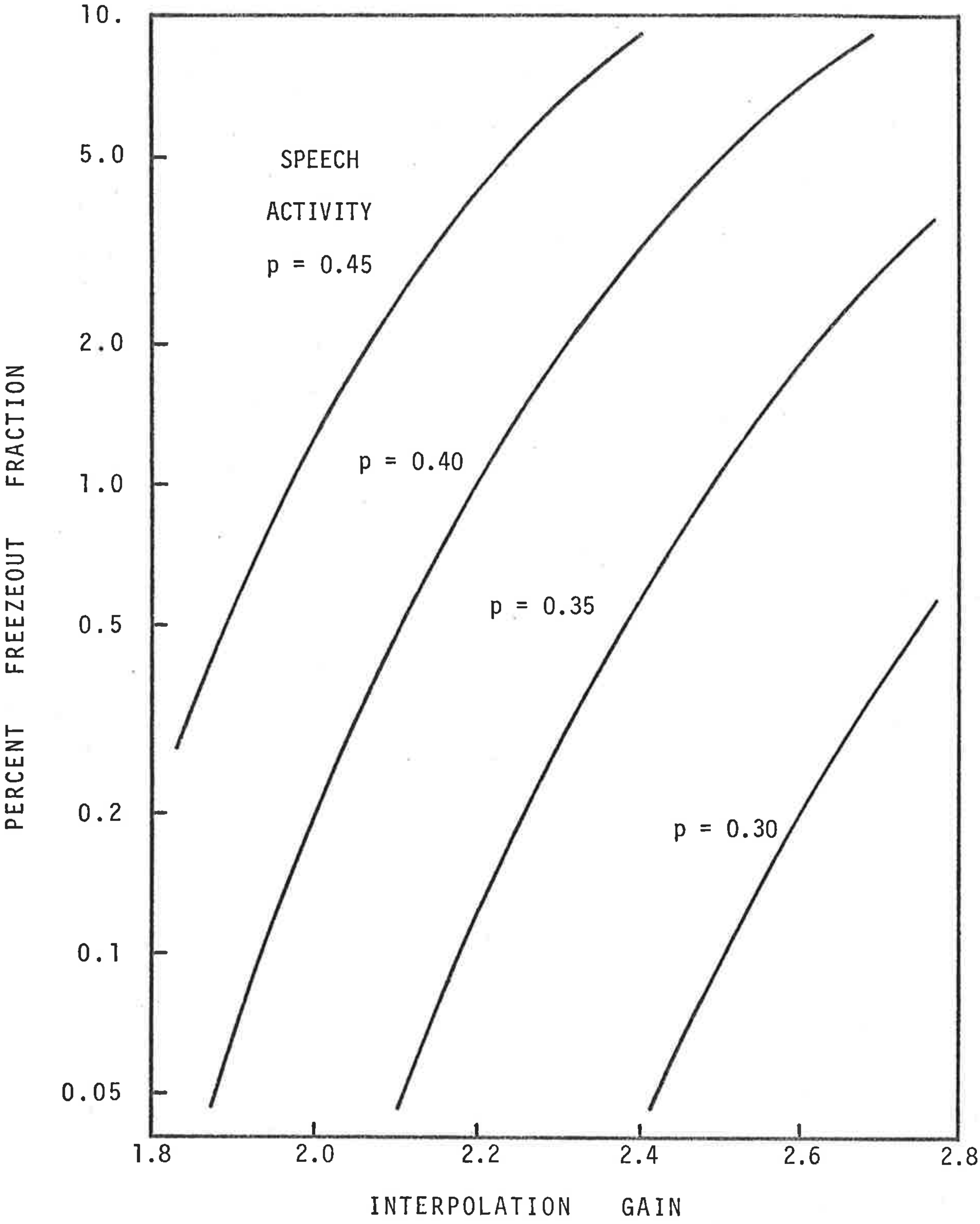


Figure 7.2 Theoretical TASI freezeout fraction for various speech activities

decreases with decreasing activity as might be expected. In a practical system the activity is set by the nature of the speech and also by the type of speech detector used. Realistic values of p would be between 0.35 and 0.4, just as in the packet system.

It is interesting to compare the freezeout fraction with the theoretically derived glitch rate of a packet system. Figure 7.3 shows the freezeout fraction for a system of C = 40 circuits and of activity, p = 0.38. Also shown are the theoretical glitch rate curves for the same size packet system with the same speech activity and various queue limits, L. Usage in a packet system is converted to an interpolation gain by the relation

$$I = \frac{N}{C} = \frac{U}{p} \tag{7.4}$$

All of the quantities defined for the TASI system are identical to their counterparts in the packet system. The freezeout fraction and and the glitch rate are directly comparable because both represent the fraction of the total speech lost.

Consider a packet system with no delay allowed, i.e. L = 0. Then if more than one packet arrives in any slot all but one will be glitched. Previously the probability of various numbers of arrivals has been described by a Poisson process. However it was stated in chapter 3 that the binomial distribution is the more correct in a finite sized system. Its use was found to produce only minor changes in the final glitch rate and therefore to avoid specifying the system size, N, a Poisson distribution was employed. Here, in a theoretical comparison of the two systems, the binomial distribution is appropriate.

The probability of any source producing a packet in a particular slot is given by

$$q = \frac{U}{N} \tag{7.5}$$

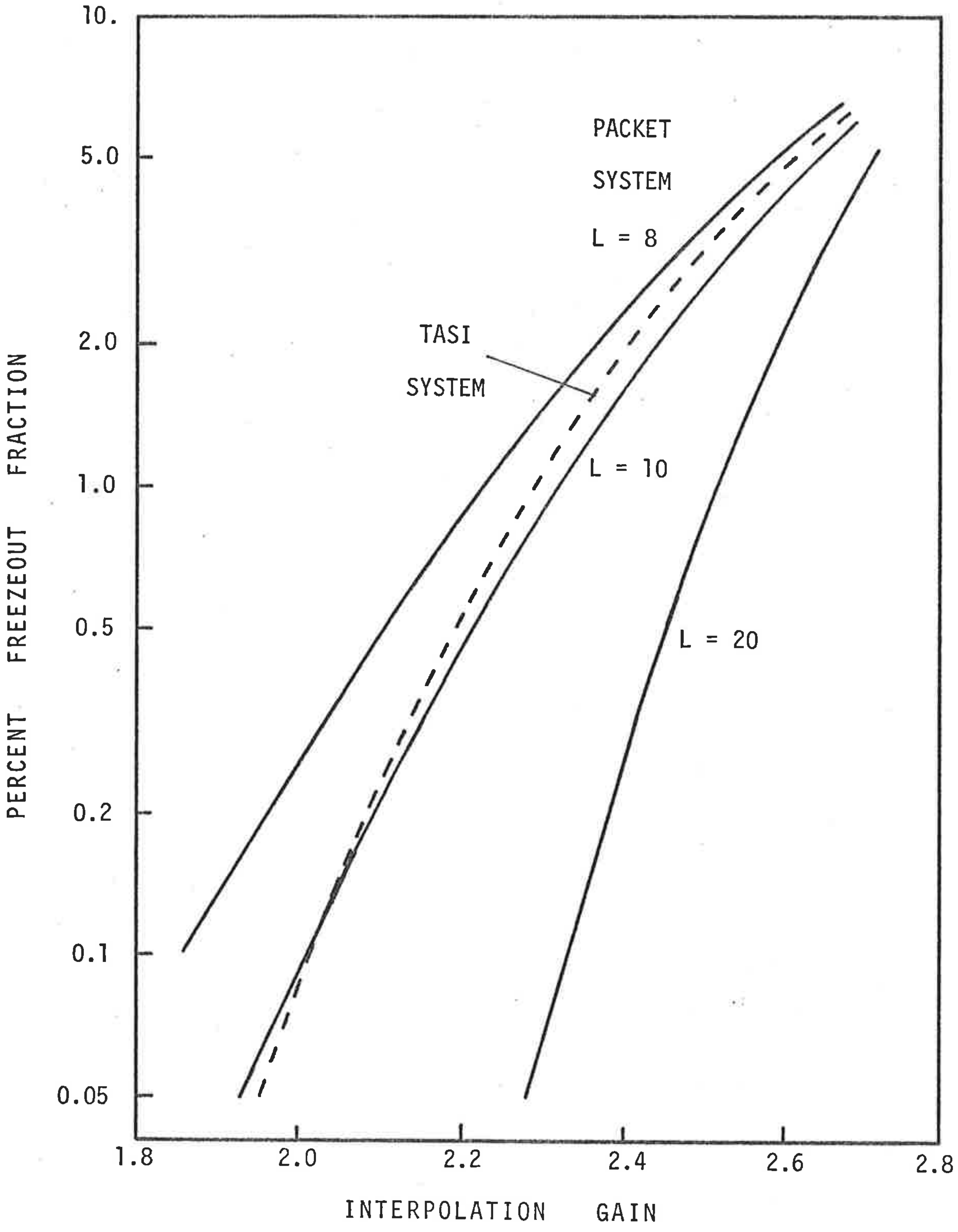


Figure 7.3 Theoretical comparison of speech loss in TASI and packet mobile telephone systems

The probability of k arrivals in any slot, given a system of N sources is b (k,N,q). In a queue with no allowed delay all but one of these will be glitched. Thus the probability of a glitch becomes

$$GP = \sum_{k=2}^{N} (k - 1) \; b\,(k,N,q) \tag{7.6}$$

and the glitch rate is

$$GR = \frac{1}{U} \sum_{k=2}^{N} (k-1) \; b\,(k,N,q) \tag{7.7}$$

This may be rearranged by using (7.4) to

$$GR = \frac{C}{N\,p} \sum_{k=2}^{N} (k-1) \; b\,(k,N,\tfrac{p}{C}) \tag{7.8}$$

If the system has only a single circuit, i.e. C = 1 then (7.8) becomes

$$GR\Big|_{C\,=\,1} = \frac{1}{N\,p} \sum_{k=2}^{N} (k - 1) \; b\,(k,N,p) \tag{7.9}$$

But this is exactly the expression for a TASI system with a single circuit (from (7.3)). Thus the two systems provide identical degradation under these circumstances. This is a common point from which the responses diverge as the system parameters are varied.

In the TASI case, increasing the number of circuits reduces the fractional speech loss as shown in figure 7.1. The effect of C in the theoretical packet system is quite different. Here increasing C merely causes the arrivals to more closely follow a Poisson destribution. At very low C this will change the glitch rate somewhat, but for C > 20 the change has been seen to be very small. Instead it is the queue limit which has the most significant effect in this case, as can be seen by the substantial changes in the packet curves of figure 7.3.

7.3 Effects of the Request Channel

In the TASI system a request must be transmitted at the start of each speech interval or "talk". At the completion of the talk the mobile ceases to transmit. This is detected by the system and the central controller is notified without the need for any special transmissions. Now the average length of a talk as determined from the computer simulations is 1.79 seconds, and there is a maximum of C in progress at any one time. Thus the maximum average number of talks that begin per second, and hence the request rate, is $\frac{C}{1.79}$.

Request packets are also required for call set up. Since this operates just as in the packet system the same number of packets per second will be required. The total request channel capacity needed is therefore (from (2.7))

$$Wr = A\ B\ C\ \left(\frac{1}{1.79} + \frac{1}{20}\right) \tag{7.10}$$

The number of bits in a request packet, B, may also be the same as in a packet system. A value of 120 bits will be assumed. The ALOHA factor A was taken as 5 in the packet system, but here a larger value is required to reduce the probability of a collision. To understand the importance of this consider the delays involved in requesting a speech circuit.

A talk begins at a random time and activates the speech detector after a short delay. At the beginning of the next request slot, a request is transmitted. Assuming this is correctly received there will be a further delay while the central computer allocates a voice circuit and then informs the mobile via an acknowledgement packet. At the mobile this packet must be decoded before speech transmission can begin.

Reasonable estimates for the times of these elements are

5 ms for speech detection

8 ms for circuit allocation

1 ms for decoding the acknowledgement packet.

Thus the total delay between speech initiation and the beginning of transmission is on average

$$\text{Speech delay} = 14 \text{ ms} + 2.5\, Z_r \tag{7.11}$$

where Z_r is the request and acknowledgement slot length.

This delay will be increased if the request packet experiences a collision. At least three slots must be allowed for an acknowledgement packet to arrive before the mobile can assume that the request packet has collided. It will then retransmit the request at random in one of the subsequent K slots. Thus the additional delay due to each collision is

$$\text{Collision delay} = \left(3 + \frac{K}{2}\right) Z_r \tag{7.12}$$

To obtain numerical values for these delays it is necessary to assume a request channel capacity. Initially a value of 10 will be assumed for A. With C = 40 and B = 120 bits the request channel capacity becomes 29.2 kbit/s. An appropriate value in practice is 30 kbit/s since the request channel would then occupy one FM voice channel at a modulation rate of 1 bit/s/Hz. This capacity produces a request slot length of Z_r = 4 ms and results in a speech delay of 24 ms. With the standard value of 5 for K the extra delay inserted by each collision is 22 ms.

The speech delay is experienced each time a circuit is required. If no compensating delay were incorporated in the speech path this amount of speech would be lost from each talk, producing a freezeout fraction of 1.3% from this cause alone. Consequently a fixed delay must be used and the most appropriate value is the speech delay itself, i.e. 24 ms.

Even with this delay some speech will be lost due to collisions. The probability distribution of collisions was measured in the computer simulation of the packet system and is shown in figure 5.3. For an ALOHA factor of 10 and with K = 5, the average number of collisions per new packet is 0.151. Thus the average delay added by collisions is 3.3 ms and the resulting freezeout fraction is 0.19%. This holds when the system is used at full capacity. With lower usage the probability of a collision and hence the fractional speech loss are reduced proportionally.

Thus the original choice of 10 for A is required to produce a reasonable figure for the speech loss from collisions. To reduce this loss even further it is necessary either to increase the fixed delay and allow for one or more collisions or to reduce the request slot size by again increasing A.

In the first case, increasing the fixed delay to 46 ms would result in a fractional speech loss due to collisions of 0.053%. Alternatively, doubling the capacity of the request channel would produce a speech delay of 19 ms and a fractional speech loss around 0.05%. In a practical system the former method is preferable because of the benefits derived by having all the speech and information channels of the same bandwidth.

Finally it is interesting to compare the request channel capacity required by the TASI system, with that of the packet system. In the packet case a capacity of 80 kbit/s was needed for a C = 40 system whereas the TASI system of the same size requires only 30 kbit/s. This difference arises because the capacity is directly proportional to the number of requests transmitted per second and hence is inversely proportional to the length of speech covered by each request. In the TASI system this is the average talk length of 1.79 seconds, while in the packet system it is simply the packet length. Since the latter is much smaller then 1.79 seconds, the capacity required by the packet system is much larger.

7.4 Simulation of the TASI System

To confirm the above theoretical figures for the freezeout fraction a computer simulation of the TASI system was performed. In this programme, speech is generated just as in the previous simulation but is not placed into packets. Instead, when a pair of talkspurts for any conversation are formed, the starting time and length of each talk and the end of the silence are stored. Then, when the final talk is transmitted more speech is generated and stored.

Requests are formed in the correct time order for each talk and are tested for collisions as before. If collisions occur the requests are resequenced in accordance with the delay calculated above. Upon a successful transmission a free channel, if available, is allocated. If all channels are in use at the time, the first to be released is allocated and the resulting speech loss is measured.

Also measured are the speech offered and the total time for which voice circuits are occupied. Conversations are started up over a 30 second period and to avoid any initial bias, the above quantities are not retained for the first 40 seconds of simulation. The actual usage in the system is determined from the long term average value of speech offered per second. This quantity can also be predicted from the values of N, C and p, but the measured values are found to differ from those predicted because of random variations in the speech input. This means that the speech activity, p, varies somewhat with time in the simulation.

Theory indicated that p must be kept constant if valid comparisons of the freezeout fraction are to be made at different numbers of users, N. Thus in each programme run the cumulative average values of p and the freezeout fraction, F, were printed at regular intervals. When the last few values of these are plotted they show the incremental changes in F with p and enable the freezeout fraction at the desired speech activity to be obtained by graphical means.

The first task undertaken with the simulation was a confirmation of the TASI theory. This involved setting all delays in the programme to zero and not resequencing requests that collided. Comparisons between the simulation results and theory over a range of values of C and N were performed as shown in table 7.1. The agreement is clearly remarkably good.

Freezeout Fraction (%)	N = 65, C = 30	N = 70, C = 30	N = 94, C = 40
Theory	0.7187	1.8613	1.4325
Simulation	0.716	1.834	1.429

Table 7.1 Confirmation of TASI theory from simulation

Further simulations of the C = 40 system were then performed with delays set at the values found previously. The results appear in figure 7.4 together with the theoretical curve. The difference between the curves increases linearly with the interpolation gain from 0.17% at a gain of 2.1 to 0.26% at a gain of 2.4.

Most of this difference arises from collisions. These have been shown to increase the speech loss by an amount which increases with the interpolation gain up to 0.19% at a gain of 2.64. Another source of loss is a speech delay in excess of the average level, at which the fixed delay was set. This depends upon exactly when a talk starts with respect to the beginning of a request slot and can add a further 0.03% to the speech loss. Finally, there is time lost in the voice circuits in changing from one source to another. This will effectively increase the usage by up to 0.5% and again increases the speech loss by an amount which is small but increases with the interpolation gain.

7.5 Comparison with the Packet System

A comparison of the TASI and packet mobile telephone systems must be based upon the number of services each provides within a given bandwidth. This is measured directly by the system interpolation gain, which for a packet system of size C = 40 is given in figures 5.8 and 5.9.

