AD-A039 018 CALIFORNIA UNIV LOS ANGELES DEPT OF COMPUTER SCIENCE F/G 9/2

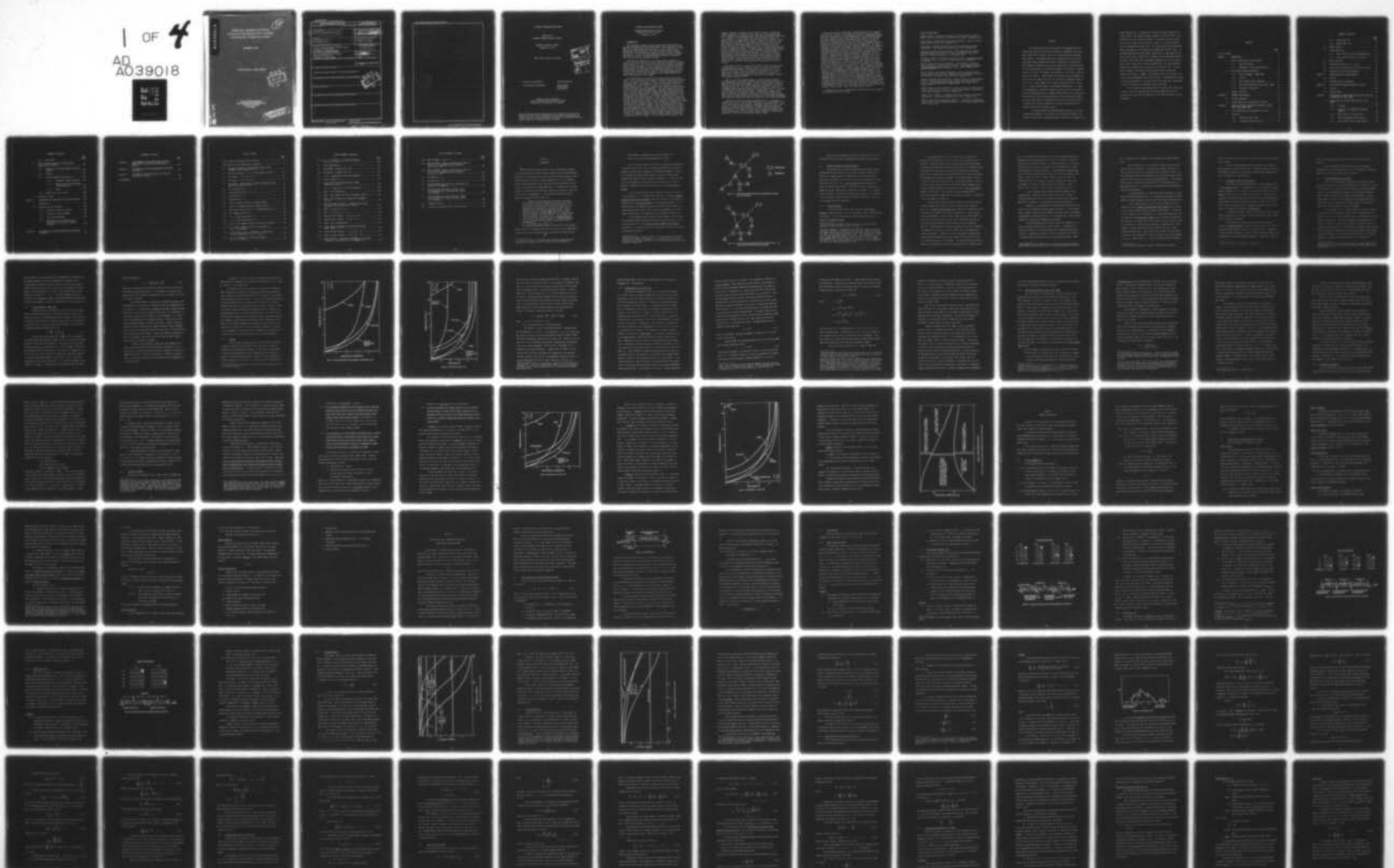
ADVANCED TELEPROCESSING SYSTEMS.(U)

DEC 76 L KLEINROCK DAHC15-73-C-0368

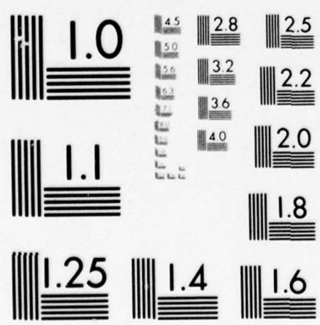
UNCLASSIFIED NL

1 OF 4

AD A039018



390



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

COMPUTER NETWORK RESEARCH
ADVANCED TELEPROCESSING SYSTEMS
SEMIANNUAL TECHNICAL REPORT

DECEMBER 31, 1976

Principal Investigator: Leonard Kleinrock

Computer Science Department
School of Engineering and Applied Science
University of California
Los Angeles

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Advanced Teleprocessing Systems		5. TYPE OF REPORT & PERIOD COVERED Semiannual Technical Report July 1 - Dece[redacted]976
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Leonard Kleinrock		8. CONTRACT OR GRANT NUMBER(s) DAHC 15-73-C-0368
9. PERFORMING ORGANIZATION NAME AND ADDRESS School of Engineering and Applied Science University of California Los Angeles, California 90024		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Advanced Research Projects Agency 1400 Wilson Boulevard Arlington, Virginia 22209		12. REPORT DATE Dec[redacted]976
		13. NUMBER OF PAGES 249
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report)
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for Public Release; Distribution Unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

S/N 0102-LF-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

ADVANCED TELEPROCESSING SYSTEMS

Sponsored by

ADVANCED RESEARCH PROJECTS AGENCY

SEMIANNUAL TECHNICAL REPORT

December 31, 1976

ARPA Contract DAHC-15-73-C0368

Principal Investigator: Leonard Kleinrock

Co-Principal Investigators: Gerald Estrin
Richard Muntz
Gerald Popek

Computer Science Department
School of Engineering and Applied Science
University of California, Los Angeles

ADVANCED TELEPROCESSING SYSTEMS

Advanced Research Projects Agency
Semiannual Technical Report

December 31, 1976

1. INTRODUCTION

This semiannual technical report covers research carried out by the Advanced Teleprocessing Systems group at UCLA under ARPA Contract DAHC 15-73-C-0368 during the period July 1, 1976 to December 31, 1976. Advancements have been made on all four contracted tasks, namely, ground radio packet switching, satellite studies, resource sharing and security. In the following paragraphs we describe that progress and point to the list of references which represent the published work results from this supported research.

Following this summary is a list of publications produced as a result of the recent research on this contract covering the six months being reported on. This list contains only those articles and reports which in fact did appear in print. Papers which have been submitted (of which there are many) are not listed here, but will be listed in future reports as they appear in the published literature. As usual, we devote the main body of this report to the detailed presentation of one aspect of this overall research, and we simply mention the other areas briefly in this summary.

The research which is detailed in the manin body of this report refers to various multiaccess schemes in packet switched radio systems. This particular problem lies at the core of our ground radio packet switching studies. The problem is how to properly gain access to a communications channel when its sources are both bursty and distributed geographically. Various random access schemes have been studied in the past under various assumptions. The work reported on here is the Ph.D. dissertation of Michel Scholl (Chairman, Leonard Kleinrock) and is entitled, "Multiplexing Techniques for Data Transmission over Packet Switched Radio Systems." A number of important aspects of this problem have been addressed in the research reported upon here. Let us summarize the results achieved. First, some new access schemes are defined and analyzed which turn out to be far more efficient than time division multiple access (TDMA). These schemes take advantage of carrier sensing and are absolutely conflict-free as opposed to many of the other multiaccess schemes studied. They are particularly well-suited to a small population of terminals rather than to the large case and perform roughly the same (slightly worse) as does polling; the advantage they have over polling is that they require no centralized control, such control being a liability from a reliability and security point-of-view. An improved version of these schemes, namely, minislotted alternating priorities (MSAP) is then introduced and analyzed. This scheme is superior to polling at all levels of traffic. Under heavy traffic conditions, it performs better than carrier sensing multiple access (CSMA) for all numbers of users and under light traffic conditions, it is competitive with the CSMA only when the number of sources of users is small. A third result of this research is to consider a mixed ALOHA carrier sense (MACS) which permits a single large user to steal slots which are unused by a large population of small users (each of which operates in an ALOHA fashion). By including this large user traffic in addition to the background traffic, we significantly increase the total channel throughput. In summary, then, the major contributions of this research are as follows: the introduction of performance evaluation of new methods for multiplexing users on a packet switched multiple access radio channel; and secondly, the inclusion of traffic from different sources from the same radio channel thereby providing efficient channel utilization and leading to the design of new mixed access schemes. Further details can be found in the abstract and dissertation reproduced in the main body following the list of publications.

Our efforts in Task I, packet radio studies, have moved along with considerable success in this period. First, of course, is the work reported upon in reference 7 which follows in its entirety as the body of this report. The work reported in references 8, 9, 10 and 11 represents our additional results in this area and is related to theoretical as well as implementation and practical problems of ground radio packet switching systems.

Task II, satellite experimentation, has been highly productive. Our experiments on the Atlantic satellite have been rather extensive in reservation TDMA and slotted ALOHA. We have observed that reservation TDMA is a fairly efficient scheme and behaves as predicted in a rather straight-forward way. Our slotted ALOHA measurement and simulation experiments have supported the fact that such channels must be controlled or their efficiency is badly degraded. In references 2 and 3, we describe results on various access control schemes which automatically adapt to channel load conditions and control the slotted ALOHA system. These control schemes provide an efficient and stable channel performance and essentially resolve the stability problem in a very effective fashion.