When this gain was calculated, the efficiency of the packet format and the capacity of the request channel had to be considered. However, in a TASI system, the transmission efficiency is always at the maximum since a channel is used for exactly the length of the talk being transmitted, and no silence is included. Therefore, to convert a TASI

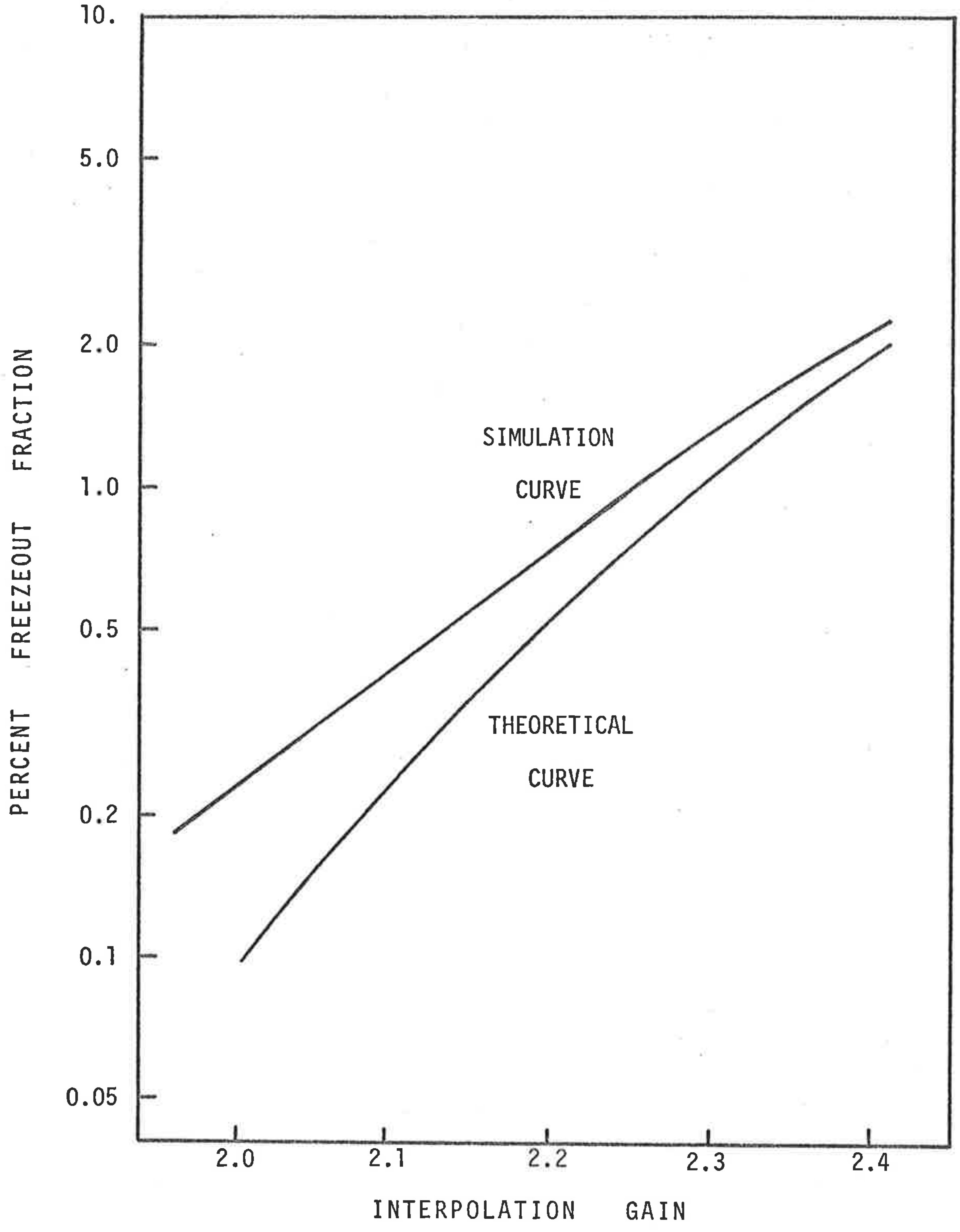


Figure 7.4 Theoretical and measured freezeout fraction in TASI system

interpolation gain (i.e. $\frac{N}{C}$) to a system interpolation gain it is necessary only to multiply by the fraction of the bandwidth devoted to speech. For the C = 40 system this is 97.56%.

Table 7.2 compares the interpolation gains of both systems at different fractional speech losses. A packet length of 0.32 seconds is assumed in the packet case and the gains are presented for assignment delays of 0.1 and 0.2 seconds.

Fractional Speech Loss	0.25%	0.5%	1.0%
Packet System V = .2	2.003	2.052	2.114
Packet System V = .1	1.971	2.023	2.086
TASI System	1.973	2.085	2.196

Table 7.2 Comparison of speech interpolation gains : C = 40

On this basis the TASI system slightly outperforms the packet system. The theoretical advantage that the packet system appeared to possess is overcome in practice by its greater request channel capacity and by the inefficiency of the packet format. Notice however that the advantage of the TASI system is less at smaller fractional speech losses. Also, the exact position of the two systems depends significantly upon the delay allowed in slot assignment. The packet system's gain can always be improved by increasing this delay.

A complicating factor in the above comparison is that the two systems use different bandwidths for a voice circuit. The bandwidth per voice circuit in the packet case is 25 kHz while that in the TASI case is 30 kHz. Thus, strictly speaking, the system interpolation gain for the latter should be reduced by 20%. However the choice of a 30 kHz bandwidth for an FM voice channel appears to be somewhat arbitrary and 25 kHz would probably suffice.

The only difficulty with this change would be fitting the request channel within the reduced bandwidth. In the worst case two of the new circuits would be required by the request channel. Even then one would be sufficient for the acknowledgement channel and the system interpolation gain would be reduced by only 1.22%. This would not substantially alter the results in table 7.2.

It is also important to compare the subjective effects of the speech loss in each system. There can be no guarantee that a given speech loss in both will result in the same subjective degradation. This aspect has already been examined in the packet scheme through statistics accumulated in the simulation programme. The same technique was used in the TASI simulations to obtain the length and frequency of freezeouts.

Figure 7.5 shows the distribution of freezeout lengths at two values of the freezeout fraction. Also shown is the distribution of glitch lengths for a packet system with a 0.4 second packet length and 4 subpackets. Clearly the speech lengths lost are much smaller in general in the TASI system, though the distribution depends strongly upon the freezeout fraction. The average freezeout length is reduced from 51 to 12 ms when the freezeout fraction is reduced from 1.8% to 0.4%. At the same time the percentage of talks experiencing some freezeout is reduced only marginally from 64% to 55%.

This is quite the reverse of the packet system response where reducing the glitch rate barely changed the glitch length distribution,

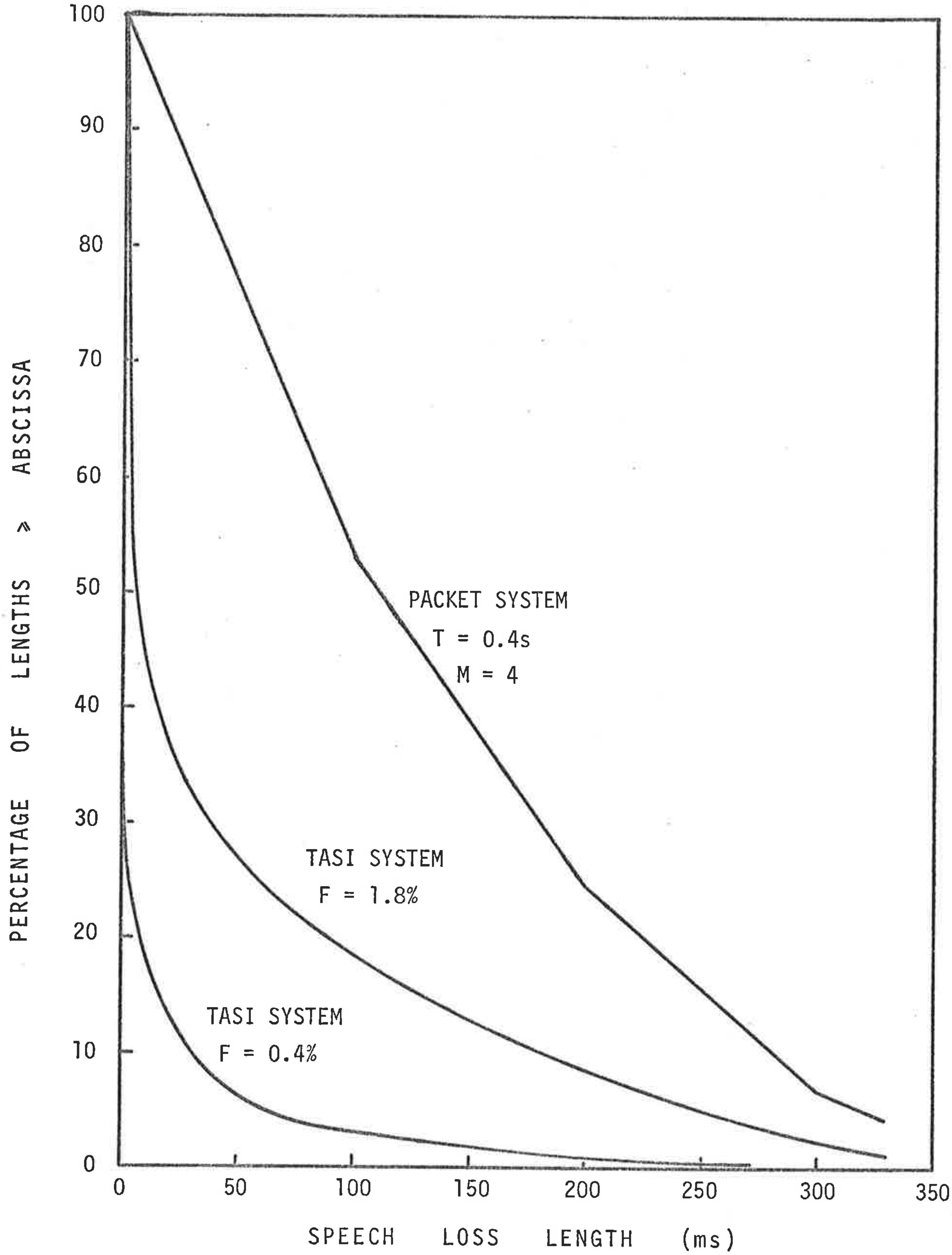


Figure 7.5 Distribution of speech loss lengths in TASI and packet systems

but substantially reduced the glitch frequency. Thus in contrast to the bursty speech loss in packet schemes the loss is quite evenly distributed in the TASI case. More than half the talks experience some speech loss but the amount is usually small.

Another major difference between the two schemes is that freezeouts only occur at the start of talks whereas glitches can occur anywhere. This may result in a subjective advantage for the TASI system. It has been suggested that for virtually all types of speech the first 20 to 30 ms of talk may be removed with no discernable effect [88]. This is because the major components of speech all exceed this length and their latter parts alone are sufficient to enable their recognition. If this is in fact the case, then the freezeout fraction will effectively be somewhat reduced from its actual value. The gain however is relatively small, since freezeouts of length less than 20 ms account for only 30% of the total speech loss.

Another aspect in which the TASI system is superior is in the amount of delay introduced. In the packet system the minimum recording delay is determined largely by radio transmission efficiency considerations and cannot be much below 40 ms. Even larger though is the slot assignment delay which directly effects the glitch rate and hence the interpolation gain achievable. Reducing the slot assignment delay from 200 ms to 100 ms reduces the interpolation gain by almost 2%. Halving the delay again to 50 ms results in at least a further 4% reduction and causes some degradation in other areas. Thus the total maximum one way delay cannot realistically be reduced below 100 ms.

The one way delay in the TASI system however is set at only 24 ms to provide the system interpolation gains of table 7.2. This will produce little or no subjective effect and may even remove the need for echo suppressors. In a normal network an echo suppressor probably would be required with this delay but because of the action of the speech switch the subjective effects of echoes will be considerably reduced.

In respect of hardware requirements, the TASI system falls midway between the packet and the small cell systems. The radio equipment required by the TASI system is almost exactly the same as that of the small cell FM system. The only difference is that the bit rate on the request channel is much greater than that on the call set up channel. Both TASI and packet systems require a speech switch and a means of delaying the speech. These would actually be a little more difficult to implement in the TASI system if it employed analogue rather than digital speech transmission.

The final area for comparison is the method of expansion in both systems. The TASI system is capable of operating in a true FM small cell arrangement with frequency reuse. The only restriction is that the cells must contain sufficient voice circuits to provide a large interpolation gain. If for instance cells possessed 20 voice circuits instead of the 40 assumed above the interpolation gain would theoretically be reduced by 11%. This compares to a loss of 8% incurred in changing from a C = 40 to a C = 20 packet system. Thus TASI system can employ frequency reuse and its gain will be close to the maximum as long as the cells do not become too small.

7.6 TASI System With Delay

The essential difference between the packet and TASI techniques is the method used to control the speech loss. In the packet case this involves allowing speech to be delayed to produce a more even load. The TASI system on the other hand relies upon the spread in starting and finishing times of talks in a number of conversations to keep freezeout lengths small. The total speech loss in this case is reduced by increasing the number of circuits in use and therefore decreasing the waiting time for a free circuit. It is also possible however to use delay to reduce the speech loss in TASI systems.

Consider the situation with a delay allowed before circuit assignment. If a talk begins when no circuit is free the speech is simply delayed until either a circuit becomes available or a set time limit is reached. A limit is necessary to maintain acceptable speech quality and also because the speech memory must have a finite size. If the delay reaches the limit, speech will then be lost in the usual manner until a circuit can be allocated.

The extent of the reduction in speech loss through the use of delay was determined by simulation. This involved modification to the TASI programme to incorporate a variable delay between the request for a circuit and its assignment. Various statistics on the length of delays used were accumulated in addition to the usual fractional speech loss and usage. Runs were performed with delay limits from 20 ms to 200 ms for several different system configurations.

When the resulting speech loss at a particular interpolation gain is plotted against the maximum delay allowed, a remarkably linear relationship in found. This holds at different values of the interpolation gain and also for systems of different size, as shown in figure 7.6. It is also clear from this figure that the slope of the lines increases in magnitude when the zero delay speech loss is lower. The percentage freezeout fraction with a delay limit of "d", donoted F (d) is given approximately by the formula

$$F\,(d) = F\,(0)\;\; 10^{-d\,(2.75\,-\,0.587\,F\,(0)\,)} \qquad (7.13)$$

where F (0) is the percentage freezeout with no delay allowed and d is measured in seconds.

In obtaining these results the various delays in the system were changed and in some cases the request channel was entirely removed. Thus (7.13) appears to be quite general. It will give an approximate value for the freezeout fraction in any useful TASI system employing delay. Alternatively it will give the reduction in freezeout fraction which may be achieved through the use of delay.

A delay also changes the distribution of freezeout lengths as shown in figure 7.7. Any delay in excess of 50 ms results in the changes shown here. Essentially the distribution is far more uniform since the vast majority of small freezeouts are eliminated. This is reflected in the percentage of talks experiencing a freezeout, which drops from 64% with no delay to 19% with a maximum delay of 50 ms.

One potential problem if an assignment delay is allowed is that noticable changes may occur in the timing of the reproduced speech. This happens if the delays of two consecutive talks differ by an amount comparable with the time separating the talks. To avoid this the change in delay between adjacent talks must be controlled. This was in fact done in the simulations. In no case was the delay on a particular talk allowed to differ from that on the previous one by more than half of the time separating the talks.