Resource allocation and sharing represent Task III. Reference 4 contains work on very large network design and is a summary paper of the work reported upon in our previous semiannual technical report. Reference 5 is a survey paper of progress in communications and networks. Reference 6 describes the deleterious effects of periodic routing to stream traffic, an example of which would be packetized speech; recognizing these effects, one can properly design systems to increase the efficiency in the face of such mixed traffic streams.

During this period, considerable research progress on the security task (Task IV) was made in the areas of operating system development, network security design, data management security, programming languages for secure computing, and system verification. Much of the research centered around the design, development and verification of the UCLA Data Secure UNIX Operating System. The Kernel of that system was completed, including the last stages of its design, complete implementation of its basic capabilities, and extensive debugging. The UNIX interface was designed, implemented, and subjected to initial debugging. Elementary versions of the system's schedules and file manager were coded and integrated with the rest of the system. The entire preliminary operating system, including all the modules mentioned above, were readied to permit installation at another U.S. Government site for testing. Documentation suitable for systems programmers' use of UCLA UNIX was produced to meet technology transfer desires as well as to support the projected government procurement of a production quality secure operating system. Progress in the verification of the UCLA system was composed of a revision of the system verification specifications, for publication in the professional literature, and proposals for changes to the ISI verification system. Those changes will be made by the ISI research group. A member of the UCLA research team participated in the development of a new programming language called Euclid, oriented to the implementation of verifiable systems programs. The language, derived from PASCAL, and motivated in part by UCLA verification experiences, was specified and its semantics axiomatized. The language developers also oversaw the accompanying commercial compiler implementation. Such activities in data management and network security also were ongoing during this period. The network activity has yielded a proposal for a secure architecture that both minimizes the amount of mechanism involved while still permitting retrofit to environments such as the ARPANET at modest cost. This work, as well as the other security research, will be coordinated with related ARPA activities.

Following is a list of publications, after which the main report on multiplexing techniques is given.

List of Publications

Abraham, Steven, "A Protection Design for the UCLA Security Kernel," Master's Thesis, University of California, Los Angeles, December 1976.

Gerla, Mario, "S-ALOHA Stability Control Algorithms," Packet Satellite Program Working Notes, #36, September 1976.

Gerla, Mario, "S-ALOHA Stability Control Algorithms and Simulation Experiments," Packet Satellite Working Notes, #46, December 1976.

Kleinrock, Leonard and Farouk Kamoun, "Data Communications Through Large Packet Switching Networks," Eighth International Teletraffic Congress, Melbourne, Australia, 10-17 November 1976.

Kleinrock, Leonard, "On Communications and Networks," IEEE Transactions on Computers, Vol. C-25, No. 12, December 1976, pp. 1326-1335.

Naylor, William and Leonard Kleinrock, "On the Effect of Periodic Routing Updates in Packet-Switched Networks," Proceedings of the National Telecommunications Conference, Dallas, Texas, November 1976, pp. 16.2.1-16.2.7.

Scholl, Michel, "Multiplexing Techniques for Data Transmission Over Packet Switched Radio Systems," Computer Science Department, University of California, Los Angeles, UCLA-ENG 76123, December 1976.

Tobagi, Fouad and Leonard Kleinrock, "Packet Switching in Radio Channels: Part III - Polling and (Dynamic) Split-Channel Reservation Multiple Access," IEEE Transactions on Communications, Vol. COM-24, No. 8, August 1976.

Tobagi, Fouad and Leonard Kleinrock, "Packet Switching in Radio Channels: Effect of Error Control Traffic on ALOHA and CSMA Channel Performance," Packet Radio Temporary Note #201, October 1976.

Tobagi, Fouad, "NETCAP: A Simulator for PR Net Channel Access Protocol Performance," Packet Radio Temporary Note, #203, December 1976.

Tobagi, Fouad and Yechiam Yemini, "NETCAP II: Derivation of Confidence Intervals for Various Performance Measures," Packet Radio Temporary Note, #204, December 1976.

ABSTRACT

We study the behavior of a population of geographically distributed data users who use packet-switching to communicate with each other and/or with a central station over a multiple-access broadcast radio channel. We introduce and analyze Alternating Priorities (AP), Round Robin (RR) and Random Order (RO) as methods for multiplexing a small number of buffered users, without control from a central station. In these access modes, we increase the channel utilization by allowing any user to "steal" the channel time assigned to a user who is idle (by sensing the carrier). These methods are effective when the number of users is not too large (less than 20) or when the propagation delay of the signal is small compared to the packet transmission time. Otherwise, a large overhead leads to a performance degradation. To reduce this degradation, we consider a natural extension of AP, called Mini-slotted Alternating Priorities (MSAP), which minimizes the overhead and performs better than existing techniques (such as Polling) for multiplexing a small number of buffered users over a single radio channel. In addition and of major importance is the fact that MSAP does not require control from a central station.

A second alternative for multiplexing the users on a packet switched radio channel is to include, on a single radio channel, the traffic of these users (called small users) as well as the traffic of a large buffered user. By dynamically sharing the channel among the two traffic classes, the required channel capacity is much less than in the unshared case of two dedicated channels. The large user within range and in line of sight of all small users "steals," by carrier sensing, the slots unused by the small users. Two access modes are studied which differ with respect to the nature of the population of small users: In the Mixed ALOHA Carrier Sense (MACS) access mode, a large population of bursty (small users) contend for the channel in a slotted ALOHA fashion. Not only is the small users' performance improved, but also the performance of the large user is better with MACS than it is when dedicated channels are assigned to the large user and the small users. The same idea is applied in the Mixed Alternating Priorities Carrier Sense (MAPCS) access mode. In MAPCS, a few buffered (small) users share the total available bandwidth under the AP protocol. Although MAPCS provides a very good performance to the small users, the large user's performance is greatly affected by the presence of the small users.

The introduction and implementation of packet radio techniques is likely to have a large impact on the data communication field. Studies such as this one are important in that they contribute to those techniques.

CONTENTS

CONTENTS (continued)

CONTENTS (continued)

CONTENTS (continued)

LIST OF FIGURES

LIST OF FIGURES (continued)

LIST OF FIGURES (continued)

CHAPTER 1

INTRODUCTION

The constantly growing need for access to computers, communication channels and distributed computer communication networks* renders the problem of allocating these large, expensive resources among an ever increasing number of users a rather fundamental one. It is projected that by 1980 approximately four to five million data terminals will be in use in the United States [KLEI 74], and it is estimated that the data traffic in seventeen western European countries will grow by a factor of six from 1972 to 1980 [EURO 73]. Kleinrock describes the resource allocation problem as follows [KLEI 74]:

> "A means for allocating these resources in order to resolve the conflicting demands is one of the most important aspects of today's system design and operation. In fact, resource allocation is at the root of most of the technical (and non-technical) problems we face today in and beyond the information processing industry. These problems occur in any multi-access system in which the arrival of demands as well as the size of the demands made upon the resources are unpredictable. The resource allocation problem in fact becomes that of resource sharing and one must find a means to effect this sharing among the users in a fashion which produces an acceptable level of performance."

A study was made of ~~We restrict our attention to~~ the allocation of a data communication channel; the demands placed upon this scarce resource ~~are~~ as made by a population of users (terminals), with the following characteristics:

* An example of which is the ARPANET (The Advanced Research Projects Agency Network) [CARR 70, FRAN 70, HEAR 70, KLEI 70, ROBE 70].

- The demands are unpredictable and bursty [KLEI 74]*
- The users are possibly geographically scattered.

One of the major functions of the data communication channel is to provide to the users access to information processing and storage capacity available from a computer or a network of computers. We will also consider the communication channel as a media providing direct communication among the users (terminals) themselves. In any case, the users are connected with each other (and/or with a central node) by means of a communication network referred to as the terminal access network.

If the major function of this network is to provide access to a local computer (or to a remote computer through a long-haul computer-computer communication subnetwork), then the terminal access network is referred to as a centralized network. By centralized network, it is understood that all demands made upon the channel are made either by a terminal which needs access to the central node (computer) or by the computer which wants to be connected with any terminal (Figure 1.a).

If the function of the terminal access network is also to provide a means by which terminals can directly access each other, we are in the presence of so-called point-to-point communication networks (Figure 1.b). In such networks, the transmission's control is distributed among the terminals.

---

*Measurement studies [JACK 69] conducted on time-sharing systems indicate that data streams are bursty, i.e., the peak data rate is much larger than the average data rate; the ratio between them may be as high as 2000 to 1 [ABRA 73].