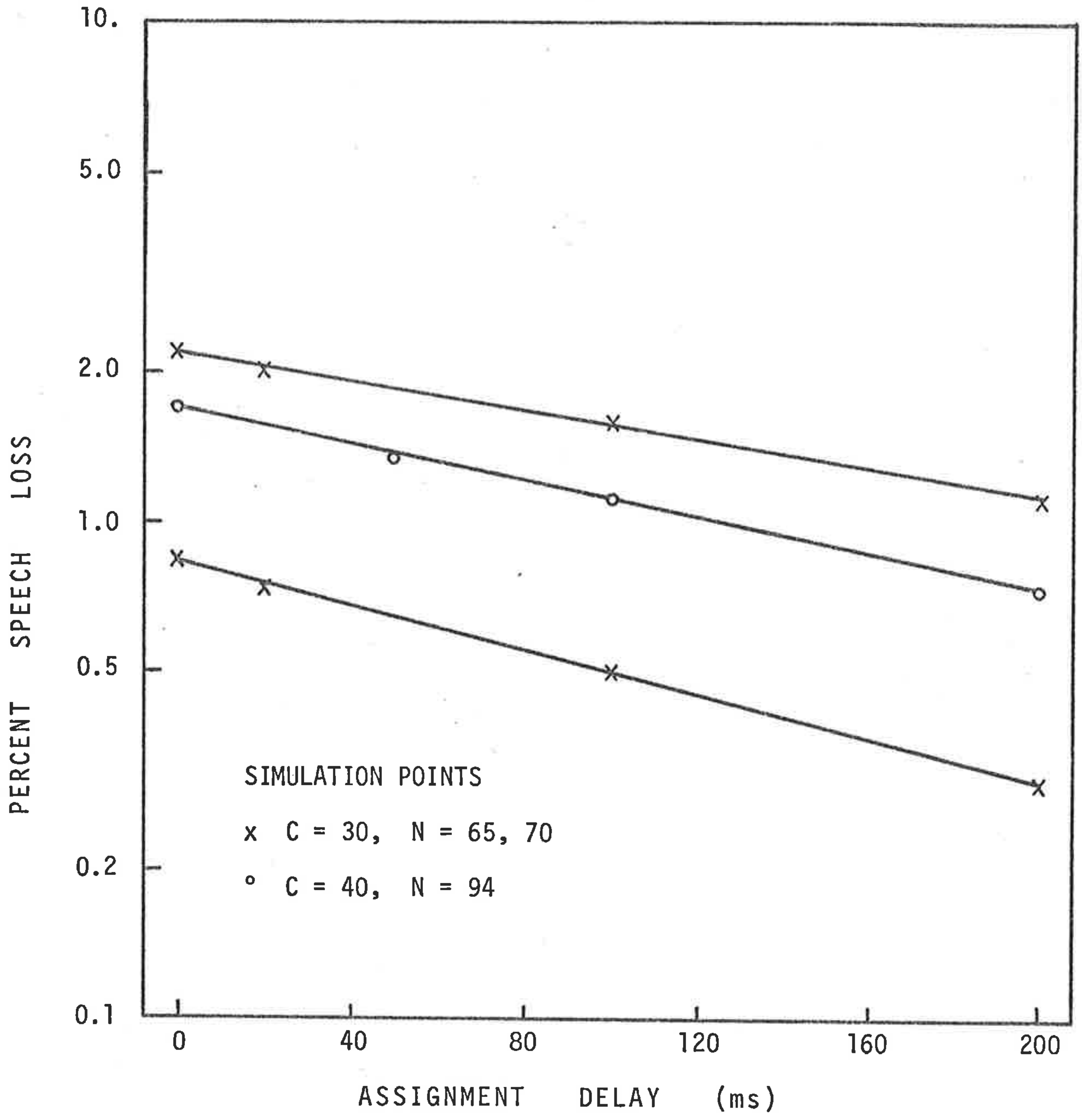


Figure 7.6 Effect of assignment delay on speech loss in TASI systems

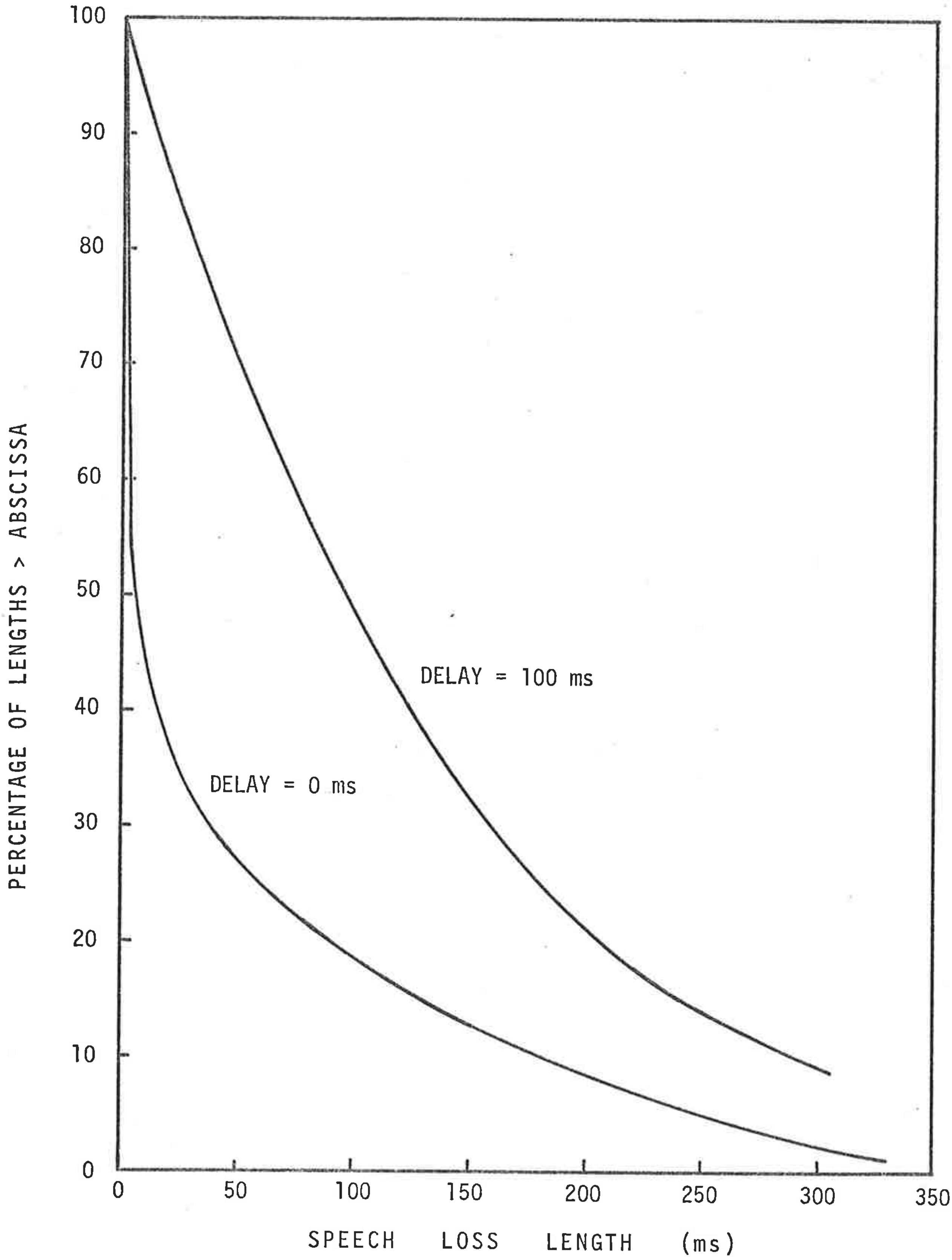


Figure 7.7 Distribution of speech loss length in TASI systems with and without assignment delay

This arrangement might be expected to increase the average delay of the speech and therefore the fraction of speech lost. To confirm this, a run was performed with no restriction upon the change in delay. For the particular case chosen this reduced the average delay from 13.05 to 12.81 ms and reduced the freezeout fraction from 1.085 to 1.076%. Thus imposing the rules on the change in delay does not produce a significant change in the average delay or the speech loss.

It is therefore possible to use delay in TASI systems to reduce the speech loss, with very few ill effects. This is particularly useful if the number of voice circuits available is restricted. In the 40 circuit case if the same delay is used as in the packet system, the TASI system can provide 9% more simultaneous calls.

7.7 Conclusions

It has been shown in this chapter that direct speech interpolation may be employed in mobile telephone systems. This TASI scheme, like the packet one, employs reservation ALOHA with a separate request channel. However it differs from the packet scheme in that voice circuits are provided individually and use essentially real time transmission. The advantage of this arrangement is that existing FM techniques can provide the circuits just as they do in the present mobile telephone schemes.

Occasionally in a TASI system, speech cannot be transmitted because all the speech channels are in use. The lost speech is termed a freezeout and is the equivalent of glitches in a packet system. A theoretical expression for the freezeout fraction is given by (7.3). The freezeout fraction is independent of the talk length distribution but depends heavily upon the size of the system and the activity level of speech. The variation of this quantity with interpolation gain, is similar to the theoretical variation in glitch rate with usage in a packet system.

The TASI system requires a request channel capacity of 30 kbit/s for a C = 40 system. This results in a delay of 24 ms at the start of each talk, which must be allowed for by a fixed delay in the speech path. The delay involved in each request collision is 22 ms and this results in the collision process adding up to 0.19% to the speech loss. The capacity of the request channel is less than that in a packet system because of the significantly longer average times covered by each request.

Simulation of the TASI system provided confirmation of the theory and also showed that the freezeout fraction is slightly greater in practice than the theoretical prediction. Collisions accounted for the majority of this difference. The system interpolation gain is very close to that of the packet system with an assignment delay of 0.2 seconds.

The manner of speech loss in the two systems differs greatly. In the TASI case, speech is lost only from the start of talks which should mean that a given fractional speech loss will be subjectively less important than in the packet case. The TASI system also has the advantage of a smaller delay in the speech path.

In terms of hardware the TASI system radio equipment requirements are the less demanding because they are virtually identical to those of small cell FM systems. However if the TASI system employs analogue transmission the speech detector and delay elements are more difficult to implement than in the digital packet case. The possible means of expansion in the two systems are quite similar. Both can use small cell frequency reuse with the same proviso that the cell size must be reasonably large to support efficient interpolation.

The use of an assignment delay in the TASI system to reduce the freezeout fraction has also been proposed. Such a delay results in an exponential reduction in freezeout with the only major degradation being the delay itself. If the same total delay is used as in the packet case, then a $C = 40$ TASI system has a 9% greater system interpolation gain.

Overall the TASI system is a little more efficient than the packet system and has the advantages of smaller delay and of using existing radio equipment. However if it uses analogue transmission for easy compatability with present technology it loses the advantages of simple speech manipulation and privacy encoding given by digital techniques. If on the other hand, it uses digital transmission, the radio equipment is made more complex and speech quality is lower than that in the packet system.

The great advantage of the packet scheme is that retransmission can be used to overcome radio channel imperfections. Also it may employ complex speech encoding and error correcting techniques because of the non-real-time transmission. It may well be that there is a place for both techniques in the future.

8. PCM AND OTHER ALTERNATIVE APPLICATIONS

8.1 TASI - PCM Systems

Packet speech interpolation has been developed in this thesis within a mobile telephone context. However it may be applied equally well in other areas of telephone communications. The first such application considered here is in pulse code modulation (PCM) telephone networks. In Australia, 30 circuit PCM (CCITT recommendation G732) systems will be employed for inter-exchange transmission in the metropolitan area.

It is possible to more than double the number of simultaneous conversations in PCM systems by using interpolation. This has been recognised by many authors who have suggested various digital speech interpolation techniques [40-48, 104]. Some of these will be discussed later. The main difference between these and the packet technique is the use of delay to reduce the speech loss. Before a packet scheme is fully investigated, the simple TASI approach will be considered since it was slightly more efficient in mobile telephone systems.

A TASI system again operates by providing a circuit only to active sources and only for the duration of their activity. Unlike the mobile telephone situation, in a PCM system all the sources are gathered at one point. Consequently no request channel is necessary to notify the controller of the beginning of a talk. It is necessary however to inform the far end of the PCM line of circuit assignments. This must be done on a separate channel, termed the assignment channel.

In practice the assignment channel is time division multiplexed with the speech on a single PCM line. The capacity required by this channel and the manner of its implementation must be determined. In the present PCM systems there are 30 time division multiplexed voice circuits and two circuits providing synchronization and signalling information. Assume initially therefore that C = 30.

Since the average length of a talk is 1.79 seconds, the maximum average number of talks beginning per second is $\frac{C}{1.79}$. At each new talk it is necessary to send an assignment message denoting the source identity and the number of the circuit to be used. This should require no more than 11 or 12 bits. Because of the importance of correct assignment, this information must be protected by an error correcting code. Thus a message size of 24 bits is appropriate. The required assignment channel capacity is therefore

$$W_s = \frac{24\ C}{1.79} = 402 \text{ bit/s} \qquad (8.1)$$

The major difficulty with this very small channel is finding an efficient way of implementing it within the structure of the PCM format, shown in figure 8.1. The same problem arises with the signalling information for there will be around twice as many input lines connected to the system as previously. There are two basic approaches to this problem. One is to continue with the present format as far as possible and the other involves abandoning at least the signalling portion of the format. In either case the initial frame synchronization word remains unaltered.

Figure 8.1 shows that each circuit has 4 bits per multiframe allocated for signalling. This requires one word of the 32 in each frame, to be virtually dedicated to this task. A simple way of

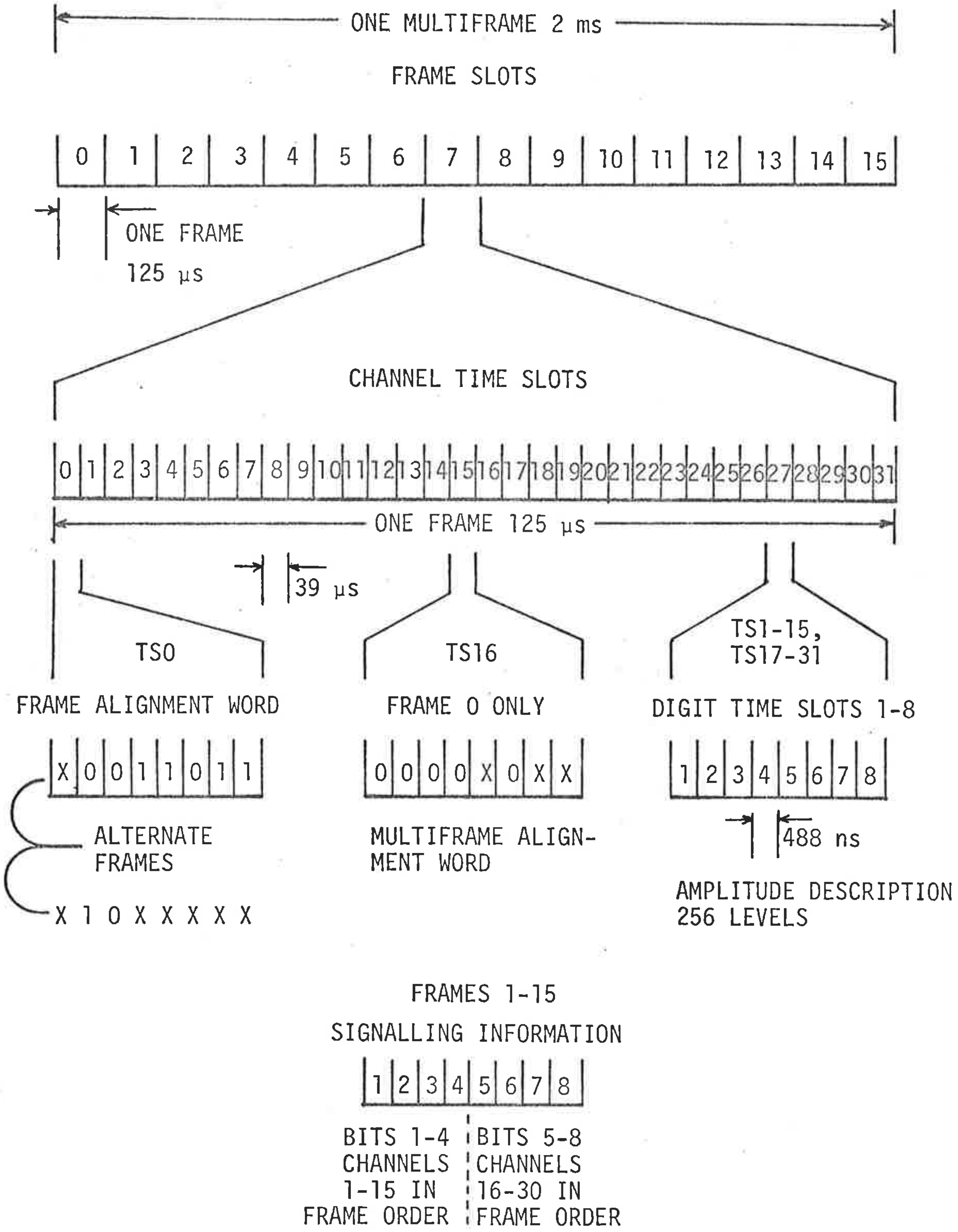


Figure 8.1 PCM frame format

NOTE : Voice channels 1-15 occupy channel time slots 1-15

Voice channels 16-30 occupy channel time slots 17-31

X - unallocated digit, set to 1 when not used providing signalling for a further 30 input lines is to use another word. This reduces the number of voice circuits available for speech from 30 to 29 but provides signalling compatible with present equipment. In addition there is one spare word per multiframe, providing 4 kbit/s, which can be used for assignment messages.

With this arrangement, three multiframes are required for each assignment message. Thus the time for transmission is 6 ms. In addition to this delay, there are delays in decoding the message at the far end and also in detecting the original speech. The total delay between the random speech beginning and a circuit being made available may therefore be up to 18 ms.

This is a fixed delay which occurs to all new talks just as in the mobile telephone case. To avoid speech loss it must be conpensated for by a delay of the same size in the speech path. Now if there is at any time a number of assignment messages waiting for transmission, some will be delayed further and speech will be lost. The situation is, however, quite different from that of the request channel in the mobile telephone case. The assignment queue behaves like the speech queue of the mobile telephone system but with a usage of only 0.1. As a result the fractional speech loss from assignment delays is of the order of 10^{-5} and can be ignored.

Thus in summary, if a compatible approach to signalling is employed the value of C is reduced from 30 to 29 and a delay of around 18 ms must be inserted into the speech path. However, the only source of additional speech loss is the small wastage in the speech channel between a circuit becoming available and being assigned. The fractional speech loss should therefore be much more accurately predicted by the usual TASI theory.

To test this, the system was simulated using a modified version of the mobile telephone TASI programme. All reference to the request channel was removed and the assignment channel was added. The various delays were changed to those above and the system was run with 29 circuits and 60 sources. This resulted in a fractional speech loss of 0.472% compared to a theoretical value of 0.391%.

The main advantages of this method of signalling are its simplicity and compatability with existing equipment. Its main disadvantage is that the number of simultaneous conversations the system can handle is limited to 60 by the availability of signalling channels. To improve the interpolation gain and work at a limit set by the fractional speech loss, a different method of signalling must be employed.

In the above system the signalling information for each call is sampled at a rate of 500 Hz. If, instead, samples are sent only when signalling conditions change, a much lower capacity is required. In fact, if dialling pulses are excluded, the signal lines change only about 5 times throughout the entire call. Even with dialling pulses included, only about 20 changes per second occur in a system providing 75 simultaneous calls, of average length 200 seconds. Coincidentally the number of assignment messages required is also about 20 per second.

It is again possible to combine these two functions onto a single channel. The number of bits required for each signalling message is now more than 4 because the input line concerned must be identified. Also, error correction is necessary since signalling takes place far less frequently and an error has more significance. A suitable message size for both signalling and assignment is therefore 24 bits.

In the present PCM signalling channel there is capacity for 2500 such messages per second, compared to the requirement of 40. Clearly if this channel is used there is negligible delay. Spare message slots can be used for repeated signalling messages so that the average time between adjacent messages is around 35 ms. This arrangement also allows the full 30 voice channels to be used for speech.

Because of the small size of the assignment delay, the total speech delay is little more than the speech detection time and is consequently under 10 ms. Also there is very little wastage in the speech channel when calls change over and absolutely no speech loss due to the assignment process. Thus to a very good approximation the speech loss in such a system is given by the C = 30 theoretical curve of figure 7.1. This was confirmed by simulation.

8.2 Packet PCM Systems

A packet PCM system operates by transmitting speech in packet length intervals. This can either take the form of transmission over the entire channel capacity in a short burst, or of a TDM technique similar to the TASI arrangement. In the first case the packet has to be recorded prior to transmission, thereby incurring the usual recording delay. In the second case however, real time transmission is used and the recording delay is zero. This is equivalent to using a large number of subpackets of length equal to one speech sample.

Clearly the continuous transmission form has the advantage of lower delay. However, if the speech is already in packets for some other reason, then the full packet technique can be used. Apart from the recording delay, the two methods are identical.

Again the request channel in this system will cease to exist and will be replaced by an assignment channel. The number of assignments required and hence the channel capacity depend upon the packet length. A smaller length implies more assignment per second and a greater capacity. However, just as in the mobile telephone system, the packets contain less silence and become more efficient. Thus the same tradeoff exists as before and there is an optimum packet length which minimizes the full channel interpolation gain.

The assignment channel capacity is (from (3.3))

$$W_s = \frac{A\ B\ C}{T} \tag{8.2}$$

Here call set up and termination messages have been ignored since they are handled by the signalling channel.

Now the number of bits in an assignment message can be the same as for the TASI system, giving $B = 24$. Similarly it may again be assumed that $C = 30$. In the mobile telephone system A is the ALOHA factor which ensures a low usage in the request channel. There is no need for a large value of A here since no collisions are possible. The assignment messages simply queue for transmission in the same manner as the speech packets themselves.

The correlation between these two queues is very high since there is one assignment for each packet. As long as the assignment channel usage is lower than that of the speech channel there can never be more assignments waiting than packets. Thus whenever the queues are long, an assignemnt is transmitted before the corresponding packet. The maximum delay introduced by this channel is one message transmission time when both of the queues are empty. Bacause of this a value of 2 for A should be sufficient.

In a PCM system the bit rate of digitized speech is D = 64 kbit/s. With this value and the product AB at 50, figure 3.11 shows how the optimum packet length is well below 0.1 seconds. The actual length, calculated by graphical means, is 55 ms and the corresponding assignment channel capacity is 27 kbit/s. The capacity provided by the original PCM signalling channel is 60 kbit/s and therefore this channel may be used for both assignment and signalling, as in the TASI case.

Since the capacity available is greater than that required, the constraint on channel provision caused by maintaining the PCM format has reduced the potential efficiency. To correct this, the packet length should be reduced to increase the interpolation gain and make better use of the assignment channel. In fact, with the above figures, a packet length of 24 ms is possible. The resulting delay on the signalling messages can be quite safely ignored since the A factor ensures that the channel operates at a usage of no more than 50%.

In the simulation of this system a 50 ms packet length was used because of the significantly greater simulation times required with a smaller packet length. This results in very little change to the system interpolation gain. The standard mobile telephone programme was used with the request channel removed and with a maximum assignment delay of 50 ms.

The simulation results and those for the TASI system of the previous section appear in figure 8.2. Because the systems use exactly the same assignment channel, they can be compared simply upon the basis of their interpolation gains. The packet system, in this case, is slightly superior. The improvement relative to the position in the mobile telephone system is due to the smaller packet length. This in turn results from the small size of assignment messages compared to request packets and from the change in the ALOHA factor. The actual number of conversations that a packet system can provide with a fractional speech loss of 0.5% is 67.

A delay may again be used with the TASI system to reduce its speech loss. Such an arrangement was simulated and the results have already been presented as the C = 30 curves in figure 7.6. When it is recalled that the C = 40 curve in this figure was produced by the mobile telephone TASI system, the generality of the conclusions drawn regarding the delay can be fully appreciated.

With a maximum of 50 ms delay allowed in the TASI-PCM system, the speech loss is reduced to that shown by the dashed curve in figure 8.2. Notice that over the operating range of interest the packet system is superior even though the two have the same delay. This occurs because the delay is better distributed in the packet system with its smaller units of transmission. The improvement is sufficient to overcome the slight loss in efficiency resulting from the use of fixed length packets.

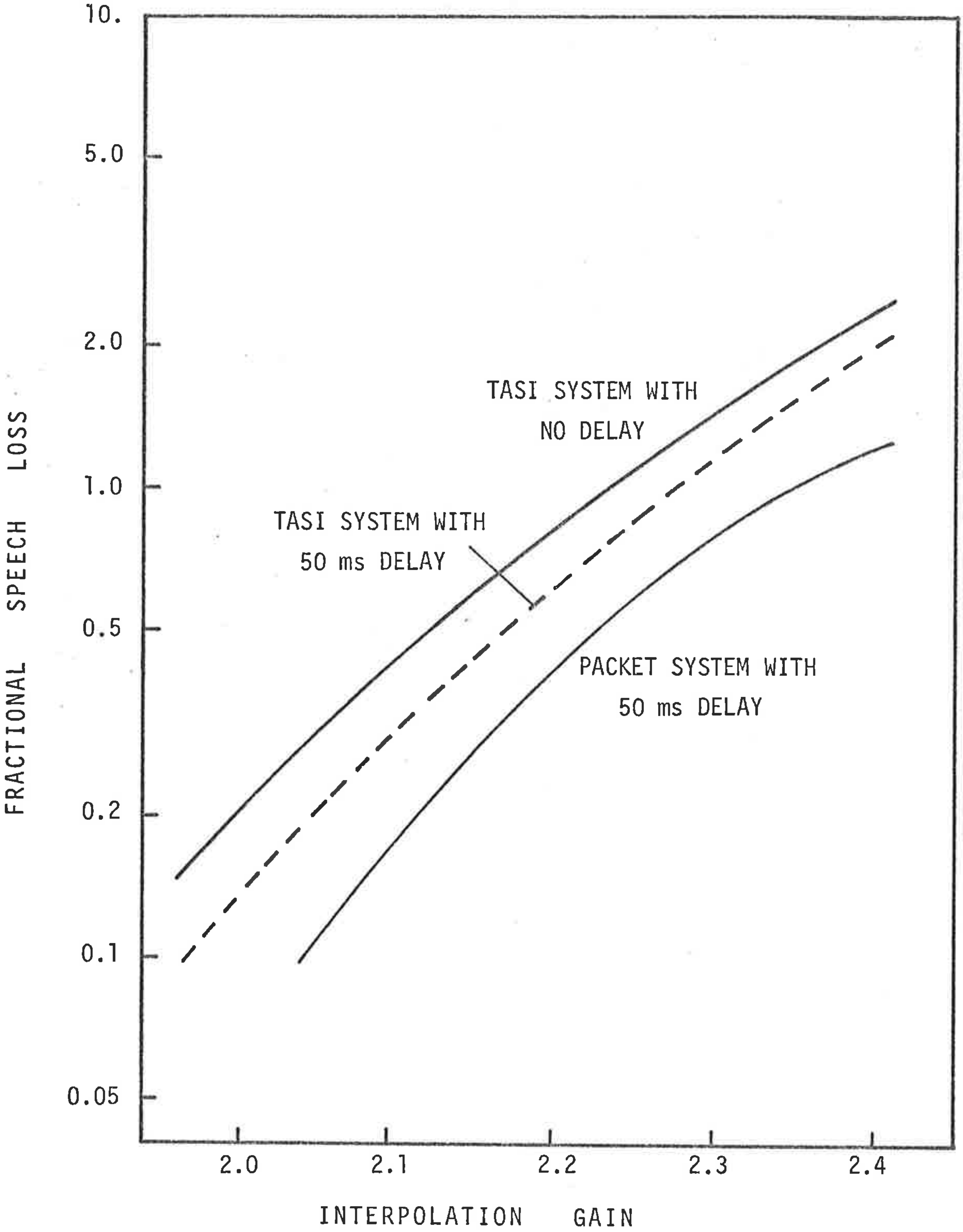


Figure 8.2 Comparison of TASI and packet PCM systems

8.3 Alternative Digital Speech Interpolation Methods

Many different techniques have been suggested for the interpolation of speech in PCM systems. The simple TASI scheme has been proposed in many forms [41,44]. However, the aim of most authors is to avoid the speech loss through freezeout that characterizes TASI. The methods used to achieve this fall into two broad categories :- bit reduction and adaptive encoding.

Bit reduction entails changing the number of bits used to encode each speech sample whenever there are more active sources than circuits. A reduction from 8 bits to 7 provides 4 extra channels within the standard PCM system. If this is insufficient the number of bits may be reduced to 6. These are times even then when the channel is overloaded and clipping occurs. However, it has been shown that the fractional speech loss may be reduced by an order of magnitude with this technique [42].

With standard PCM encoding there is a 6 dB reduction in the signal to noise ratio (SNR) for each bit lost. This reduction occurs for the same time as the freezeout would have lasted but effects eight circuits instead of one. Thus the SNR loss in any one circuit occurs far more often than a freezeout would, but the subjective effect may still be an improvement. Bit reduction can be used in the same manner in the packet scheme. The subjective gain should be just as great since the delay and the SNR reduction will not interact.

It is possible to avoid freezeout completely by continuing to reduce the number of bits per sample. To maintain acceptable speech quality at low bit rates however, more advanced encoding techniques must be used. For instance, sample sizes down to 3 bits can be tolerated by adaptive PCM. Thus some form of adaptive quantizer must be used [45], resulting in extra cost and complexity.

The last proposal is an example of the second technique of avoiding freezeout. This is also embodied in the speech encoding communications (SPEC) system [44,48]. Here samples taken from each source are transmitted only if they differ from the previous sample by more than a certain threshold. No speech detector is required because if there is no input, the level is constant and no samples are transmitted. Overload is avoided by increasing the threshold to ensure that the number of transmissions always falls within the channel capacity. Thus in this case, freezeout is replaced by an increase in the quantization noise.

When more complex speech encoding techniques are used with a SPEC arrangement, greater efficiency is attained [47]. Woitawitz [46] proposes such a system, capable of handling 78 subscribers on a 30 circuit PCM line with no freezeout and a SNR of 32 dB. This compares with a 38 dB SNR for normal PCM at 64 kbit/s.

A limiting factor with certain SPEC type systems is that one assignment bit is required for each input source in each frame. This results from the need to define the sources from which the samples originate. Thus up to 25% of the capacity is dedicated to assignment. To reduce this the samples must be transmitted in blocks (i.e. in packets).

Advanced speech encoding techniques may be used directly in the TASI and packet systems. A SNR very close to that of PCM may be provided at substantially lower than 64 kbit/s. This effectively increases the number of voice circuits available and enables an increase in the number of calls connected, or a decrease in the speech loss, or some combination of both.

Thus a packet system can be designed which will match the service provided by advanced systems. The former also makes possible a reduction in the speech degradation due to overload by increasing the delay beyond the 50 ms used previously. Another advantage of the packet system is its relative ease of implementation and its compatability with present PCM techniques.

The memory and control circuitry required by a packet system are fairly straightforward and an increase in the allowable delay involves only an increase in the memory size. However adaptive quantization of speech sources, selection of appropriate samples for transmission and complete system revision every frame interval, is much more demanding. The hardware requirements of the packet system should therefore be somewhat less costly and complex than those of the equivalent SPEC type system.

8.4 Interpolation in Satellite Channels

Telephone connections via satellite have traditionally employed frequency modulation and frequency division multiplexing. Satellite capacity is split between the various ground stations by frequency division multiple access (FDMA). However, more telephone circuits can be provided if digital transmission is used with time division multiple access (TDMA) [105-107]. This occurs because in FDMA systems, intermodulation between multiple carriers restricts the capacity. There is therefore a move toward digital satellite transmission as exemplified by the proposed Japanese domestic system [107,108].

In the majority of suggested TDMA schemes the channel is split into frames which include a preamble, the data and an error correcting code. The simplest way of allocating channel capacity to individual ground stations is to preassign to each, certain frames, or certain parts of every frame. More advanced schemes have also been suggested in which the assignment is dynamic, and techniques such as reservation ALOHA are used [16,18,109,110].

A problem with such schemes is the delay involved in transmitting requests back and forth over a satellite channel. This can be overcome if each ground station is preassigned a certain fraction of the capacity, and requests more whenever the average transmission delay becomes too large.

While it is quite possible to interpolate speech at this level, a simpler method is to use interpolation in PCM type streams. These can then be multiplexed with other information for transmission within the frame structure. An advantage of a satellite channel is the high capacity it permits. This allows more speech circuits to be included in each interpolation system.

The importance of the number of circuits, C, has been shown previously. In the TASI-PCM system if C is increased from 30 to 60 the freezeout fraction is reduced from 0.5% to 0.008%. Alternatively the interpolation gain may be increased by 8.5%. The equivalent packet system provides just as much gain as can be seen from figure 5.10. An additional bonus is that at the same time the assignment delay is halved (from (5.8)).

There is of course nothing preventing the use of even larger numbers of circuits. Though the interpolation gain is limited by the average speech activity, the delay can be made insignificant. This certainly means that the packet scheme will outperform a TASI one both in the number of simultaneous conversations provided and in the fractional speech loss. The delay introduced in a packet system can be kept below 30 ms and is completely dwarfed by the transmission delay.

8.5 Integrated Speech and Data Networks

Speech in packets is naturally accomodated in an integrated speech and data network. This type of network will become more common as the amount of computer communication increases. It is natural to combine speech and data channels to avoid duplication of transmission facilities, and digital techniques, with TDMA, must inevitably be employed [12,45,109].

Such wide bandwidth networks consist of interconnected nodes acting as sources and sinks of data or speech. Each node is a concentration or multiplexing point for several smaller sources such as computer terminals and voice lines. Packets for any particular destination must be collated and ordered for transmission in sequence. This is a perfect framework for packet speech interpolation.

If data and speech packets are handled in the one system there will be gains in efficiency due to the more random nature of the input. Correlation between speech packets is the main reason for the glitch rate being higher than theoretically predicted in the mobile telephone system. When speech forms only a portion of the total input the queue correlation will be reduced and the glitch rate will fall.

If speech packets are given priority over data packets then the channel usage is effectively reduced. Even if speech constitutes 70% of the total input the effective usage is so low that very small delays and virtually no glitches will result. This arrangement amounts to transmitting data in the breaks between speech and will result in substantial delays for the data. The critical aspect for data however is the fractional packet loss, not the delay.

In practice, speech will not be given complete priority because a glitch rate of 0.1% provides quite adequate quality. Thus a dual system will operate, with a low maximum delay and a reasonable glitch rate for speech and a high maximum delay and very low glitch rate for data.

Consider a 60 circuit system with nominally half of the capacity dedicated to speech and half to data. The interpolation gain achievable for speech will be greater than that of a $C = 60$ system because of the different nature of the data input. If the data arrives in a purely random fashion, theory shows that a $C = 30$ system can provide a glitch rate of 10^{-10} for data usages of 75% at a maximum delay of 0.5 seconds.

If there is any correlation in the data sources the glitch rate will be somewhat higher. For instance, the glitch rate under the same circumstances with a speech input, is 10^{-5}. In the combined speech and data system the effect of any correlation will be reduced and the data usage can be increased.

An application similar to this in principle is the high capacity subscriber loop. Such a system conveys two way information between a major distribution centre and many subscribers via a single high capacity line. Potential services include newspaper distribution, child and adult instruction, various shopping and purchasing operations, message and mail delivery and interaction with library, computer and work facilities [111].

This system involves a subscriber transmitting a request for desired information or for appropriate transmission capacity. Voice communication can be handled easily as the arrangement is remarkably similar to the original radiotelephone system. Packet TDMA would provide the most efficient use of resources though other arrangements are possible [112]. Again the high capacity ensures that packet speech interpolation can be used with very little delay and negligible speech loss.

8.6 Conclusions

In this chapter several alternative applications of packet speech interpolation have been explored. The first was in PCM telephone links for which many interpolation schemes have already been put forward. This popularity is due to the ease of implementing speech interpolation in digital networks. The simplest arrangement is a TASI scheme similar to that of the mobile telephone system but with the request channel replaced by an assignment channel.

It is possible to provide an assignment channel within the existing basic PCM frame format (figure 8.1). This involves modifying the signalling channel so that only changes in the signalling conditions are transmitted. The channel can then be used for both signalling and assignment with any spare capacity being filled with redundant signalling messages. Because of the small message assignment transmission time and the extremely low usage on this channel, all sources of speech loss, except for overload freezeout, disappear. The loss is therefore given exactly by the theoretical model of chapter 7.