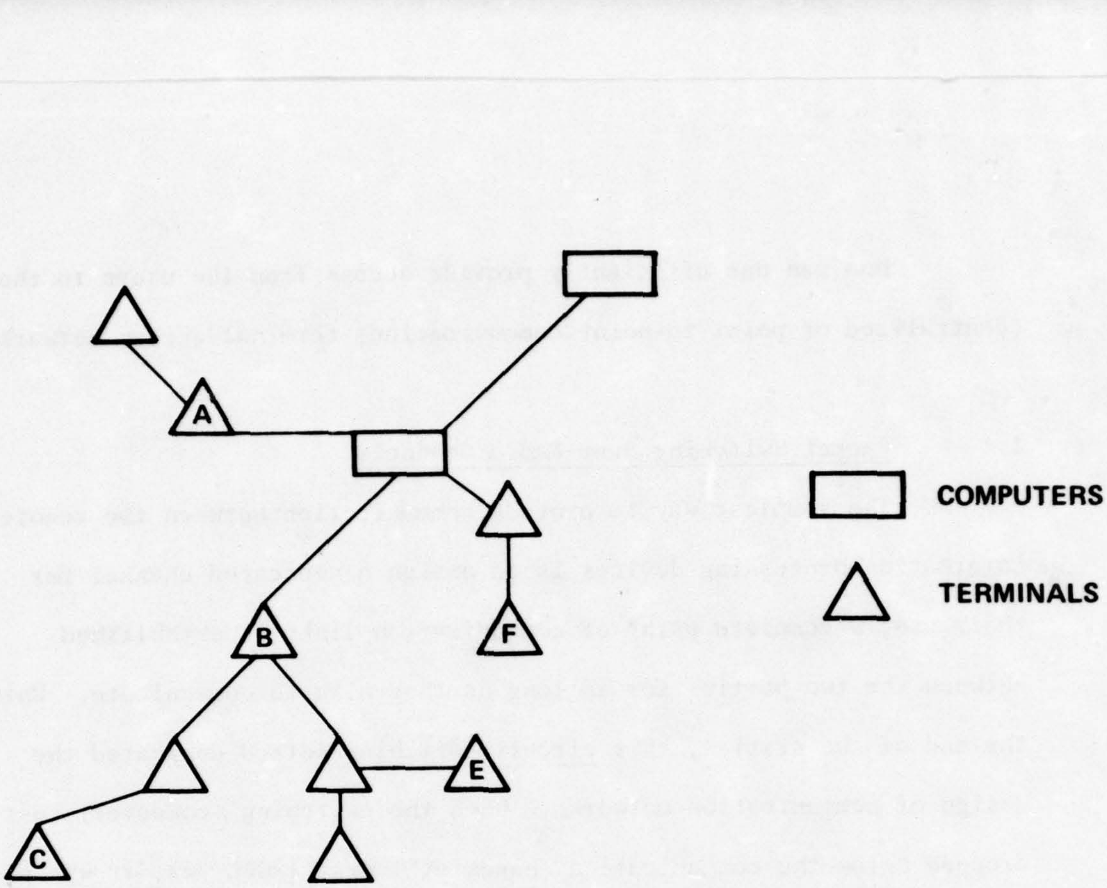


Figure 1.1a. **Centralized Terminal Access Network. All communications are directed *to or from the computer.***

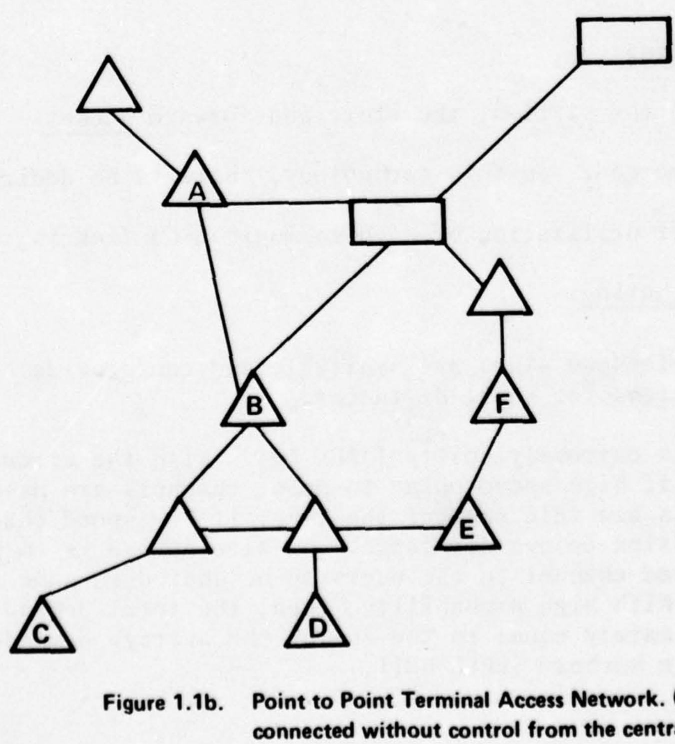


Figure 1.1b. **Point to Point Terminal Access Network. (A,C), (B,D), (C,F), etc. . ., are connected without control from the central computer.**

How can one efficiently provide access from the users to the (centralized or point-to-point communication) terminal access network?

1.1 Packet Switching over Radio Channels

The simplest way to provide communication between two remote information processing devices is to assign a dedicated channel for their use; a complete path* of communication links is established between the two parties for as long as they wish to communicate. Until the end of the sixties, this circuit-switching method dominated the design of communication networks. When the switching (computer) cost dropped below the communication (bandwidth) cost [ROBE 74], it was possible to develop a cost-effective communication means as an alternative to the widespread, expensive circuit-switching technology†. This led to the packet-switching technology.

1.1.1 Packet-Switching

At the end of the sixties, the store-and-forward packet-switched technology emerged. In this technology, there is no dedication of resources. A fuller utilization of each communication link is obtained by resource sharing.

---

*Dial-up and leased telephone lines are available and can provide inexpensive communications for short distances.

†The users' behavior is extremely bursty [JACK 69]. With the circuit-switching technology, if high speed point-to-point channels are used, the communication links are idle most of the time; if low speed channels are used, the transmission delays are large. An alternative is to provide a single high speed channel to the users to be shared in some multi-access method. With high probability, then, the total demand at any instant is approximately equal to the sum of the average demands of all users (Law of Large Numbers [FELL 68]).

The communication links are not allocated into paths for specific source-destination pairs of nodes; instead, each link is statistically shared by messages from different source-destination pairs. In addition, each message can be broken into packets of information with the addresses of the source and destination attached to each packet. Packets are individually routed through the network to their destination by "hopping" from one node to another. The required channel capacity is much less than in the unshared case of dedicated channels. This concept has not only been applied in distributed computer-communications networks (ARPANET [BARA 64], [KLEI 64]) but also in some wire communication access schemes. These schemes include Asynchronous Time Division Multiplexing (ATDM) [CHU 69], loop systems [HAYE 71], [PIER 71] and Polling [MART 72], [KONH 74].

More recently, packet switching techniques have been applied to radio communications (both satellite and ground radio channels) [KLEI 76]. Packet-switched random access schemes were introduced as an alternative for terminal and computer communications. These schemes include ALOHA [ROBE 72, ABRA 73, LAM 74, KLEI 75A] and CSMA [TOBA 74], [KLEI 75B]. One of the first packet radio communication systems was the ALOHA system developed at the University of Hawaii [ABRA 70], [KUO 73]. The ALOHA network originally applied packet radio techniques for communication between a central computer and its geographically scattered terminals. Terminals are now minicomputers, communicating among themselves through the central computer and are able to access all computing resources of the ARPANET. The Advanced Research Projects Agency of the Department of Defense has undertaken a new effort whose goal is to develop new techniques for packet radio communication among geographically distributed (possibly) mobile user terminals [KAHN 75]. A packet radio broadcast network is currently developed as an interface between a point-to-point wire network (like the ARPANET) and a number of geographically scattered terminals. This broadcast system provides local collection and distribution of data over large geographical areas [KAHN 75, KLEI 75B, BIND 75C, FRAN 75 FRAL 75A, FRAL 75B, BURC 75]. Furthermore, there is currently an immense worldwide interest in the development of satellite communications systems [PUEN 71, ABRA 73, CROW 73, KLEI 73, ROBE 73, TELE 73, GRAY 74, LAM 74, KLEI 75A].

At last, broadcast packet switching may be applied to wire communications. The Mitre's Mitrix [WILL 73], the U.C. Irvine's Distributed Computing System (DCS) [FARB 73] and the Xerox's Ethernet [METC 76] are examples of local computer networking experiences.

Ethernet is an experimental wire packet-switched broadcast network for local communication among computing stations. It uses tapped coaxial cables to carry variable length data packets among, for example, minicomputers, printing facilities, file storage devices, larger central computers and longer-haul communication equipment.

The shared communication facility called Ether [METC 76] is a passive broadcast medium with no central control. Coordination of access to the Ether is distributed among the contending transmitting stations*. Switching of packets to their destinations on the Ether is distributed among the receiving stations using packet address recogni-

* DCS [FARB 73] like Ethernet uses distributed control, while Mitrix [WILL 73] has a central station for switching and bandwidth allocation.

tion. Ethernet can be extended by using repeaters for signal regeneration.

Packet transmissions are coordinated through statistical arbitration from the tranceivers. Collisions are avoided by carrier sensing (no station will start transmitting while hearing the carrier) and by interference detection (interference occurs when two or more stations detect no carrier and start transmitting simultaneously). If the transceiver notices a difference between the bit it is receiving from the Ether and the bit it is transmitting (Interference), the transmission is aborted and rescheduled.

An operating Ethernet of 100 nodes along a kilometer of coaxial cable has been experimented with. This experience encourages the development of numerous computer networking and multiprocessing applications.

In this dissertation, we shall focus our attention on data communication over packet-switched ground* radio systems as an alternative for data transmission among terminals and computers. Of interest to this research is the consideration of a single high-speed radio channel shared in some multi-access scheme, and in a packet-switching mode. The problem we are faced with is how to share and how to control access to the channel in a fashion which provides an acceptable level of performance. A number of multiple access schemes exist that we have already mentioned above. Some of the access schemes for wire communications may also be applied to a radio environment with various degrees of success. However there is a need for new multiple-access methods.