In the packet PCM system there is again a tradeoff in the determination of the packet length and an optimum length may be found. However, there is little loss involved in using the same assignment channel as for the TASI case. The simulation of a packet system with 50 ms maximum queue delay showed that in this case the packet system outperformed the equivalent TASI one. This reversal from the mobile telephone situation is due to the low assignment channel requirement permitting a smaller packet length and hence a more efficient operating position.

Other PCM interpolation schemes were considered. The techniques they employ for increasing efficiency can also be applied in the packet system. The essential benefit of most of these schemes is a partial or total elimination of the speech loss, in favour of a temporary reduction in the signal to noise ratio. Overall the difference in performance between these schemes and the packet one is not significant and the implementation of the packet scheme should be easier.

The second application proposed was in the field of satellite transmission. Here the great capacity available gives packet schemes a real advantage, for as well as providing better performance than a TASI system, a packet system has negligible delay.

Finally, packet speech interpolation was considered in intergrated communication networks. This is the most obvious area of application since the packet format is already in use. Mixing speech and data has the advantage of reducing input correlation and increasing performance toward the theoretical limit. It should be quite possible within a capacity of 4 Mbit/s to run a speech system with extremely small delays and subjectively negligible speech loss, in conjunction with a data system with virtually no packet loss and delays under 1 second. An illustration of such a system is the high capacity subscriber loop service.

9. CONCLUSIONS AND FURTHER WORK

In the search for more efficient methods of transmitting speech, interpolation stands out as one of the simplest and cheapest solutions. This is even more true for digital systems than for analogue. The field of mobile telephony is one where such efficiency is required because of the limited radio spectrum available. For this reason the practicability of digital speech interpolation in mobile telephone systems has been studied.

From an investigation of the nature of radio propagation in the urban environment, it is apparent that the high capacity channel required by such a system presents an immediate difficulty. However, with parallel transmission in a number of channels, bit rates in excess of 1 Mbit/s are theoretically possible. If, in conjunction with this, speech encoding techniques are used which provide appropriate quality speech at bit rates of 25 kbit/s, then an efficient interpolation system is practical. It is under these assumptions that the research described herein was undertaken.

In the mobile telephone system designed, a packet technique is employed with reservation ALOHA. The packet length effects the efficiency of the system both directly, in terms of the silence contained within the packet, and also in terms of the reservation or request channel capacity required. These aspects may be combined to produce the full channel interpolation gain; an initial figure of merit. In any given system this quantity is maximized at a particular packet length, but this does not necessarily optimize the system.

There are certain delays which arise in packet speech interpolation, the two most important of which are the packet recording time and the wait for transmission slot assignment. When the latter is restricted to a certain maximum value there results some speech loss due to temporary overload. This has been investigated both theoretically and by simulation to determine methods of minimizing the loss and its effects.

To keep the speech loss below a desired level with a fixed assignment delay, the rate of input to the system must be limited. This limit changes with the packet length and may be combined with the original figure of merit to produce the system interpolation gain. This is the final measure of capacity of a given system subject to a set maximum delay and speech loss.

Because of various conflicting effects it eventuates that the system interpolation gain is almost independent of the packet length over a wide range. Nonetheless, an optimized system can be designed, based upon the two previously mentioned assumptions. This has a maximum fractional speech loss of 0.5% and a maximum round trip delay of 0.3 seconds. It is able to provide 87 two way conversations within a total radio channel capacity of 2.16 Mbit/s and can service 4100 subscribers.

The most modern alternative technique of providing mobile telephony is the small cell scheme. Compared to this the packet system has many advantages arising from its digital nature and more efficient arrangement. The advantages of the small cell scheme are its camparative simplicity, low cost and its ability to reuse voice circuits many times. Such reuse is also possible in packet systems, although there is some loss in the interpolation gain.

The use of packet speech interpolation in mobile telephone systems is, overall quite feasible. However, in the near future simpler methods such as the FM small cell systems will be used. Interpolation may be incorporated into these schemes through a time assigned speech interpolation (TASI) system which has advantages over the packet system of simplicity, lower cost and marginally greater efficiency. In the future, though, as the telephone network is digitized, the packet system will become cheaper and more compatible. Then the advantages it provides in transmission quality and privacy will make it a strong contender for mobile telephone use.

There are more immediate prospects for the use of packet speech interpolation in other areas. In applications such as PCM telephone lines, satellite, and integrated data networks, packet systems can outperform their TASI counterparts. The investigation of these alternatives showed that the same basic principles apply as in the mobile telephone system. It may therefore be justly claimed that the information presented in this thesis, though centred on mobile telephone use, is equally applicable to many other areas.

Because of the wide scope covered here several topics have only been considered briefly. A great deal more study is required on the provision of high capacity digital channels in the urban environment. Other aspects of the mobile telephone system hardware also need further investigation. Two examples are the provision of echo suppressors and the communication between the repeating stations and a central controller. More comprehensive speech trials are also required to better define the subjective effects of glitches. Applications other than mobile telephony have their own particular problems.

It is hoped that the information presented in this thesis will prove to be of use in the implementation of packet speech interpolation.

APPENDIX A Determination of the correct queue arrangement

Consider the timing diagram shown in figure A.1. Each slot interval has a length equal to the transmission time of one packet. The packet transmission queue may therefore be superimposed upon this diagram with each packet (or request) occupying one slot. Requests arrive at the queue at a random time and are placed into the first empty slot. The earliest that a request can be serviced is in the first slot beginning after its arrival. Thus there is a random delay of between zero and one slot length. This delay does not form part of the actual slot assignment or queueing delay in a discrete time system and, for the remainder of the analysis, it will be ignored.

In queue theory an entry is assumed to be serviced instantaneously either at the end, or start, of a slot. These two alternatives will be considered in turn.

Assume initially that service occurs at the end of a slot and that the queue length is measured at the start of a slot [40,50,92]. Then a request arriving during a slot can be serviced before the queue length is determined. Thus a request may nominally be serviced in the slot in which it arrives.

Now consider this arrangement in terms of delay when a request arrives in slot "n" as shown in figure A.1. If the queue is empty when the request arrives it is served immediately, there is no delay, and the queue length measured after the service is zero. If there is already one entry in the queue the length will remain at one when this entry has been serviced. The delay until the new request's

Figure A.1 Timing diagram for the request queue

service begins (at the end of slot n+1) is one slot length. Similarly, in any other case, a request experiences a delay equal to the queue length in its arrival slot.

If there are the maximum number of L entries already in the queue, a request suffers a delay of L slots, which is the maximum amount possible. If in this case two requests arrive in slot n, the second cannot be transmitted until the end of slot n+L+1 and hence must be glitched. The queue length for this second request is L+1, i.e. greater than the allowed limit.

Now consider the case where transmissions nominally occur instantaneously at the start of a slot and the queue length is determined at the end of a slot [51,53]. This prohibits a request being serviced in the slot in which it arrives. The queue length includes all new requests arriving during a slot and also the request about to be serviced.

If the queue is empty when a request arrives in slot n, the queue length measured at the end of the slot is one. Yet the request can be serviced in slot n+1 with zero delay. Similarly, for any other condition the queue length is one greater than the delay suffered by the latest arrival. If there are L entries in the queue the length will be given as L+1 though the new request can be serviced in slot n+L+1 with a delay of only L.

Thus to achieve the same maximum allowable delay in the second case, the queue limit must be one slot greater than in the first case. Also, despite the apparent difficulty of transmitting a request in the slot in which it arrives, this theoretical arrangement results in a queue length limit equal to the desired delay limit.

Theoretical values based upon both cases were computed and checked against a simulation of the actual queue system. The results, shown in table A.1, confirm the above reasoning and show that in practice the first case should be used.

Usage %	100	96	92
Case 1, L = 20	2.344	0.898	0.268
Case 2, L = 20	2.459	0.992	0.318
Case 2, L = 21	2.344	0.898	0.268
Simulation, L = 20	2.33	0.905	0.276

Table A.1 Comparison of glitch rates

APPENDIX B Queues with dependent inputs

The state probabilities of queues with dependent inputs may be calculated by a method due to Gopinath and Morrison [53]. Because of the complexity of this technique the single input, uncorrected case will be considered first to establish the methodology.

Denote the state of the queue (i.e. the number of queue entries) at time slot "n" by "b_n" and the number of inputs in slot n as "x_n". Then the state of the queue at the next slot is (using case 2 of appendix A)

$$b_{n+1} = b_n - 1 + x_n \qquad \text{if } b_n \geq 1$$
$$= x_n \qquad \text{if } b_n = 0$$

or $$b_{n+1} = (b_n - 1)^+ + x_n \tag{B.1}$$

where $(z)^+$ denotes the maximum of z and 0.

The probability distribution of arrivals is defined by

$$\text{Probability } [x_n = i] = p_i \tag{B.2}$$

After the queue has operated for sufficient time, and if it satisfies the conditions of a Markov process, the probability of various values of b_n tend toward steady state values denoted by

$$\text{Probability } [b_n \overset{n \to \infty}{=} i] = \pi_i \qquad i = 0,1,\ldots \tag{B.3}$$

Relationships between the π_i may be obtained by considering transition probabilities. This approach provides solutions based upon the recursive expression

$$\Pi_i = \frac{1}{p_0} \left(\Pi_{i-1} + p_i \Pi_0 - \sum_{j=1}^{i} p_j \Pi_{i-j} \right) \qquad i \geq 1 \tag{B.4}$$

where $$\Pi_i = \sum_{j=0}^{i} \pi_j \tag{B.5}$$

Further, if there is a queue length limit of L

$$\Pi_L = 1 \tag{B.6}$$

The solutions provided by (B.4) and (B.6) are exactly the same as those derived in section 4.2.

Now consider the simplest possible example of a queue with dependent inputs. Each input results in an addition to the queue in the slot in which it arrives and another addition in the subsequent slot. The state of the queue is then given by

$$b_{n+1} = (b_n - 1)^+ + x_n + x_{n-1} \tag{B.7}$$

In this case the simple approach above is invalid because the input in slot n depends upon the input in the previous slot, n-1. However, a two dimensional approach may be adopted. Two state variables, b_n and z_n, are required with

$$z_n = b_n + x_{n-1} \tag{B.8}$$

The states (b_n, z_n) form a two dimensional Markov process and the steady state probability distribution is defined

$$\text{Probability } [b_n \overset{n \to \infty}{=} i \text{ and } z_n \overset{n \to \infty}{=} j]$$

$$= \pi_{i,j} \qquad i = 0,1,\ldots \quad ; \quad j = 0,1,\ldots \tag{B.9}$$

The actual probabilities required are those of the queue length, b_n. These are the marginal state probabilities.

$$\text{Probability } [b_n \overset{n \to \infty}{=} i] = \omega_i \tag{B.10}$$

A recursive relationship along the lines of (B.4) can be formulated when there is no limit on the queue length. However, if there is a limit, L, the relationship breaks down since there may be an infinite number of states $\pi_{L,j}$ $(L \leqslant j < \infty)$. The expression for ω_L contains certain of the $\pi_{i,j}$ as well as other marginal state probabilities and is

$$\omega_L = \frac{1}{p_0} \left[(1-p_0)\,\Omega_{L-1} + \pi_{L,L} - \sum_{i=1}^{L-1} \left(\Omega_\alpha + \sum_{k=0}^{\alpha} \sum_{j=0}^{k} \pi_{L-i-k,\, L-i-k+j} \right) \right] \qquad (B.11)$$

where $$\Omega_i = \sum_{j=0}^{i} \omega_j \qquad (B.12)$$

and $$\alpha = \frac{L-i-1.5}{2} - \frac{(-1)^{L-i}}{4} \qquad (B.13)$$

The steady state probabilities and glitch rate were determined from the above expression and were checked by simulation. In addition, this system, with each input causing an entry in two adjacent time slots, is theoretically equivalent to a queue with a single entry per arrival and a limit of half the size. The glitch rate of the latter was also calculated and is compared with those from the other two approaches in table B.1.

Multidimensional theory	21.410	15.000	7.895
Simulation of dual entry	21.95	15.062	7.900
Theory of single entry half limit	21.410	15.000	7.895

Table B.1 Confirmation of multidimensional queue theory

It is clear from this example that in general the individual state probabilities will be required to obtain the glitch rate in a queue of limited length. While it is not necessary to calculate an infinite number of state probabilities, the number can become very large.

The dimension of each state is the number of slots over which correlation occurs. In a queue with M subpackets per packet and C slots between the entry points of adjacent subpackets, the correlation length is (M-1)C. In each dimension there are L+1 possible values. Thus for example if M = 10, C = 40 and L = 20 there are 10^{476} different states. Not all of these state probabilities need be calculated but the problem is none the less intractable.

APPENDIX C Simulation programmes and flowcharts

Included here are the flowcharts for the speech simulation programme and the full mobile telephone system simulation. The speech generation programme is in the form of a subroutine with several small attached subroutines which accumulate various statistics.

In the main simulation programme the following arrays are used.

CALLS (N,7) - holding the current status of each of the N attached sources

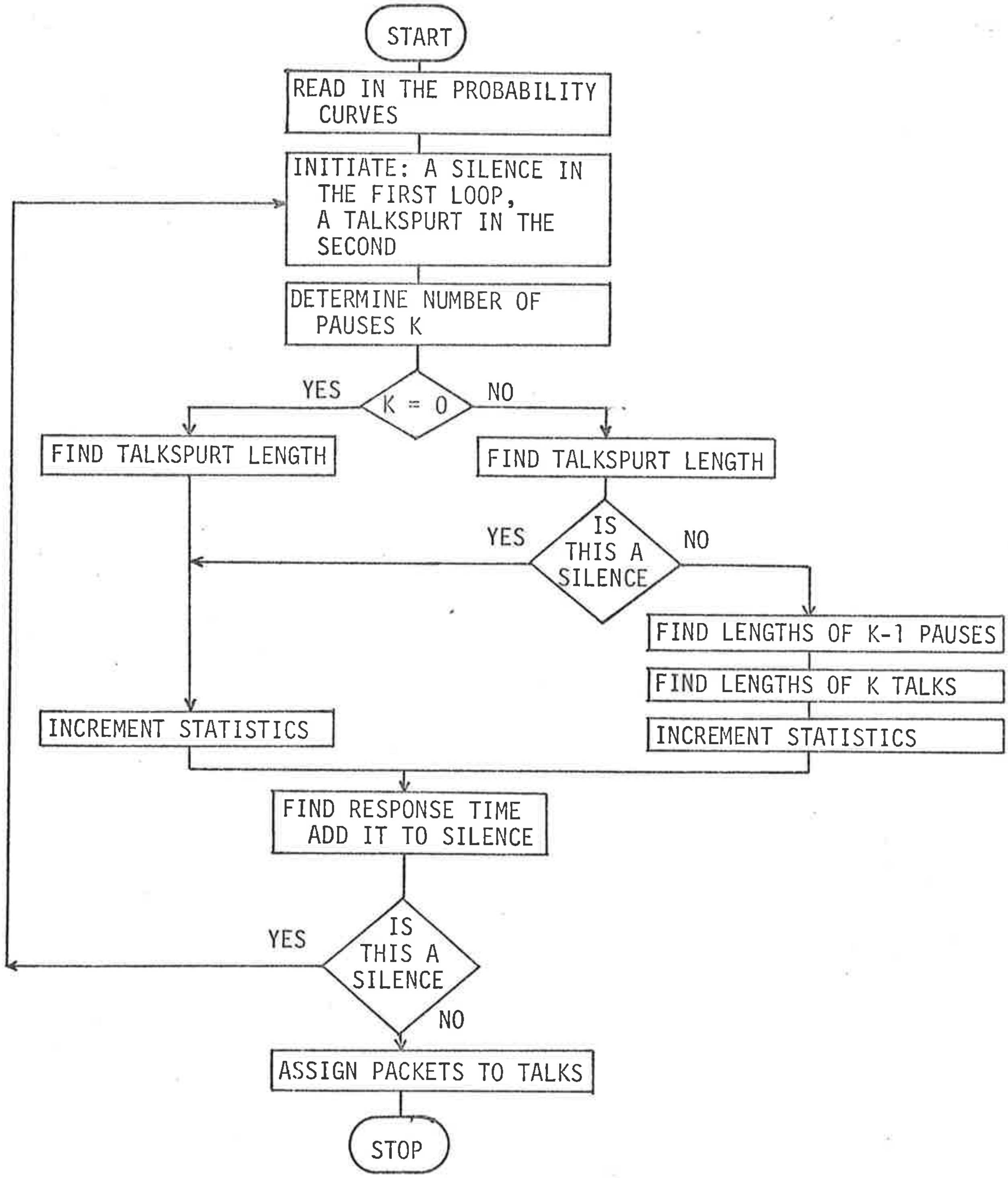
PACK (800,3) - holds further packet information for all calls

ISLOT (800,4) - the queue holding subpackets for transmission.