---

*We exclude here the study of satellite communications systems.

This will be made clear by the study in Section 1.2 of existing multiple access schemes.

For such data communications among terminals and computers, wire connections could have been used. In the next section, we present the numerous advantages in using packet-switched radio communications over conventional wire communications.

1.1.2 Advantages of Radio Communications

One of the most important properties of radio communications is the broadcast property: Any number of users may access the channel and the transmission of a signal by a user may be received over a geographically wide area by any number of receivers. Consequently, the collection and distribution of data over large areas are independent of the availability of pre-existing wire networks. However if two (or more) signals at the same carrier frequency overlap in time, this may result in information destruction (interference).

In addition a broadcast mode is particularly suitable when the users are mobile or are located in remote regions where a wire connection is not easy to implement.

The second major advantage of broadcast radio systems is related to design flexibility. Provided that all users are in line of sight and within range of each other, the provision of a completely connected network topology by a radio channel* eliminates complex topological design and routing problems [FRAN 72], [GERL 73]. Moreover,

---

*due to the broadcast and multiaccess properties.

the size of the system and the location of the users are easily modified.

Security problems and the need for legal and regulatory procedures may present difficulties in the use of broadcast radio communications; we do not consider those issues here.

1.2 Existing Multiplexing Techniques

Here we describe the main techniques which have been implemented (or proposed) for packet switching multiple access over a single radio channel; in the figures below we compare their delay-throughput profiles. These multiplexing techniques are Time Division Multiple Access (TDMA), Frequency Division Multiple Access (FDMA), Polling, ALOHA random access (pure and slotted), Carrier Sense Multiple Access (CSMA), and finally a reservation technique referred to as Carrier Sense Split Reservation Multiple Access (CS SRMA).

For the purpose of this simple comparative study, we consider a population of N statistically identical users communicating with a "master" user (e.g., a central station) over a radio channel of limited bandwidth W (bits/sec). Each user is assumed to generate fixed size packets according to a Poisson point process with intensity λ/N; we assume that the full packet is instantaneously generated at those points. The radio channel is characterized as a wideband channel with a propagation delay between any source-destination pair which is very small compared to the packet transmission time*. Each technique is

*On the contrary, when one considers satellite channels [LAM 74], the propagation delay is a relatively large multiple of the packet transmission time.

characterized by its maximum achievable throughput (also referred to as the channel capacity), where the throughput S is defined as the average number of packets transmitted during a period of time equal to a packet transmission time T ($T = b_m/W$ if b_m is the number of bits per packet); under equilibrium, S is equal to the average input rate normalized with respect to T: $S = \lambda T$. The main performance measure we consider in our comparative study is the average normalized packet delay D versus the throughput S, where the normalization is with respect to T.

1.2.1 Fixed Assignment: FDMA, TDMA

In FDMA, each user is assigned a fraction W/N of the channel, along with buffering capabilities required to handle the statistical fluctuations due to the random packet arrival instants. The system can be modelled by M/D/1 queues in which the service time is the constant transmission time of a packet on one of the channels. We then have by the Pollaczek-Khinchin formula [KLEI 75C]

$$D = N\left[1 + \frac{S}{2(1 - S)}\right] \qquad (1.1)$$

In Time Division Multiple Access (TDMA), also called Synchronous Time Division Multiplexing (STDM), each user is assigned a periodic sequence of time slots on the channel. The channel slots are usually switched to users in a round-robin (i.e., cyclic) fashion. The system can be modelled by M/D/1 queues with a rest period (see Appendix A), in which the service time is equal to N slots (N slot times are required to "serve" a packet generated at a given user; 1 slot is the actual transmission time of the packets and (N - 1) slots are assigned to the other (N - 1) users). The rest period is equal to N slots. We then have (see Appendix A):

$$D = 1 + N\left[\frac{S}{2(1 - S)} + \frac{1}{2}\right] \tag{1.2}$$

Note that the average delay with FDMA is higher than the average delay with TDMA, the difference being $\frac{N}{2} - 1$; with TDMA, the slotting effect counts for $\frac{N}{2}$, but in FDMA the transmission time of a packet is N, while it is only 1 with TDMA.

Both FDMA and TDMA are inefficient in channel utilization since the bandwidth or time slots are allocated to each user independently of his activity; it is inefficient to permanently assign a segment of bandwidth that is utilized only a portion of the time. For a population of users whose requests for the channel are very infrequent (i.e., bursty users), the probability is not negligible that a large portion of the channel bandwidth (FDMA) or channel time (TDMA) is unused (infrequent requests) and thus wasted for the requests waiting at other terminals. The latter incur a higher delay than they would incur if they had access to the wasted part of the channel. Indeed, the channel assignment is based on an average request rate (much lower than the peak request if these requests are highly infrequent).

A number of disadvantages of FDMA exist when compared with TDMA: wasted bandwidth for adequate frequency separation, lack of flexibility in achieving dynamic allocation of bandwidth, and lack of broadcast operation. The major disadvantage in TDMA is the need to provide rapid burst synchronization and sufficient burst separation to avoid time overlap.

In Figures 1.2 and 1.3 we plot the average normalized delay in TDMA versus the throughput for two values of the number of users, namely N = 10 and N = 50, respectively.

In order to increase the channel utilization beyond FDMA and TDMA, statistical multiplexing or Asynchronous Time Division Multiple Access (ATDM) has been proposed [CHU 69]; this is simply a single M/D/1 queue (possibly with burst arrivals). Basically this technique consists of switching the allocation of the channel from one user to another whenever the former is idle and the latter is ready to transmit. ATDM is suitable in wired terminal-access networks when the terminals are concentrated and gives very small expected delays. It is less attractive when radio communication is employed since the connections between the terminals and the multiplexor then require some synchronous techniques such as FDMA or TDMA and the performance of the system will be limited by the performance of those synchronous techniques. Moreover, it is inadequate in situations where the terminals are geographically scattered and want to access each other (point-to-point networks).

1.2.2 Polling

In the polling technique known as Roll-Call Polling [MART 70], a central station asks (polls) the N users one by one in sequence* whether they have anything to transmit. If the user has some packets to transmit, he goes ahead; if not, a negative reply (or absence of reply) is received and the next user is polled. Packets generated at a user are queued in his buffer (in a first-come-first-served basis)

*The central station broadcasts a polling message containing the identification of the user.

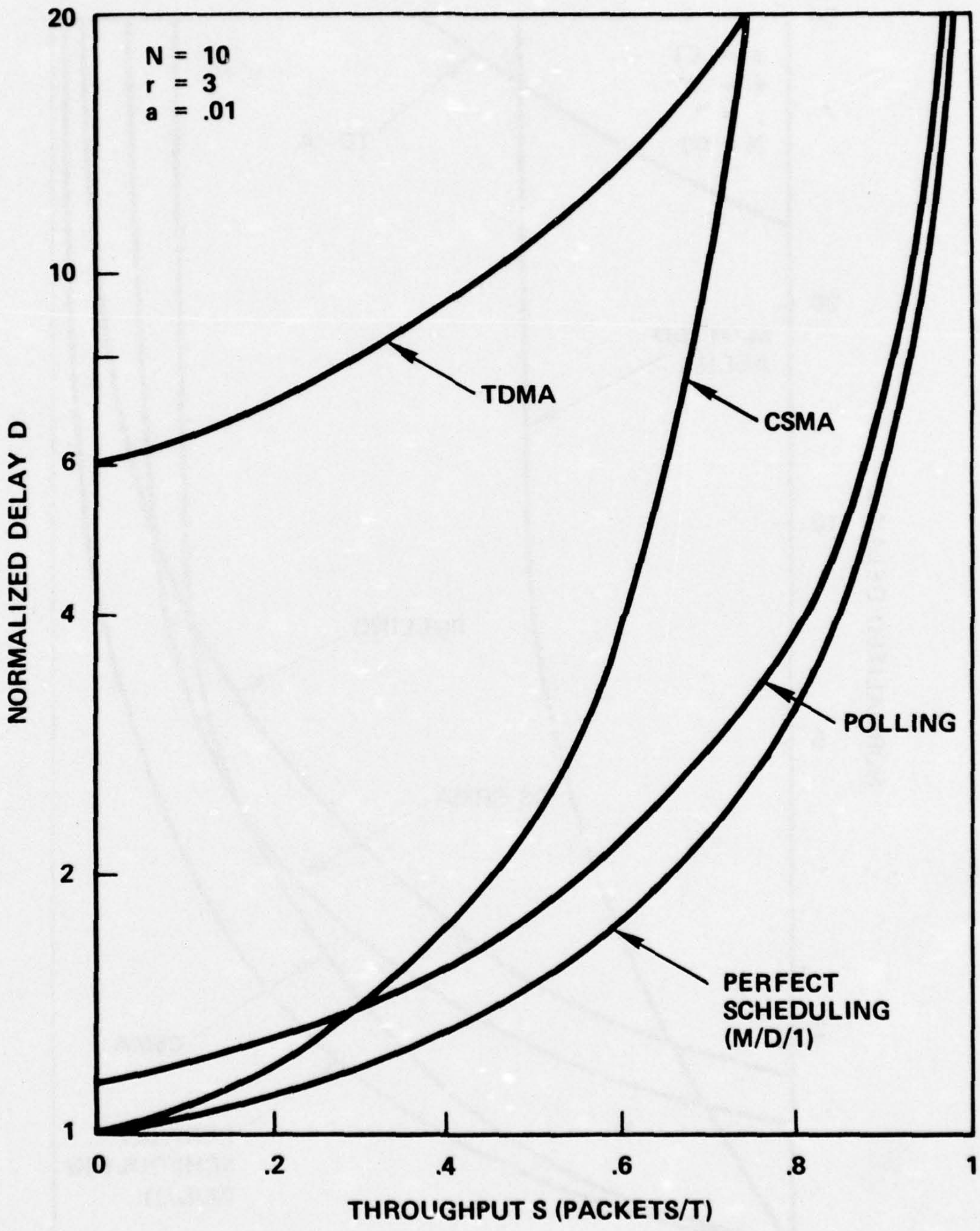


Figure 1.2. Expected Packet Delay versus Throughput: Small Number of Users.