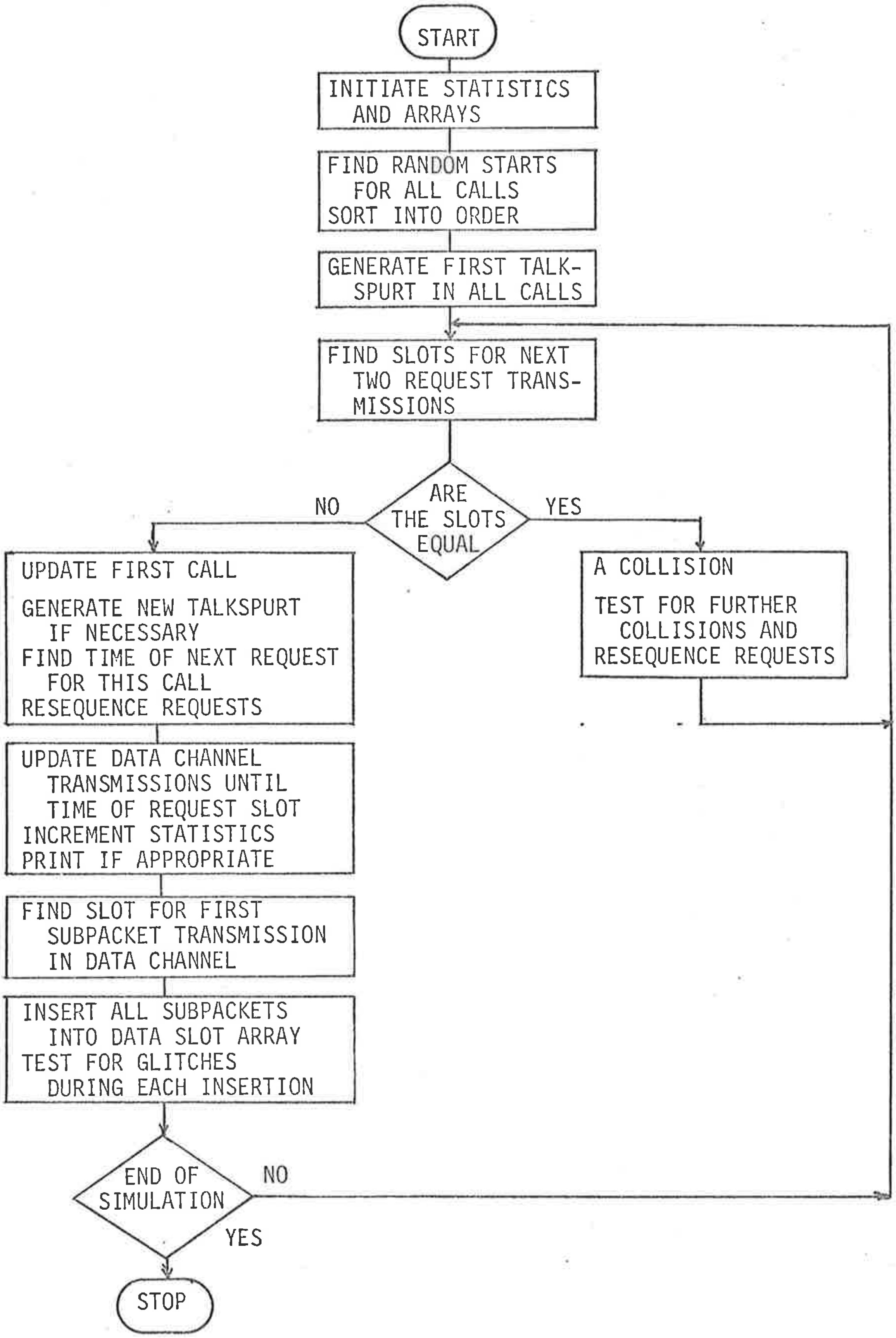
These arrays hold the following details.

CALLS (X,1)	-	starting time of the current talkspurt in call X
(X,2)	-	number of packets in the current sequence
(X,3)	-	address in PACK of next packet sequence for call X
(X,4)	-	starting time of next talkspurt
(X,5)	-	ending time of conversation for call X
(X,6)	-	number of the request slot for the next packet
(X,7)	-	number of next call in sequence of request transmission.
PACK (Y,1)	-	starting time of packet sequence Y
(Y,2)	-	number of packets in sequence Y
(Y,3)	-	address of next packet sequence in PACK for the appropriate call or zero if this is the last sequence in a talkspurt.
ISLOT (Z,1)	-	slot priority of the subpacket in slot Z
(Z,2)	-	number of the call from which the subpacket originates
(Z,3)	-	number of the subpacket within the packet
(Z,4)	-	address in ISLOT of next subpacket in sequence.