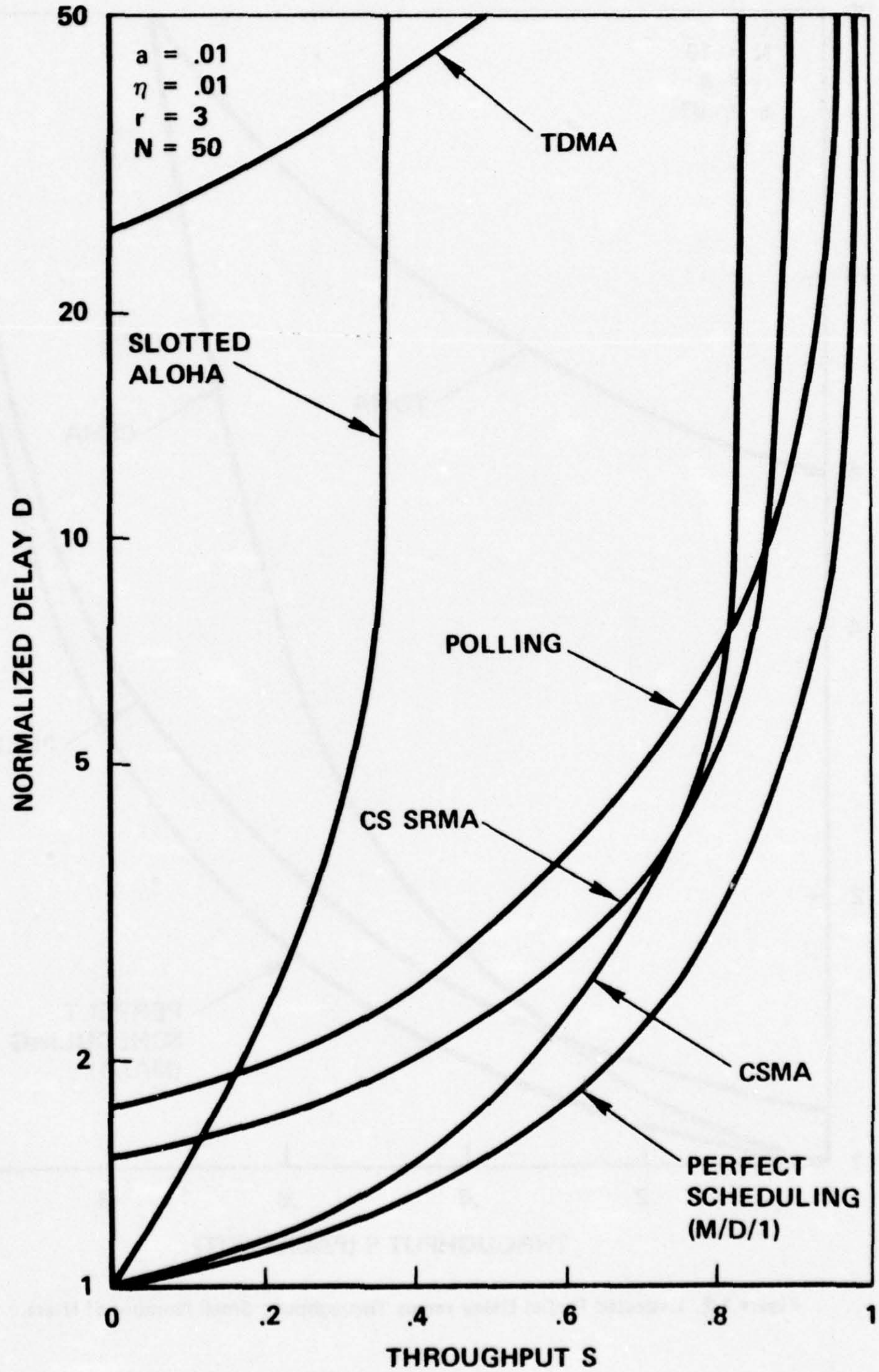


Figure 1.3. Large Number of Users: D vs S

until the user is polled, at which time the buffer is completely emptied (the user captures the channel until he has no packet to transmit).* The time axis is divided into slots of equal size and equal to the maximum one-way propagation delay τ from the users to the station. τ constitutes only a very small fraction ($a \ll 1$) of the transmission time of a packet T. Konheim and Meister [KOHN 74] analyzed this polling technique, deriving stationary distributions for queue lengths and waiting times. In an application of their results to packet radio, Tobagi [TOBA 76A] showed that the expected packet delay (normalized with respect to the packet transmission time T) is given by

$$D = 1 + \frac{S}{2(1 - S)} + \frac{a}{2}\left(1 - \frac{S}{N}\right)\left(1 + \frac{Nr}{1 - S}\right) \qquad (1.3)$$

where $a = \tau/T$ ($\ll 1$) and $r = 2 + T_p/\tau$

T_p = transmission time of a polling message.

In Figures 1.2 and 1.3, the expected delay is plotted versus the throughput for N = 10 and N = 50 respectively. Eq. (1.3) shows that the delay is the sum of the Pollaczek-Khinchin terms (for M/D/1) [KLEI 75C] plus an additional term due to the polling time. When N is not too small ($N \geq 10$), the average polling time is approximately proportional to Nra for a given throughput. With polling, a throughput of 1 is achievable (with infinite delays). Indeed, when the traffic is very high each user captures the channel for long periods, and therefore transmits packets at the rate of 1 packet/packet transmission time. It is also noteworthy that even for small values of N ($N \approx 20$), the

* "Hub Go Ahead" Polling is an alternative to Roll Call Polling which is advantageous on long lines communications [MART 70] but is not readily applicable to our radio system and will not be considered throughout the dissertation.

expected queue length is less than 1 packet/user for most values of throughput ($S < .95$) [TOBA 74].

1.2.3 ALOHA Random Access Techniques

The ALOHA system [ABRA 70, KUO 73] is operational and appears to have been the first computer-communication system to employ wireless connections. Each user transmits packets to the central computer over the same high-speed data channel in a completely unsynchronized manner. Errors are due to two major causes: (1) random noise on the channel, and (2) interference at the receiver with packets transmitted by other terminals. If, and only if, a packet is received without error is it acknowledged by the central station. After transmitting a packet, the terminal waits a given amount of time for an acknowledgement; if none is received, the packet is retransmitted. This process is repeated until successful transmission and acknowledgement occur, or until the process is terminated by the terminal. This random-access packet switching technique is referred to as pure ALOHA. In this mode, interference due to overlapping packets definitely prevents a channel utilization of 100%. (Whenever a portion of one user's transmission overlaps another user's transmission, the two collide and "destroy" each other.) For a fixed packet size, the channel capacity was calculated to be $1/2e = 18\%$ [ABRA 70]. Roberts [ROBE 72] suggested modification of the completely unsynchronized use of the channel by slotting time into segments whose duration is equal to the transmission time T of a single packet. If we require each user to start his packet's transmission only at the beginning of a slot, then when two packets conflict, they overlap completely rather than partially. This provides an increase in channel efficiency, and the channel capacity is $1/e \cong 36\%$. This technique is referred to as slotted ALOHA. Those random access modes assume a large number N of users who generate packets infrequently and each packet is assumed to be successfully transmitted in a time interval much less than the average time between successive packet generations at a given user*. Thus, for performance analysis purposes, the large population of users is approximated by an infinite population, each user of which has at most one packet requiring transmission at any time (including any previously blocked packet, i.e., any packet which previously "collided" and which is in the process of being retransmitted). The traffic offered to the channel from the population of users consists of newly generated packets and previously collided packets, with a mean offered traffic rate denoted by G ($\geq$ S, the mean input rate). Abramson [ABRA 70] first showed for pure ALOHA that

$$S = Ge^{-2G} \tag{1.4}$$

Thus in pure ALOHA, a maximum throughput is achieved at $G = 1/2$ and equal to $1/2e \cong .184$.

Roberts [ROBE 72] extended Abramson's result for slotted ALOHA. The throughput equation becomes:

$$S = Ge^{-G} \tag{1.5}$$

The maximum throughput is increased by a factor of two to $1/e \cong .368$ (at $G = 1$). From these results, it is all too evident that a significant fraction of the channel's ultimate capacity ($C = 1$) is not

---

*This will be assumed to be true when the number of users is not too small, say $N \geq 20$ (with Polling, when $N = 10$, the expected queue length is greater than 1 only if the throughput S is greater than .9).

utilized with the ALOHA access modes*. Slotted ALOHA has been analyzed by LAM [LAM 74]. Neglecting the propagation delay and letting the maximum retransmission delay be an integer number K of slots, the expected normalized delay is then given by [LAM 74]

$$D = 1 + \frac{E(K+1)}{2} \tag{1.6}$$

where $E = \dfrac{1 - q_n}{q_t}$

$$q_n = \left[e^{-G/K} + \frac{G}{K} e^{-G}\right]^K e^{-S}$$

$$q_t = \frac{e^{-G/K} - e^{-G}}{1 - e^{-G}} \left[e^{-G/K} + \frac{G}{K} e^{-G}\right]^{K-1} e^{-S}$$

$$S = G \frac{q_t}{q_t + 1 - q_n}$$

For each value of S, an optimum value of K can be selected so as to achieve minimum delay. In Figure 1.3 (N = 50), the lower envelope of all constant-K delay curves is plotted versus the throughput. From this figure it is clear that for light traffic, slotted ALOHA provides delays far better than those obtained with TDMA (or FDMA)†.

As discussed earlier (Section 1.2.1), FDMA and TDMA are

* The ALOHA random access with FM capture (FM capture occurs if, when two signals collide, the most powerful is received correctly) may result in a channel capacity larger than 1/e [ROBE 72].

† We did not compare the slotted ALOHA performance to that obtained with TDMA and Polling, for N = 10, since the ALOHA performance predicted by the infinite population model is not accurate enough for such a value of N. Indeed this model assumes that the expected time to successfully transmit a packet is less than the average time between successive packet generations at a given user. This may not be true for N = 10; for a small number of users ($N \leq 20$) buffering capabilities are needed.

inefficient in channel utilization, and the motivation behind the interest in using random access ALOHA modes (despite their low channel capacity) over TDMA and FDMA was to provide a single high-speed data rate channel to be shared by all users. Because of the bursty nature of the traffic, providing the entire bandwidth (scarce communication resource) to be shared by all users is shown (see [KLEI 76]) to be far superior to FDMA or TDMA.

When we are in the presence of bursty users, slotted ALOHA can support many more users than FDMA or TDMA at the same packet delay; in order to support a large number of users FDMA and TDMA require a larger bandwidth for the same delay performance. This motivated further consideration of new protocols (CSMA, CS SRMA) for which the maximum throughput is higher than under ALOHA access modes.

Before we consider CSMA and CS SRMA, we note an important characteristic of all random access modes: After some finite time period of quasi-stationary conditions, the channel will drift into saturation with probability one, i.e., the throughput will go to zero, while the channel load will increase without any bound. A theory has been proposed [KLEI 75A] which characterizes the instability phenomenon by defining stable and unstable channels and control policies for unstable channels have been proposed and analyzed. In stable channels, the throughput-delay results obtained under the channel equilibrium assumption are achievable over an infinite time horizon, while in unstable channels such channel performance is achievable only for some finite time period before the channel goes to saturation. By applying dynamic channel control policies, a channel throughput-delay performance close to the optimum performance envelope is achievable over an infinite time horizon for originally unstable channels.

1.2.4 Carrier Sense Multiple Access Mode (CSMA)

The radio channel is considered as a wide band channel with a propagation delay between any source-destination pair which is very small compared to the packet transmission time. This suggests a new approach for using the channel, namely the Carrier Sense Multiple Access mode*. In this scheme, one attempts to avoid collisions before transmitting by listening to the carrier due to another user's transmission. Based on this information about the state of the channel, one may think of various actions the user may take. Kleinrock and Tobagi introduced and analyzed three protocols [KLEI 75B] which differ by the action that a terminal takes after sensing the channel. In all cases, when a user determines (by the absence of a positive acknowledgement) that his transmission was unsuccessful, he then reschedules the transmission of his packet according to a randomly distributed retransmission delay. The three protocols are referred to as 1-persistent, p-persistent and non-persistent CSMA. As an example we consider below the Non-Persistent CMSA protocol.

- If the channel is sensed idle, the user transmits its packet.
- If the channel is sensed busy, the user schedules the transmission of the packet at some later time, at which point in time, it senses the channel and repeats the algorithm described.

---

*Sensing the carrier prior to transmission is a well-known concept in use for (voice) aircraft communication. In the context of packet radio channels, it was originally suggested by D. Wax of The University of Hawaii in an internal memorandum dated March 4, 1971.

A slotted version can be considered by slotting the time axis; the slot size is τ (the maximum propagation time). All users are synchronized and are forced to start transmission only at the beginning of a slot. When a packet's generation occurs during a slot, the user senses the channel at the beginning of the next slot and operates according to the protocol described above. *Slotted non-persistent* CSMA protocol provides the best performance among all CSMA protocols; this is the one we will consider and refer to throughout the dissertation.

The performance of CSMA modes is highly dependent on the sensing ability of each user; unfortunately many situations exist in which some users are "hidden" from others, either because they are out-of-sight or out of range*

Using the same infinite population model as that used with the ALOHA modes†, Kleinrock and Tobagi [KLEI 75B] derived the throughput equations for all CSMA protocols. The equations for the throughput S are expressed in terms of a (the ratio of maximum propagation time to packet transmission time) and G (the offered traffic rate: $G \geq S$). For slotted non-persistent CSMA, S is given by

$$S = \frac{aGe^{-aG}}{(1 + a)(1 - e^{-aG}) + a}$$

---

*To eliminate the hidden terminal problem, a natural extension of CSMA has been proposed and analyzed [TOBA 75], namely the Busy Tone Multiple Access mode (BTMA), which performs almost as well as CSMA without hidden terminals.

†In this model, the traffic source consists of a very large population of bursty users who collectively can be approximated by an infinite population, each user of which has at most one packet requiring transmission at any time. The traffic offered to the channel consists of newly generated and previously collided packets.

The channel capacity C is obtained by maximizing S with respect to G. Slotted non-persistent CSMA provides the largest capacity; for example, when a = .01, then C = .857. We note that most of the channel capacity which was unavailable with ALOHA is recovered with CSMA. However, when a is very large (a > .3), slotted ALOHA is superior to any CSMA mode [KLEI 75B].

The CSMA modes, as with all random multi-access broadcast channels, are characterized by the fact that after some finite time of quasi-stationary conditions, the channel will drift into saturation. The stability theory introduced in the study of ALOHA [LAM 74] has been extended by Tobagi [TOBA 76B] for CSMA modes. The interesting result is that whatever the number N of users may be, one can always find an optimal value of the average retransmission time (drawn from a geometric distribution) so that the channel is stable (i.e., the throughput-delay results obtained under the channel equilibrium assumption are achievable over an infinite time horizon). The model considered for this study assumes a finite population of N users, among which n are blocked (i.e., in the process of transmitting a packet) and (N - n) are thinking* (generating packets at a global rate of $(N - n)\sigma$). It is shown that up to N = 1000, the delay-throughput performance obtained (a stable channel) is better than that predicted by the infinite population model (an unstable channel). For larger values of N, in order to get a stable channel, a delay degradation is observed due to the large increase in retransmission delay.

*The average thinking time is denoted by $1/\sigma$.

In Figure 1.3, slotted non-persistent CSMA performance (stable channel for $N = 50$ users) is compared to TDMA, Polling and ALOHA. When the traffic load is not too heavy ($S < .8$), CSMA performs much better than Polling; the situation reverses beyond $S \cong 0.8$.

For $N = 10$ users, the CSMA model (no buffering capabilities: each user has at most one packet requiring transmission at any time) is not accurate enough to allow us to compare the performance predicted by this model to TDMA and Polling.

Tobagi and Kleinrock [TOBA 76C] modified the slotted non-persistent CSMA model by considering a (small) number of buffered terminals and a geometric retransmission time after collision (a terminal which has a non-empty queue will avoid repeated conflicts by transmitting the packet at the head of the queue in the next slot with probability p). Simulation results [TOBA 76C] show that the throughput-delay performance deteriorates as the number of terminals N increases; with $N = 5$, the simulation shows that we already reach the "infinite population of unbuffered terminals" performance.

Therefore in Figure 1.2 ($N = 10$), we plot the delay-throughput performance as predicted by this infinite population model, as an approximation of the performance obtained by $N = 10$ buffered users communicating over the radio channel under the slotted non-persistent CSMA protocol. From Figure 1.2, it is clear that Polling performs better than CSMA for a small number of users ($N \leq 10$).

1.2.5 Reservation Techniques

Several reservations schemes based on the slotted ALOHA random access mode have been proposed for satellite packet-switching systems

[BIND 72, CROW 73, ROBE 73]. A hybrid technique has been proposed for satellite packet-switching systems which combines a reservation technique with a dynamic allocation of the channel [BIND 75B]. In packet radio environments, characterized by short propagation delays, a possible implementation of the reservations techniques is performed by dividing the available bandwidth into two channels: one used to transmit requests for reservations, the second one used for the (information) packet traffic itself. This gives rise to the Split-Channel Reservation Multiple Access (SRMA) modes [TOBA 76A]. In SRMA, the user makes a request on the channel whenever he has a packet to transmit. When a request is received, the station schedules the allocation of the channel to the user placing the request. Random multiple access modes (ALOHA or CSMA) are chosen as the method of multiplexing the requests on the channel. We will consider below only the case where requests compete in a CSMA mode, referred to as Carrier Sense SRMA (CS SRMA), since this latter mode gives the best performance. In CS SRMA, the bandwidth is divided into three sub-channels:

- the request channel
- the answer-to-request channel
- the (information) packet channel.