FLOWCHART FOR SIMULATION OF SPEECH IN PACKETS



FLOWCHART FOR SIMULATION OF MOBILE TELEPHONE SYSTEM



```
      PROGRAM SIMUL (DATA,OUTPUT,TAPE1=DATA,TAPE2=OUTPUT)
C   THIS PROGRAM SIMULATES A MOBILE TELEPHONE SYSTEM
C   WITH PARAMETERS AS SET OUT BELOW.
      COMMON CALLS(100,7),PACK(800,2),IFREE(800),ISLOT(800,4)
      COMMON TL0(25),TL1(21),TL2(21),TL3(21),TL4(21)
      COMMON FLEN(24),CLEN(21),TTLR(21),TALK(101),PAUSE(100)
      COMMON/1/ITT(31),NT,ST/2/IPT(31),NP,SP
      COMMON/3/IST(31),NS,SS/4/IPW(11),SPIP,SPACE,T
      COMMON/5/IPD(11),ND,NFA,NFB,NC,IFF,NFT,IFPT,IFTPT
      DIMENSION ICA(30),IPOINT(11),DELY(10),ISTART(10,2)
      INTEGER C
      READ (1,1) CLEN
1     FORMAT(10F4.0/11F4.0)
      READ (1,2) TTLR
2     FORMAT(10F6.0/11F6.0)
      READ (1,3) TL0
3     FORMAT(10F6.0/10F6.0/5F6.0)
      READ (1,4) TL1
4     FORMAT(10F6.0/11F6.0)
      READ (1,5) TL2
5     FORMAT(10F6.0/11F6.0)
      READ (1,6) TL3
6     FORMAT(10F6.0/11F6.0)
      READ (1,7) TL4
7     FORMAT(10F6.0/11F6.0)
      READ (1,8) FLEN
8     FORMAT(12F6.0/12F6.0)
C
C   SET PARAMETER VALUES
      C=40
      L=20
      T=0.4
      M=4
      LIMIT=L*M
      N=95
      A=5.
      K=5

C   INITIATE STATISTICS
      DO 9 I=1,31
      ITT(I)=0
      IPT(I)=0
9     IST(I)=0
      NT=0
      NP=0
      NS=0
      ST=0
      SP=0
      SS=0
      NPT=0
      DO 10 I=1,11
      IPW(I)=0
10    IPD(I)=0
      SPACE=0.
      SPIP=0.
      AAVD=0.
      CAVD=0.
      NCLIT=0
      NREQR=0
      DO 11 I=1,800
11    IFREE(I)=I
```

```
      NFA=1
      IFP=1
      IFPT=1
      IFTP=1
      IFTPT=801
      IGLITT=0
      RAN=RANF(0)
      IPENT=0
      AVD=0.
      AV1=0.
      AVGDN=2000.
      LAP=0
      LAP1=600
      LASLOT=0
      NFAST=20000
      NP20=0
      GP20=0.
      IAVGDN=AVGDN
      WRITE(2,12) C,L,N,T,M,IAVGDN
12    FORMAT(1H1//10X37HSIMULATION OF MOBILE TELEPHONE SYSTEM
     */10X37(1H-)//10X19HSYSTEM CAPACITY C = I3,
     *18H   QUEUE LIMIT L = I3,22H   NUMBER OF CALLS N = I4
     *//10X17HPACKET LENGTH T = F8.3,13H   NUMBER OF
     *25HSUBPACKETS PER PACKET M = I3//10X16HNUMBER OF SLOTS
     *30HOVER WHICH DELAY IS AVERAGED = I5//21X5HAVERAGE
     *5HDELAY 25X21HCURRENT    NO. SUBPK. /60X10HSLOT NO.
     *11HTRANSMITTED /)
      DO 13 I=1,600
      ISLOT(I,4)=I+1
13    ISLOT(I,1)=0
      ISLOT(600,4)=1
      DO 14 I=1,N
C
C  FIND STARTING TIME OF ALL CALLS
      CALLS(I,1)=RANF(0)*20.
14    CALLS(I,4)=CALLS(I,1)
C  SORT TIMES INTO ORDER
      DO 16 I=1,N
      KE=0
      DO 15 J=2,N
      IF(CALLS(J,1).GE.CALLS(J-1,1)) GO TO 15
      TS=CALLS(J-1,1)
      CALLS(J-1,1)=CALLS(J,1)
      CALLS(J-1,4)=CALLS(J,1)
      CALLS(J,1)=TS
      CALLS(J,4)=TS
      KE=KE+1
15    CONTINUE
      IF(KE.EQ.0) GO TO 17
16    CONTINUE

C  GENERATE FIRST TALKSPURT IN ALL CONVERSATIONS
17    DO 18 I=1,N
      NC=I
      CALL GEN
      IAD=CALLS(I,3)
      CALLS(I,2)=PACK(IAD,2)
      CALLS(I,3)=PACK(IAD,3)
      IFREE(IFTP)=IAD
      IFTP=IFTP+1
      IFTPT=IFTPT+1
      IF(IFTP.EQ.601) IFTP=1
      IF(CALLS(I,3).EQ.0.) CALL GEN
      CALLS(I,6)=CALLS(I,1)+A*C/T
```

```
18      CALLS(I,7)=I+1
        CALLS(N,7)=0.
        NR=1
        IPOINT(2)=2
        DO 19 I=3,10
19      IPOINT(I)=(I-2)*N/10
        IPOINT(11)=N

C   FIND TRANSMITTED REQUESTS
20      NC=NR
        NR1=NR
        ICA(1)=NC
        KSLOT=CALLS(NR,6)+1.
        NR=CALLS(NR,7)
        I=1
        ITS=CALLS(NR,6)+1.
        IF(ITS.NE.KSLOT) GO TO 28
C   COLLSION
        DO 26 I=1,30
        ICA(I)=NR1
C   RESEQUENCE REQUESTS
        CALLS(NR1,6)=CALLS(NR1,6)+2.+0.012*A*C/T+K*RANF(0)
        IF(CALLS(NR1,6).GT.CALLS(NR,6)) GO TO 21
        NR=NR1
        GO TO 25
21      IPOINT(2)=CALLS(NR,7)
        IPOINT(1)=NR1
        NR2=NR1
        DO 22 IRS=2,11
        IF(CALLS(NR1,6).LT.CALLS(IPOINT(IRS),6)) GO TO 23
        IPOINT(IRS-1)=CALLS(NR2,7)
22      NR2=IPOINT(IRS)
        IPOINT(11)=NR1
        CALLS(NR2,7)=NR1
        CALLS(NR1,7)=0
        GO TO 25
23      NR3=IPOINT(IRS-1)
24      NR2=NR3
        NR3=CALLS(NR3,7)
        IF(CALLS(NR1,6).GE.CALLS(NR3,6)) GO TO 24
        CALLS(NR2,7)=NR1
        CALLS(NR1,7)=NR3
        IPOINT(IRS-1)=CALLS(IPOINT(IRS-1),7)
25      ITS=CALLS(NR,6)+1.
        IF(ITS.NE.KSLOT) GO TO 20
        NR1=NR
26      NR=CALLS(NR,7)
        PRINT 27
27      FORMAT(* TOO MANY COLLISIONS*)
        STOP

C   UPDATE ASSIGNMENT QUEUE
28      NSLOT=(KSLOT+4)*M/5
        IF(NSLOT.EQ.LASLOT) GO TO 36
        LASLOT=NSLOT
29      LAP=LAP+1
        LAP1=ISLOT(LAP1,4)
        IF(LAP.NE.35000) GO TO 30
        NR20=NREQR
        QR20=NQLIT+IQLIT
30      AV1=AV1+1.
        IF(ISLOT(LAP1,1).EQ.0) GO TO 31
        NREQR=NREQR+1
```

```
      AVD=AVD-(ISLOT(LAP1,1)-LAP)/AVGDN
      ISLOT(LAP1,1)=0
 31   IF(AV1.LT.AVGDN) GO TO 35
      IPRNT=IPRNT+1
      DELY(IPRNT)=AVD
      AAVD=AAVD+AVD
      CAVD=CAVD+1
      AVD=0.
      AV1=0.
      IF(IPRNT.LT.10) GO TO 35
      IPRNT=0
      IF(IGLIT.EQ.0) GO TO 33
      NGLIT=NGLIT+IGLIT
      GR=(NGLIT-GR20)*100./(NREQP+NGLIT-NR20-GR20)
      USAGE=(NREQP+NGLIT-NR20-GR20)*100./(LAP-35000.)
      WRITE(2,32) IGLIT,NGLIT,USAGE,GR
 32   FORMAT(/6X,15H ** GLITCHES ** I6,8H   TOTAL I6,
     *8H    USAGE F6.2,15H%   GLITCH RATE F7.3,1H%)
 33   WRITE(2,34) (DELY(IL1),IL1=1,10),LAP,NREQP
 34   FORMAT(F6.1,9F6.1,I8,I10)
      IGLIT=0
 35   IF(LAP.LT.NSLOT) GO TO 28
C
C  UPDATE THE CALL
 36   DO 58 J=1,I
      NC=ICA(J)
      MSLOT=CALLS(NC,1)*A*C/T+1.
      CALLS(NC,1)=CALLS(NC,1)+T
      CALLS(NC,2)=CALLS(NC,2)-1.
      IF(CALLS(NC,2).NE.0.) GO TO 37
      IAD=CALLS(NC,3)
      CALLS(NC,1)=PACK(IAD,1)
      CALLS(NC,2)=PACK(IAD,2)
      CALLS(NC,3)=PACK(IAD,3)
      IFREE(IFTP)=IAD
      IFTP=IFTP+1
      IFTPT=IFTPT+1
      IF(IFTP.EQ.801) IFTP=1
      IF(CALLS(NC,3).EQ.0.) CALL GEN
C  RESEQUENCE REQUESTS
 37   CALLS(NC,6)=CALLS(NC,1)*A*C/T
      IF(CALLS(NC,6).GT.CALLS(NR,6)) GO TO 38
      INTEM=CALLS(NC,7)
      NR=NC
      GO TO 21
 38   IPOINT(1)=NC
      NR2=NC
      DO 39 IPS=2,11
      IF(CALLS(NC,6).LT.CALLS(IPOINT(IPS),6)) GO TO 40
      IPOINT(IPS-1)=CALLS(NR2,7)
 39   NR2=IPOINT(IPS)
      IPOINT(11)=NC
      CALLS(NR2,7)=NC
      CALLS(NC,7)=0.
      GO TO 42
 40   NR1=IPOINT(IPS-1)
 41   NR2=NR1
      NR1=CALLS(NR1,7)
      IF(CALLS(NC,6).GE.CALLS(NR1,6)) GO TO 41
      CALLS(NR2,7)=NC
      CALLS(NC,7)=NR1
```

```
      IPOINT(IRS-1)=CALLS(IPOINT(IRS-1),7)

C   INSERT REQUESTS IN SLOT ARRAY
42    NFASL=NFAST
      NFAST=MSLOT*M/A+C
      DO 58 MK=1,M
      NFAS=MSLOT*M/A+MK*C
      IF(NFAS.LE.LAP)  GO TO 43
      IF(NFAST.GE.NFASL) GO TO 44
      IF(MK.EQ.1) GO TO 43
      ISTART(MK,1)=ISTART(MK-1,1)
      ISTART(MK,2)=ISTART(MK-1,2)
      GO TO 44
43    ISTART(MK,1)=LAP+1
      ISTART(MK,2)=ISLOT(LAP1,4)
      NSLOTL=LAP1
44    ITS4=ISTART(MK,1)
      NSLOT=ISTART(MK,2)
      IF(ITS4.LE.LAP) GO TO 43
45    IF(ISLOT(NSLOT,1).GT.NFAS) GO TO 50
      IF(ISLOT(NSLOT,1).EQ.0) GO TO 48
      IF(ITS4.GE.(NFAS+LIMIT)) GO TO 47
46    ITS4=ITS4+1
      NSLOTL=NSLOT
      NSLOT=ISLOT(NSLOT,4)
      GO TO 45
C   A GLITCH
47    IGLIT=IGLIT+1
      GO TO 49
C   THIS SLOT IS FREE
48    IF(ITS4.LT.NFAS) GO TO 46
      ISLOT(NSLOT,1)=NFAS
      ISLOT(NSLOT,2)=NC
      ISLOT(NSLOT,3)=MK
49    ISTART(MK,1)=ITS4
      ISTART(MK,2)=NSLOT
      GO TO 58

C   ENTER SHIFT SEQUENCE
50    JSLOT=NSLOT
      JSLOTL=NSLOTL
      ISTART(MK,1)=ITS4
51    ITS4=ITS4+1
      IF(ITS4.GE.(ISLOT(JSLOT,1)+LIMIT)) GO TO 52
      JSLOTL=JSLOT
      JSLOT=ISLOT(JSLOT,4)
      IF(ISLOT(JSLOT,1).NE.0) GO TO 51
      GO TO 53
C   A GLITCH
52    IGLIT=IGLIT+1
      ITS4=ITS4-1
      IF(JSLOT.EQ.NSLOT) GO TO 54
53    ISLOT(JSLOTL,4)=ISLOT(JSLOT,4)
      ISLOT(JSLOT,4)=NSLOT
      ISLOT(NSLOTL,4)=JSLOT
```

```
54    ISLOT(JSLOT,1)=NFAS
      ISLOT(JSLOT,2)=NC
      ISLOT(JSLOT,3)=MK
      ISTART(MK,2)=JSLOT

      IF(MK.EQ.M) GO TO 58
      MK2=MK+1
      DO 56  MK1=MK2,M
      IF(ISTART(MK1,1).LT.ISTART(MK,1)) GO TO 56
      IF(ISTART(MK1,1)-ITS4) 55,57,56
55    ISTART(MK1,1)=ISTART(MK1,1)+1
56    CONTINUE
      GO TO 58
57    ISTART(MK1,2)=JSLOTL
58    CONTINUE

C  TEST FOR END
      IF(LAP.LT.1000000) GO TO 20
      GR=(NGLIT-GR20)*100./(NREQP+NGLIT-NR20-GR20)
      USAGE=(NREQP+NGLIT-NR20-GR20)*100./(LAP-35000.)
      AAVD=AAVD/CAVD
      WRITE(2,59) USAGE,GR,AAVD
59    FORMAT(//10X15HAVERAGE USAGE = F7.2,1H%11H   AVERAGE
     *13HGLITCH RATE = F5.3,1H%//10X15HAVERAGE DELAY =
     *F5.1,6H SLOTS )
      CALLEN=ST+SP+SS
      AT=ST/NT
      AP=SP/NP
      AS=SS/NS
      PT=ST/CALLEN*100.
      PP=SP/CALLEN*100.
      PS=SS/CALLEN*100.
      TPW=100.*(NPT*T-ST)/(NPT*T)
      UTPW=USAGE*ST/(NPT*T)
      WRITE(2,60)PT,PP,PS,CALLEN,NT,AT,NP,AP,NS,AS,NPT,TPW,UTPW
60    FORMAT(///20X21HCUMULATIVE STATISTICS /20X22(1H-)//
     *10X23HTIME PERCENTAGES     TALK F11.3/29X5HPAUSE F10.3/
     *29X7HSILENCE F8.3/16X15HTOTAL CALL TIME F9.1,4H SEC //
     *10X11HNO.OF TALKS I9,13H    AVG.LENGTH F7.3,4H SEC /
     *10X12HNO.OF PAUSES I8,13H    AVG.LENGTH F7.3,4H SEC /
     *10X14HNO.OF SILENCES I6,13H    AVG.LENGTH F7.3,4H SEC //
     *10X13HNO.OF PACKETS I7,16H    PERCENT WASTE F7.3,
     *15H    SPEECH USAGE F8.4)
      END
```

```
      SUBROUTINE GEN
C   THIS SUBROUTINE GENERATES A TALKSPURT AND A SILENCE,
C   AND ASSIGNS SPEECH TO PACKETS
      COMMON CALLS(100,7),PACK(800,3),IFREE(800),ISLOT(800,4)
      COMMON TL0(25),TL1(21),TL2(21),TL3(21),TL4(21)
      COMMON PLEN(24),CLEN(21),TTLR(21),TALK(101),PAUSE(100)
      COMMON/1/ITT(31),NT,ST/2/IPT(31),NP,SP
      COMMON /3/IST(31),NS,SS/4/IPW(11),SPIP,SPACE,T
      COMMON /5/IPD(11),ND,NPA,NPB,NC,IPP,NPT,IPPT,IPTPT
      CALLS(NC,3)=NPA

C   A TALKSPURT
4     DO 66 LCT=1,2
      R=RANF(0)
      R2=RANF(0)
      J=0
      IF(R.GT..6) GO TO 8
C   ZERO PAUSES
      IF(R2.GE..05) GO TO 5
      I=100.*R2+1.
      TL=TL0(I)+(TL0(I+1)-TL0(I))*(100.*R2+1.-I)
      GO TO 7
5     IF(R2.NE.1.) GO TO 6
      TL=7.
      GO TO 7
6     I=20.*R2+5.
      TL=TL0(I)+(TL0(I+1)-TL0(I))*(20.*R2+5.-I)
7     TSL=TL
      IF(LCT.EQ.2) CALL TSTAT(TL)
      K=0
      TALK(1)=TL
      GO TO 33

8     IF(R.GE..6) GO TO 12
C   ONE PAUSE
C   FIND TALKSPURT LENGTH
9     IF(R2.NE.1.) GO TO 10
      TSL=13.
      GO TO 11
10    I=20.*R2+1.
      TSL=TL1(I)+(TL1(I+1)-TL1(I))*(20.*R2+1.-I)
11    K=1
      GO TO 26

12    IF(R.GE..88) GO TO 16
C   TWO PAUSES
C   FIND TALKSPURT LENGTH
13    IF(R2.NE.1.)GO TO 14
      TSL=16.
      GO TO 15
14    I=20.*R2+1.
      TSL=TL2(I)+(TL2(I+1)-TL2(I))*(20.*R2+1.-I)
15    K=2
      GO TO 26

16    IF(R.GE..925) GO TO 20
C   THREE PAUSES
C   FIND TALKSPURT LENGTH
17    IF(R2.NE.1.) GO TO 18
      TSL=19.
```

```
      GO TO 19
18    I=20.*R2+1.
      TSL=TL3(I)+(TL3(I+1)-TL3(I))*(20.*R2+1.-I)
19    K=3
      GO TO 26

20    IF(R.GE..95) GO TO 24
C   FOUR PAUSES
C   FIND TALKSPURT LENGTH
21    IF(R2.NE.1.) GO TO 22
      TSL=22.
      GO TO 23
22    I=20.*R2+1.
      TSL=TL4(I)+(TL4(I+1)-TL4(I))*(20.*R2+1.-I)
23    K=4
      GO TO 26

C   GREATER THAN FOUR PAUSES
C   NUMBER OF PAUSES=K
24    IF(R.LT..99985) GO TO 25
      K=100
      TSL=550.
      GO TO 26
25    K=(.6148/(1.-R))**.5525
      I=20.*R2+1.
      TSL=K/4.*(TL4(I)+(TL4(I+1)-TL4(I))*(20.*R2+1.-I))


C   FIND PAUSE LENGTHS
26    IF(LCT.EQ.1)GO TO 33
      SUMP=0.
      DO 30 L=1,K
      R3=RANF(0)
      IF(R3.GE..05) GO TO 27
      I=100.*R3+1.
      PAUSE(L)=PLEN(I)+(PLEN(I+1)-PLEN(I))*(100.*R3+1.-I)
      GO TO 30
27    IF(R3.GE..95) GO TO 28
      I=20.*R3+5.
      PAUSE(L)=PLEN(I)+(PLEN(I+1)-PLEN(I))*(20.*R3+5.-I)
      GO TO 30
28    IF(R3.LT..9999) GO TO 29
      PAUSE(L)=6.
      GO TO 30
29    PAUSE(L)=.8721*ALOG10(.7846/(1.-R3))+.4*(.7136/(1.-R3))
     ***.2079
30    SUMP=SUMP+PAUSE(L)
      IF(SUMP.LT.TSL) GO TO 32
      J=J+1
      IF(J.LT.100) GO TO 26
      J=0
      R2=RANF(0)
      IF(K-4) 31,21,24
31    IF(K-2) 9,13,17

C   FIND THE TALK LENGTHS
32    SUMT=TSL-SUMP
      TL=SUMT
      K1=K+1
      SUMR=0.
```

```
      DO 61 L=1,K1
      TALK(L)=RANF(0)
61    SUMR=SUMR+TALK(L)
      DO 161 L=1,K1
161   TALK(L)=TALK(L)*SUMT/SUMR

C  INCREMENT TALK AND PAUSE STATISTICS
      DO 62 L=1,K
      CALL TSTAT(TALK(L))
62    CALL PSTAT(PAUSE(L))
      CALL TSTAT(TALK(K+1)).

C  FIND RESPONSE TIME
33    R=RANF(0)
      IF(R.LT..05) GO TO 67
      IF(R.GE..95) GO TO 68
      I=20.*R
      RL=TTLR(I)+(TTLR(I+1)-TTLR(I))*(20.*R-I)
      GO TO 69
67    RL=.3643-(.008722/R)**.4556
      GO TO 69
68    RL=.8174*ALOG10(.2729/(1.-R))+.5315*(.1245/(1.-R))**.2757
69    IF(LOT.EQ.2) GO TO 70
      RT1=RL
      SL=RL+TSL
      TSL1=TSL
66    CONTINUE
70    SL=SL+RL
      IF(SL.LT.0.) SL=0.
      CALL SSTAT(SL)
C  PACKET DETERMINATION
45    IF(SPACE.EQ.0.) GO TO 77
      IF(K.EQ.0) GO TO 73
71    DO 72 I=1,K
      IF(SPACE.LT.TALK(I)) GO TO 75
      SPACE=SPACE-TALK(I)
      IF(SPACE.LT.PAUSE(I)) GO TO76
      SPACE =SPACE-PAUSE(I)
      SPIP=SPIP+PAUSE(I)
72    CONTINUE
73    IF(SPACE.LT.TALK(K+1)) GO TO 74
      SPACE=SPACE-TALK(K+1)
      GO TO 90
74    TALK(K+1)=TALK(K+1)-SPACE
      CALL SPSTAT
      GO TO 68
75    TALK(I)=TALK(I)-SPACE
      CALL SPSTAT
      GO TO 78
76    SPIP=SPIP+SPACE
      PAUSE(I)=PAUSE(I)-SPACE
      CALL SPSTAT
      CALL SOSTAT(PAUSE(I))
      IF(I.EQ.K) GO TO 88
      I=I+1
      GO TO 78

C  NO SPACE CARRYOVER
77    ND=0
      IF(K.EQ.0) GO TO 88
      I=1
```

```
78      DO 87 J=I,K
        IF(SPACE.GT.0.) GO TO 84
79      ND=ND+1
        NPT=NPT+1
        IF(TALK(J).LE.T) GO TO 80
        TALK(J)=TALK(J)-T
        CALL SPSTAT
        GO TO 79
80      SPACE=T-TALK(J)
86      IF(SPACE.GE.PAUSE(J)) GO TO 81
        SPIP=SPIP+SPACE
        PAUSE(J)=PAUSE(J)-SPACE
        CALL SPSTAT
        CALL SQSTAT(PAUSE(J))
        GO TO87

81      SPIP=SPIP+PAUSE(J)
        SPACE=SPACE-PAUSE(J)
82      IF(J.NE.K) GO TO 87
        IF(SPACE.GE.TALK(K+1)) GO TO 83
        TALK(K+1)=TALK(K+1)-SPACE
        CALL SPSTAT
        GO TO 88
83      SPACE=SPACE-TALK(K+1)
        GO TO 90
84      IF(SPACE.GE.TALK(J)) GO TO 85
        TALK(J)=TALK(J)-SPACE
        CALL SPSTAT
        GO TO 79
85      SPACE=SPACE-TALK(J)
        GO TO 86
87      CONTINUE

C  LAST TALK AND NO PAUSE CASE
88      ND=ND+1
        NPT=NPT+1
        IF(TALK(K+1).LE.T) GO TO 89
        TALK(K+1)=TALK(K+1)-T
        CALL SPSTAT
        GO TO 88
89      SPACE=T-TALK(K+1)
90      IF(SPACE.GT.SL) GO TO 91
        SPIP=SPIP+SPACE
        SL=SL-SPACE
        CALL SPSTAT
        CALL SQSTAT(SL)
        GO TO 92
91      SPIP=SPIP+SL
        SPACE=SPACE-SL
        GO TO 4
92      PACK(NFB,3)=0.
        RETURN
        END
```

```
      SUBROUTINE TSTAT(TL)
      COMMON/1/ITT(31),NT,ST
      NT=NT+1
      ST=ST+TL
      IF(TL.GE.1.5) GO TO 1
      I=10*TL+1.
      GO TO 3
1     IF(TL.GE.2.5) GO TO 4
      I=5.*(TL-1.5)+16.
      GO TO 3
4     IF(TL.GE.5.) GO TO 5
      I=2.*(TL-2.5)+21.
      GO TO 3
5     IF(TL.GE.10.) GO TO 2
      I=TL-5.+26.
      GO TO 3
2     I=31
3     ITT(I)=ITT(I)+1
      RETURN
      END
```

```
      SUBROUTINE PSTAT(PL)
      COMMON/2/IPT(31),NP,SP
      NP=NP+1
      SP=SP+PL
      IF(PL.LT.3.) GO TO 1
      IPT(31)=IPT(31)+1
      RETURN
1     I=10.*PL+1.
2     IPT(I)=IPT(I)+1
      RETURN
      END
```

```
      SUBROUTINE SSTAT(SL)
      COMMON/3/IST(31),NS,SS
      NS=NS+1
      SS=SS+SL
      IF(SL.GE.3.) GO TO 1
      I=5.*SL+1.
      GO TO 3
1     IF(SL.GE.8.) GO TO 4
      I=2.*(SL-3.)+16.
      GO TO 3
4     IF(SL.GE.18.) GO TO 2
      I=(SL-8.)/2.+26.
      GO TO 3
2     I=31
3     IST(I)=IST(I)+1
      RETURN
      END
```

```
      SUBROUTINE SPSTAT
      COMMON/4/IPW(11),SPIP,SPACE,T
      SPACE=0.
      IF(SPIP.NE.0.) GO TO 1
      IPW(1)=IPW(1)+1
      RETURN
1     I=10.*SPIP/T+2.
      IF(I.EQ.12) I=11
      IPW(I)=IPW(I)+1
      SPIP=0.
      RETURN
      END
```

```
      SUBROUTINE SQSTAT(QUIET)
      COMMON CALLS(100,7),PACK(800,3),IFREE(800)
      COMMON/5/IPD(11),ND,NFA,NFB,NC,IFP,NFT,IFPT,IFTPT
      COMMON/4/IPW(11),SPIP,SPACE,T
      NFB=NFA
      PACK(NFA,1)=CALLS(NC,4)
      CALLS(NC,4)=CALLS(NC,4)+ND*T+QUIET
      PACK(NFA,2)=ND
      IFP=IFP+1
      IF(IFP.EQ.801) IFP=1
      IFPT=IFPT+1
      IF(IFPT.GE.IFTPT) PRINT 2,IFTPT,IFPT
2     FORMAT(* PACK TOO SMALL*2I10)
      NFA=IFREE(IFP)
      PACK(NFB,3)=NFA
      IF(ND.GT.10) GO TO 1
      IPD(ND)=IPD(ND)+1
      ND=0
      RETURN
1     IPD(11)=IPD(11)+1
      ND=0
      RETURN
      END
```

REFERENCES

1 C.C. Burton, "The growth and development of mobile telephone and paging systems in the US", Telecommunications J., Vol. 42, No. 8, Aug. 1975, p. 471-5.

2 M. Watanabe & K. Miyauchi, "Research and development on land mobile telephone systems", Rev. Elec. Commun. Labs., Vol. 25, No. 11-12, Nov.-Dec. 1977, p. 1141-6.

3 R.E. Fisher, "A description of the Bell system 850 MHz high capacity mobile telecommunications system", Proc. Int. Conf. Commun. (ICC), Vol. 2, 1976, p. 27/1-4.

4 L. Armstrong, "Motorola cellular system gets nod", Electronics, Vol. 50, No. 24, Nov. 20, 1977, p. 74-6.

5 G.M. Dewine, "Advanced mobile telephone systems - now!", Telecommunications J., Vol. 41, No. 9, Sept. 1974, p. 542-5.

6 R. Giener & M. Gunther, "Automation of the public mobile radio-telephone system", Philips Telecom. Rev., Vol. 33, No. 1, March 1975, p. 35-50.

7 J.W. Bayless et al., "Voice signals : bit by bit", IEEE Spectrum, Vol. 10, No. 10, Oct. 1973, p. 28-34.

8 J.W. Forgie, "Speech transmission in packet-switched store-and-forward networks", Proc. AFIPS Nat. Computer Conf., 1975, p. 137-41.

9 A.B. Carleial & M.E. Hellman, "Bistable behaviour of ALOHA type systems", IEEE Trans. on Commun., Vol. COM-23, No. 4, April 1975, p. 401-9.

10 I. Gitman, R.M. Van Slyke & H. Frank, "Routing in packet-switching broadcast radio networks", IEEE Trans. on Commun., Vol. COM-24, No. 8, Aug. 1976, p. 926-30.

11 I. Gitman, "Comparison of hop-by-hop and end-to-end acknowledgement schemes in computer communication networks", IEEE Trans. on Commun., Vol. COM-24, No. 11, Nov. 1976, p. 1258-62.

12 L.G. Roberts, "The evolution of packet switching", Proc. IEEE, Vol. 66, No. 11, Nov. 1973, p. 1307-13.

13 N. Abramson, "The ALOHA system - another alternative for computer communication", Proc. AFIPS Joint Computer Conf., 1970, Vol. 37, P. 281-5.

14 R. Binder et al., "ALOHA packet broadcasting - a retrospect", Proc. AFIPS Nat. Computer Conf., 1975, p. 203-15.

15 R.E. Kahn, "The organization of computer resources into a packet radio network", Proc. AFIPS Nat. Computer Conf., 1975, p. 177-86.

16 N. Abramson, "Packet switching with satellites", Proc. AFIPS Nat. Computer Conf., 1975, p. 695-702.

17 L.G. Roberts, "Dynamic allocation of satellite capacity through packet reservation", Proc. AFIPS Nat. Computer Conf., 1975, p. 711-16.

18 L. Kleinrock & S.S. Lam, "Packet switching in a slotted satellite channel", Proc. AFIPS Nat. Computer Conf., 1975, p. 703-10.

19 S.S. Lam & L. Kleinrock, "Dynamic control schemes for a packet switched multi-access broadcast channel", Proc. AFIPS Nat. Computer Conf., 1975, p. 143-53.

20 L. Kleinrock & F.A. Tobagi, "Packet switching in radio channels : part 1 - carrier sense multiple-access modes and their throughput-delay characteristics", and "part 11 - the hidden terminal problem in carrier sense multiple-access and the busy tone solution", IEEE Trans. on Commun., Vol. COM-23, No. 12, Dec. 1975, p. 1400-33.