The request channel is operated as in CSMA. At the correct reception of the request, the station computes the time at which the information packet channel will be available and transmits back to the terminal, on the answer-to-request channel, the time at which it can start transmission. Given the ratios $\eta_r = b_r/b_m$ and $\eta_a = b_a/b_m$ where b_m, b_r and b_a are the number of bits respectively in a (message) packet, a request and an answer-to-request, the optimal splitting of the channel into three parts is computed so as to maximize the bandwidth utilization and to minimize the delay (for a given load) [TOBA 76A]. When η_r and η_a are small (in Figure 1.3, we choose a = .01 and $\eta_r = \eta_a = \eta = .01$), the channel capacity in CS SRMA is found to be very close to 1 (C = .95 for $\eta = .01$).

CS SRMA provides a significant improvement over CSMA, not only in terms of maximum achievable throughput, but also in terms of delay for a given throughput. We plot in Figure 1.3 the performance predicted by the analysis [TOBA 76A] assuming a large population of users (infinite population, unstable channel). This should be an upper bound on the performance when N = 50*. CS SRMA performance has not been plotted in Fig. 1.2 since the model that predicted this performance is not suitable for a small number of buffered users.

In comparing CS SRMA to Polling, we note from Figure 1.3 that when N is large and when the throughput is not too close to 1 (e.g., $S \leq .88$ with N = 50), CS SRMA exceeds the performance of Polling because of the overhead due to control information in Polling. (From Eq. (1.3) we know that the average polling time is proportional to Nra.)

1.3 Summary of Results

Random access techniques (such as ALOHA, CSMA or CS SRMA) have

* The upper bound will hold if the stable channel model defined for CSMA [TOBA 76B] also holds for CS SRMA. However for larger values of N, it is possible that we get higher delays than those predicted by the infinite population model. Indeed, with increasing values of N, the operating value of the retransmission (after collision) time K, which renders the channel stable, must be increased. This leads to significantly increasing delays.

previously been analyzed with the assumption of an infinite population of unbuffered terminals. In the presence of a large (but finite) number of users, this model is accurate in the sense that the assumption upon which it is based is realistic. This assumption is that the inter-generation time between requests is long enough so that there is no need for buffering capabilities.

However, when we have a population composed of a very small number ($N \leq 10$) of users, it is necessary to provide those users with buffering capabilities. The random-access modes discussed above may then not be suitable for such an environment, since they assume that a user cannot generate a packet while he is in the process of transmitting a previously generated packet*.

Among those models discussed in the previous section which account for buffering (i.e., TDMA, FDMA and Polling), we see that Polling uses the channel efficiently and gives the best delay. However the main disadvantage of Polling is the requirement of a "master" user (central station) which controls the access to the channel. This control is not desirable for reliability and security reasons; in point-to-point communication; and in mobile environments (possible reassignment of the master user). In addition, when N increases, the polling time increases, which leads to delays worse than those to be expected with random access techniques when the throughput is not too close to 1.

*We already mentioned that the performance of a small number of buffered users communicating over the radio channel under CSMA (as well as under ALOHA) has been simulated [TOBA 76C]. However neither an exact solution nor a good analytic approximation has yet been found that could give a good prediction of this system's performance.

Consequently, our first goal is two-fold:

(1.a) For a population composed of a small number of users requiring buffering capabilities, we wish to introduce and analyze multiple access modes over a single radio channel which do not require control from a central station. Our approach is to introduce a new class of access modes which, in all but one case, are conflict free; in particular, we do not treat the conflict-prone systems (ALOHA and CSMA).

(1.b) For a population of a larger number of buffered users, we shall introduce and analyze new multiple access modes over a single radio channel which do not require control from a central station, and the performance of which is comparable to that obtained with random access techniques like CSMA.

In the previous section, we discussed the advantage of random access modes over fixed assignment schemes (TDMA, FDMA). Although slotted ALOHA only provides a maximal throughput of 1/e, it has the following advantages over CSMA:

- Implementation is simple.
- ALOHA does not require that all users be in line-of-sight and within range of each other. This is an important consideration.

When we are in the presence of a large number of small users (terminals) which do not "hear" each other, slotted ALOHA provides excellent delays and an efficient channel utilization at low traffic; however, we are dismayed that the maximal achievable throughput is only 1/e.

Therefore, the second goal of the dissertation is

(2) In order to increase the channel capacity, we shall analyze the case where we include traffic from a large user (e.g., a computer) as well as traffic from a large population of small users (e.g., terminals communicating with the computer) over a single channel.

In Chapter 2, we develop a general channel configuration model. The simplifying assumptions are specified, the traffic model is characterized and operational features are presented.

As a response to our first objective (Goal (1.a)), we introduce, analyze and compare new access modes in Chapter 3, which use the channel much more efficiently than TDMA. The delay-throughput performance of these new schemes is slightly larger than that obtained with Polling (see Figure 1.4). At light traffic, CSMA provides slightly shorter delays than the new schemes. Under heavy traffic conditions, the new schemes perform better than CSMA (see Figure 1.4). As in CSMA, we use the ability of each user to detect the presence or the absence of the carrier τ seconds after the beginning of the transmission, where τ is the maximum propagation time between any source-destination pair.

In these new modes, the time axis is slotted. Each slot is assigned to a user to transmit a packet, and contains an overhead, during which the other users will be able to "sense" the carrier. If the carrier is not detected, another user will transmit a packet over this slot. Therefore by carrier-sensing, we avoid as much as possible, wasted channel time, and thus we use the channel much more efficiently than with TDMA.

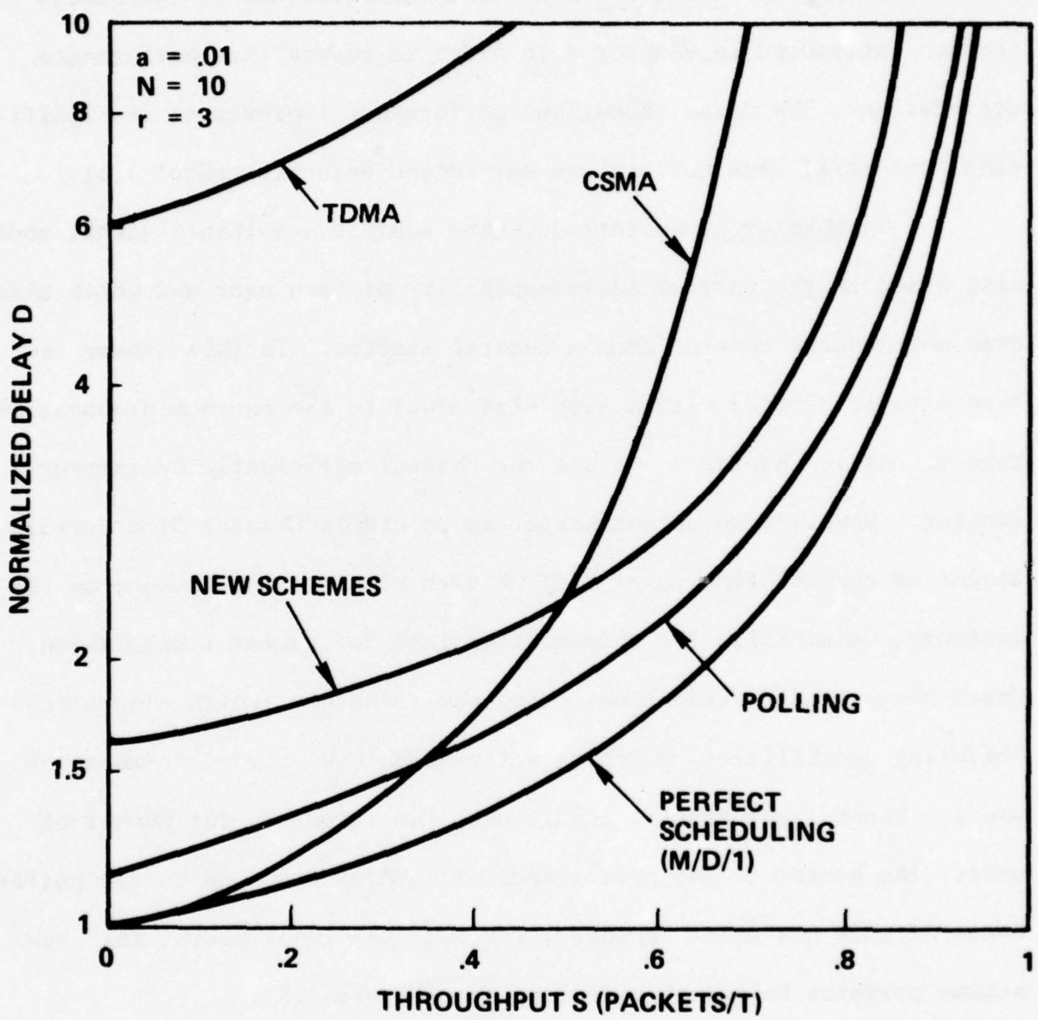


Figure 1.4. New Schemes: D vs S (N = 10).

However, when the number of users N increases, so does the overhead in each slot; therefore we observe a performance degradation with increasing N. Chapter 4 describes modifications to the access schemes introduced in Chapter 3 in order to reduce this performance degradation. The delay-throughput performance improvement is significant, but still does not achieve our second objective (Goal 1.b).

In Chapter 5, we introduce and analyze a multiple-access mode, also based on the carrier sense capability of each user and which also does not require control from a central station. In this scheme the time axis is slotted with a slot size equal to the maximum propagation time τ. As in Chapter 3, we use the channel efficiently by carrier-sensing. However, we do not waste (as we did in Chapter 3) a certain amount of channel time (overhead) at each packet transmission; on the contrary, we minimize the channel time lost for packet transmission. Under heavy traffic conditions, this new technique (which also allows buffering capabilities) performs better than CSMA for all numbers of users. Under light traffic conditions, the larger is the number of users, the better is the performance of CSMA as compared to the performance of this new mode. However, for all numbers of users, this new scheme performs better than Polling (see Figure 1.5).