21 F.A. Tobagi & L Kleinrock, "Packet switching in radio channels : part 111 - polling and (dynamic) split-channel reservation multiple access", IEEE Trans. on Commun., Vol. COM-24, No. 8, Aug. 1976, p. 832-44.

22 I. Gitman, "On the capacity of slotted ALOHA networks and some design problems", IEEE Trans. on Commun., Vol. COM-23, No.3, March 1975, p. 305-317.

23 M. Ferguson, "Control stability waiting time in slotted ALOHA", IEEE Trans. on Commun., Vol. COM-23, No. 11, Nov. 1975, p. 1306-11.

24 W. Crowther et al., "A system for broadcast communication : reservation ALOHA", Proc. 6th. Hawaii Int. Conf. on System Sciences, 1973, p. 371-4.

25 F. Borgonovo & L. Fratta, "A new technique for satellite broad-cast channel communication", Proc. 5th. Data Communications Symp., 1977, p. 2/1-4.

26 S.S. Rappaport, "Traffic capacity of DAMA systems using collision type request channels", Rec. Nat. Telemetering Conf. (NTC), 1977, Vol. 3, p. 48/1/1-6.

27 D.C. Cox, "Time and frequency domain characteristics of multipath propagation at 910 MHz in a suburban mobile-radio environment", Radio Science, Vol. 7, No. 12, Dec. 1972, p. 1069-77.

28 D.C. Cox, "910 MHz urban mobile propagation : multipath character-istics in New York City", IEEE Trans. on Vehic. Tech., Vol. VT-22, No. 11, Nov. 1973, p. 104-10.

29 D.C. Cox, " Correlation bandwidth and delay spread multipath propagation ststistics for 910 MHz urban mobile radio channels", IEEE Trans. on Commun., Vol. COM-23, No. 11, Nov. 1975, p. 1271-80.

30 T. Nomura & N. Yoshikawa, "Mobile radio link design", Rev. Elec. Commun. Labs., Vol. 23, No. 1-2, Jan.-Feb. 1975, p. 144-60.

31 R.C. French, "Mobile radio data transmission in the urban environment", Proc. Int. Conf. on Commun. (ICC), 1976, Vol. 2, p. 27/15-20.

32 J.C. Feggeler, "Experiments with companding in mobile telephony", Proc. Int. Conf. on Commun. (ICC), 1976, Vol. 2, p. 27/5-10.

33 R.W. Nettleton & G.R. Cooper, "Error performance of a spread spectrum mobile communications system in a rapidly-fading environment", Rec. Nat. Telemetering Conf. (NTC), 1977, Vol. 3, p. 35/5/1-8.

34 R.E. Kahn et al., "Advances in packet radio technology", Proc. IEEE, Vol. 66, No, 11, Nov. 1978, p. 1468-96.

35 E.N. Skomal, "Distribution and frequency dependence of unintentionally generated man-made VHF/UHF noise in metropolitan areas", IEEE Trans. on Electro. Compat., Vol. EMC-7, No. 3, Sept. 1965, p. 263-78.

36 S. Fralick & J. Garret, "Technological considerations for packet radio networks", Proc. AFIPS Nat. Computer Conf., 1975, p. 233-43.

37 K. Bullington & J.M. Fraser, "Engineering aspects of TASI", Bell Syst. Tech. J., Vol. 38, No. 3, March 1959, p. 353-64.

38 J.M. Fraser, D.B. Bullock & N.C. Long, "Over-all characteristics of a TASI system", Bell Syst. Tech. J., Vol. 41, No. 4, July 1962, p. 1439-54.

39 H. Miedema & M.G. Schachtman, "TASI quality - effect of speech detectors and interpolation", Bell Syst. Tech. J., Vol. 41, No. 4, July 1962, p. 1455-73.

40 T.G. Birdsall, M.P. Ristenbatt & S.B. Weinstein, "Analysis of asynchronous time multiplexing of speech sources", IEEE Trans. on Commun. Systems, Vol. CS-10, Dec. 1962, p. 390-7.

41 C. Ota & K. Amano, "A digital speech interpolation system", Electr. and Commun. in Japan, Vol. 56-A, No. 8, 1973, p. 26-34.

42 E. Lyghounis, I. Poretti & G. Monti, "Speech interpolation in digital transmission systems", IEEE Trans. on Commun., Vol. COM-22, No. 9, Sept. 1974, p. 1179-89.

43 S.J. Campanella, "Performance of digital speech interpolation systems for commercial telecommunications", Rec. Nat. Telemetering Conf. (NTC), 1975, Vol. 2, p. 46/1-8.

44 S.J. Campanella, "Digital speech interpolation", Comsat. Tech. Rev., Vol. 6, No. 1, Spring 1976, p. 127-58.

45 A.M. Molinari & F.C. Vagliani, "Bit rate per channel halving in PCM multipliers by speech interpolation and adaptive quantization", IEEE Trans. on Commun., Vol. COM-26, No. 5, May 1978, p. 638-46.

46 R. Woitawitz, "The effects of speech statistics in an instantaneous priority speech interpolation system", Proc. Int. Conf. on Commun. (ICC), 1978, Vol. 3, p. 50/1/1-5.

47 D. Seitzer, H. Gerhauser & G. Landenbucher, "A comparative study of high quality digital speech interpolation", Proc. Int. Conf. on Commun. (ICC), 1978, Vol. 3, p. 50/2/1-5.

48 H.G. Suyderhoud, J.A. Jankowski & R.P. Ridings, "Results and analysis of the speech predictive encoding communications system field trial", Comsat. Tech. Rev., Vol. 4, No. 2, Fall 1974, p. 371-91.

49 D.M. Kennedy, "Digital developments in the satellite field", Symposium on Digital Communications Systems, Sydney 1977.

50 M.N. Dor, "Guide to the length of buffer storage required for random (Poisson) input and constant output rates", IEEE Trans. on Electronic Computers, Vol. EC-16, No. 10, Oct. 1967, p. 683-4.

51 W.W. Chu, "Buffer behaviour for batch Poisson arrivals and single constant output", IEEE Trans. on Commun., Vol COM-18, No. 5, Oct. 1970, p. 613-18.

52 H. Kobayashi & A.G. Konheim, "Queueing models for computer communications systems analysis", IEEE Trans. on Commun., Vol. COM-25, No. 1, Jan. 1977, p. 2-28.

53 B. Gopinath & J.A. Morrison, "Single server queues with correlated input", Bell Syst. Tech. J., Vol. 59, No. 9, Nov. 1977, p. 1743-68.

54 Z. Fluhr & E. Nussbaum, "Switching plan for a cellular mobile telephone system", IEEE Trans. on Commun., Vol. 21, No. 11, Nov. 1973, p. 1281-6.

55 M.R. Karim, "A controller for the mobile logic unit in the Bell system high-capacity mobile telecommunications system", Proc. IEEE Vehicular Technology Group Conf. (VTG), 1976, p. 72-4.

56 E.J. Addeo, "An approach to digital signal processing for a high capacity mobile telephone system", Proc. IEEE Vehicular Technology Group Conf. (VTG), 1976, p. 52-7.

57 L.H. Rosenthal & L.E. Schloemer, "DAS-11 : A microprocessor-based toll ticketing and traffic analysis system for the Motorola pulsar mobile telephone control terminal", Proc. Int. Conf. on Commun. (ICC), 1976, Vol. 2, p. 27/21-6.

58 M. Stern & V. Graziano, "Dyna-tac update", Proc. Int. Conf. on Commun. (ICC), 1978, Vol. 1, p. 4/1/1-4.

59 N. Yoshikawa & T. Nomura, "Fundamental consideration of radio zone repetition in a mobile radio system", Rev. Elec. Commun. Labs., Vol. 23, No. 1-2, Jan.-Feb. 1975, p. 161-70.

60 N. Abramson, "The ALOHA system - another alternative for computer communications", Proc. AFIPS Joint Computer Conf., 1970, p. 281-5.

61 L. Kleinrock & S. Lam, "Packet switching in a multi-access broadcast channel : performance evaluation", IEEE Trans. on Commun., Vol. COM-23, No. 4, April 1975, p. 410-23.

62 N.S. Jayant, "Digital coding of speech waveforms : PCM, DPCM and DM quantizers", Proc. IEEE, Vol. 62, No. 5, May 1974, p. 611-33.

63 P. Cummiskey, N.S. Jayant & J.L. Flanagan, "Adaptive quantization in differential PCM coding of speech", Bell Syst, Tech. J., Vol. 52, No. 7, Sept. 1973, p. 1105-19.

64 P. Noll, "A comparative study of various quantization schemes for speech encoding", Bell Syst. Tech. J., Vol. 54, No. 9, Nov. 1975, p. 1597-614.

65 N.S. Jayant, "Step-size transmitting differential coders for mobile telephony", Bell Syst. Tech. J., Vol. 54, No. 9, Nov. 1975, p. 1557-81.

66 D.L. Schilling, J. Garodnick & H.A. Vang, "Voice encoding for the space shuttle using adaptive delta modulation", IEEE Trans. on Commun., Vol. COM-26, No. 11, Nov. 1978, p. 1652-9.

67 R. Zelinski & P. Noll, "Adaptive transform coding of speech signals", IEEE Trans. on Acoustics Speech and Signal Processing, Vol. ASSP-25, No. 4, Aug. 1977, p. 299-309.

68 R.E. Crochiere et al., "A study of objective measures for speech waveform coders", Proc. Int. Zurich Seminar on Digital Commun., 1978, p. H1/1-7.

69 T. Bially & W.M. Anderson, "A digital channel vocoder", IEEE Trans. on Commun. Tech., Vol. COM-18, No. 4, Aug. 1970, p. 435-42.

70 B.S. Atal & S.L. Hanauer, "Speech analysis and synthesis by linear prediction of the speech wave", J. Acoustical Society of America, Vol. 50, No. 2, Part 2, Aug. 1971, p. 637-55.

71 L.R. Morris & M.J. Gough, "An economical hardware realization of a digital linear predictive speech synthesizer", IEEE Trans. on Commun., Vol. COM-22, No. 1, Jan 1974, p. 82-8.

72 G.A. Arredondo & J.I. Smith, "Voice and digital transmission in a mobile radio channel at 850 MHz", Proc. Nat. Electronics Conf., 1974, Vol. 29, p. 74-9.

73 R. Esposito & R.E. Buck, "A mobile wide band measurement system for urban man made noise", IEEE Trans. on Commun., Vol. COM-21, No. 11, Nov. 1973, p. 1225-32.

74 W.F. Utlaut, "Spread-spectrum : principles and possible application to spectrum utilization and allocation", IEEE Commun. Society Magazine, Vol. 16, No. 5, Sept 1978, p. 21-31.

75 H. Frank & I. Gitman, "Packet radio system - network considerations", Proc. AFIPS Nat. Computer Conf., 1975, p. 217-31.

76 R.E. Kahn, "The organization of computer resources into a packet radio network", IEEE Trans. on Commun., Vol. COM-25, No. 1, Jan. 1977, p. 169-78.

77 A.C. Norwine & O.J. Murphy, "Characteristic time intervals in telephone conversations", Bell Syst. Tech. J., Vol. 17, No. 2, April 1938, p. 281-91.

78 D.N. Sherman, "Storage and delay estimates for asynchronous multiplexing of data in speech", IEEE Trans. on Commun. Tech., Vol. COM-19, No. 8, Aug 1971, p. 551-5.

79 H.B. Kerkre & C.L. Saxena, "Three state Markov model of speech on telephone lines and optimal utilization of communication systems by TASI techniques", Comput. & Elect. Engn., Vol. 4, No. 3, Sept. 1977, p. 235-50.

80 C.J. Weinstein, "Fractional speech loss and talker activity model for TASI and for packet-switched speech", IEEE Trans. on Commun., Vol. COM-26, No. 8, Aug. 1978, p. 1253-7.

81 E.J. Klemmer, "Subjective evaluation of transmission delay in telephone conversation", Bell Syst. Tech. J., Vol. 46, No. 5-7, May-July 1967, p. 1141-7.

82 P.T. Brady, "Effect of transmission delay on conversational behaviour on echo-free telephone circuits", Bell Syst. Tech. J., Vol. 50, No. 1, Jan. 1971, p. 115-34.

83 G.K. Helder, "Customer evaluation of telephone circuits with delay", Bell Syst. Tech. J., Vol. 45, No. 7, Sept. 1966, p. 1157-91.

84 J.R. Cavanaugh, R.W. Hatch & J.L. Sullivan, "Models for the subjective effects of loss, noise, and talker echo on telephone conversations", Bell Syst. Tech. J., Vol. 55, No. 9, Nov. 1976, p. 1319-71.

85 P.T. Brady & G.K. Helder, "Echo suppressor design in telephone communications", Bell Syst. Tech. J., Vol. 42, No. 6, Nov. 1963, p. 2893-917.

86 CCITT White Book, Vol. V, Recommendation P.14, 1968.

87 D.L. Duttweiler, "A twelve channel digital echo canceller", IEEE Trans. on Commun., Vol. COM-26, No. 5, May 1978, p. 647-53.

88 R. Ahmend & R. Fatechand, "Effect of duration on sound articulation", J. Acoustical Society of America, Vol. 31, No. 7, July 1959, p. 1022-29.

89 D.R. Cox & W.L. Smith, "Queues", London, Methuen; New York, Wiley, 1961.

90 L. Kleinrock, "Queueing systems, Vol. 1, Theory", New York, Wiley, Interscience, 1975.

91 B. Gavish & A.G. Konheim, "Computer communication via satellites - a queueing model", IEEE Trans. on Commun., Vol. COM-25, No. 1, Jan. 1977, p. 140-7.

92 A.D. Wyner, "On the probability of buffer overflow under an arbitrary bounded input-output distribution", SIAM J. Applied Maths., Vol. 27, No. 12, Dec. 1974, p. 544-70.

93 G.S. Fishman, "Estimating sample size in computing simulation experiments", Management Science, Vol. 18, No. 1, Sept. 1971, p. 21-38.

94 R. Fuller, "Report on speech assessment system", Report No. 2/77, Electrical Engineering Dept., University of Adelaide, Sept. 1977.

95 CCITT - 111 rd. Plenary Assembly (Geneva 1964), Blue Book, Vol. 11, p. 239, Published by the ITU, Geneva, 1965.

96 R. Lane, "Spectral and economic efficiencies of land mobile radio system", IEEE Trans. on Commun., Vol. COM-21, No. 11, Nov. 1973, p. 1177-87.

97 R. Frenkiel, "High capacity radiotelephone using small zones", IEEE Trans. on Vehic. Tech., Vol. VT-19, No. 2, May 1970, p. 173-7.

98 A.K. Johnson & J.I. Smith, "Radio frequency design for a high capacity mobile telephone system (HCMTS)", Proc. Nat. Telemetering Conf. (NTC), 1977, Vol. 1, p. 16/4/1-3.

99 S. Ito &. Y. Matsuzaka, "800 MHz band land mobile telephone system - an overall view", Rev. Elec. Commun. Labs., Vol. 25, No. 11, Nov.-Dec. 1977, p. 1147-56.

100 T. Kamata, M. Sakamota & K. Fukuzumi, "800 MHz land mobile telephone radio system", Rev. Elec. Commun. Labs., Vol. 25, No. 11-12, Nov.-Dec. 1977, p. 1157-71.

101 J.C. Feggeler, "A study of digitized speech in mobile telephony", Proc. Microwave Mobile Communications Symposium, 1976.

102 L.G. Anderson, "A simulation study of some dynamic channel assignment algorithms in a high capacity mobile telecommunications system", IEEE Trans. on Commun., Vol. COM-21, No. 11, Nov. 1973, p. 1294-301.

103 D.C. Cox & D.O. Reudink, "Increasing channel occupancy in large scale mobile radio systems : dynamic channel assignment", IEEE Trans. on Commun., Vol. COM-21, No. 11, Nov. 1973, p. 1302-6.

104 B. Pehani & S. Leonardis, "Integrated communication system with delta codecs", Proc. Int. Zurich Seminar on Digital Communications, March 1978, p. B3/1-5.

105 J.M. Balderston, "Multiple access telephony for an Australian satellite system", Telecom. Australia Research Labs. Report No. 6811.

106 G.D. Dill, Y. Tsuji & T. Muratani, "Application of SS-TDMA in a channelized satellite", Proc. Int. Conf. on Commun. (ICC), 1976, Vol. 3, p. 51/1-5.

107 H. Fuketa & Y. Watanabe, "TDMA system outline for domestic satellite communications", Rev. Elec. Commun. Labs., Vol. 26, No. 1-2, Jan.-Feb. 1978, p. 56-62.

108 Y. Watanabe & S. Samejima, "TDMA system for K band domestic satellite communications", Rev. Elec. Commun. Labs., Vol. 26, No. 1-2, Jan.-Feb. 1978, p. 63-79.

109 I. Jacobs et al., "CPODA - a demand assignment protocol for SATNET", Proc. 5 th. Data Communications Symp., Sept. 27-29, 1977, Smowbird Utah, p. 2/5-9.

110 M. Gerla, L. Nelson & L. Kleinrock, "Packet satellite multiple access models and measurements", Rec. Nat. Telemetering Conf. (NTC), 1977, Vol. 1, p. 12/2/1-8.

111 P. Baran, "Broad-band interactive communications services to the home : part 1 - potential market demand", IEEE Trans. on Commun., Vol. COM-23, No. 1, Jan.1975, p. 5-15.

112 E.R. Hafner, Z. Nenadal & M Tschanz, "A digital loop communications system", IEEE Trans. on Commun., Vol. COM-22, June 1974, p. 877-81.