Chapter 6 satisfies our third objective (Goal 2). We present, analyze and compare two techniques in which we include the traffic of a large user as well as the traffic of a population of small users over a single channel. The large buffered user "steals," by carrier sensing, slots which remain unused by the background of small users. Two models are studied which differ with respect to the nature of the

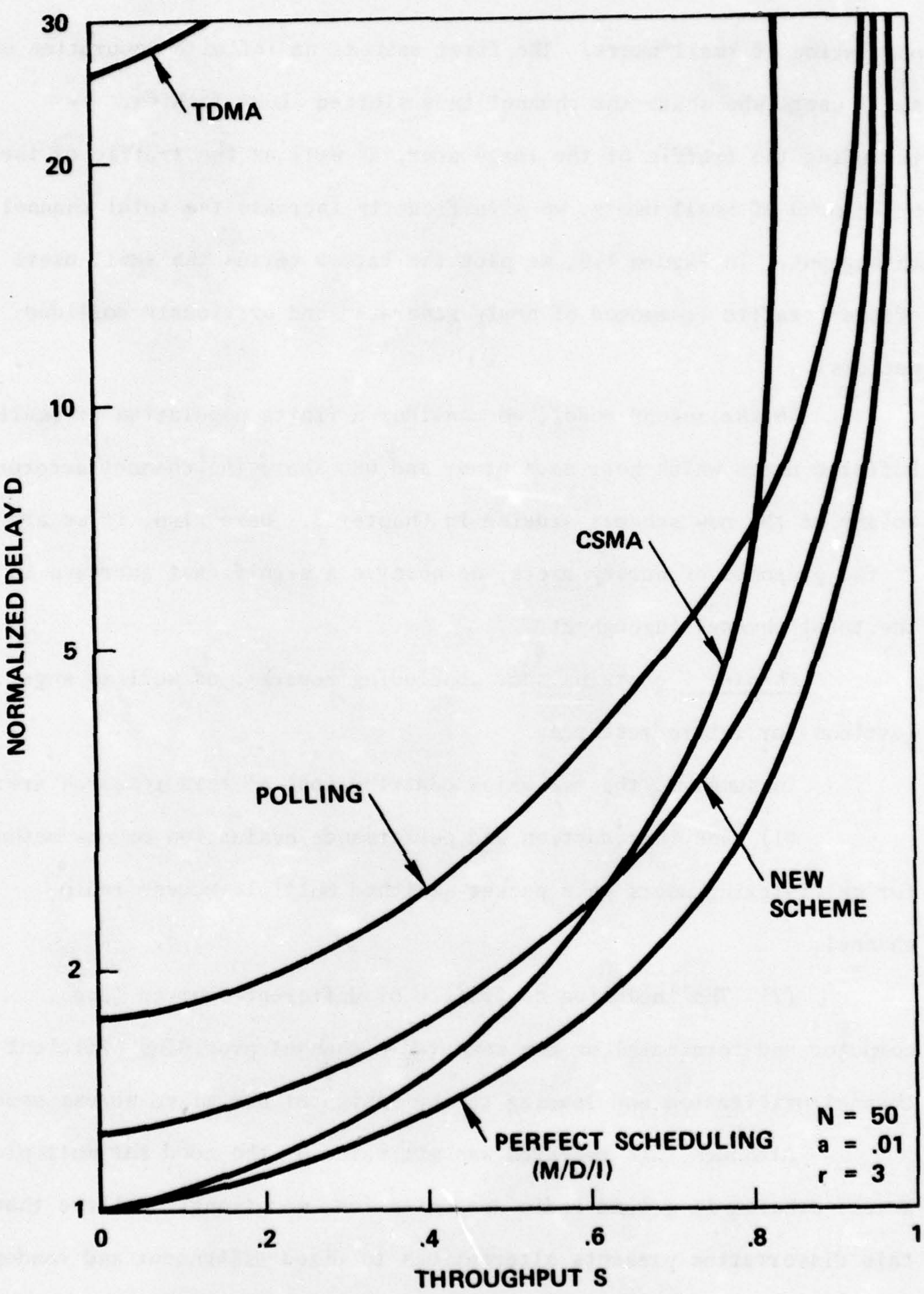


Figure 1.5. New Schemes: D vs S (N = 50).

population of small users. The first assumes an infinite population of small users who share the channel in a slotted ALOHA fashion. By including the traffic of the large user, as well as the traffic of the background of small users, we significantly increase the total channel throughput. In Figure 1.6, we plot the latter versus the small users offered traffic (composed of newly generated and previously collided packets).

In the second model, we consider a finite population of small buffered users which hear each other and who share the channel according to any of the new schemes studied in Chapter 3. Here also, if we are in the presence of bursty users, we observe a significant increase in the total channel throughput.

Chapter 7 contains some concluding remarks, as well as suggestions for future research.

In summary, the two major contributions of this research are:

(1) The introduction and performance evaluation of new methods for multiplexing users on a packet-switched multiple-access radio channel.

(2) The inclusion of traffic of different sources (e.g., computer and terminals) on the same radio channel providing efficient channel utilization and leading to the design of new mixed-access modes.

Although this research was motivated by the need for multiple-access schemes in ground radio communication, we strongly believe that this dissertation presents alternatives to fixed assignment and random access methods in general resource sharing systems.

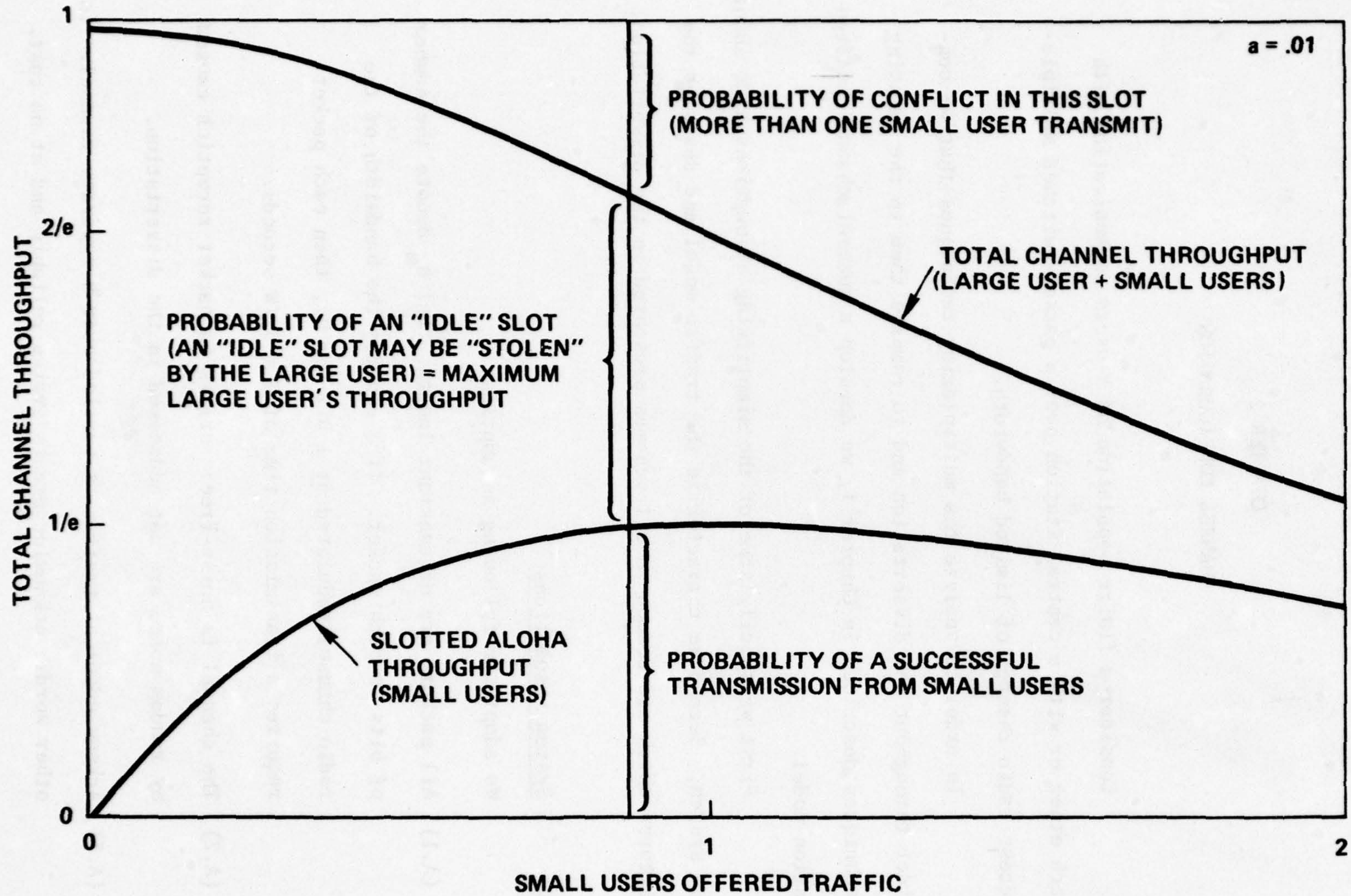


Figure 1.6. Aloha and Total Channel Throughputs versus Small Users Offered Traffic.

CHAPTER 2

CHANNEL CONFIGURATION

Consider a finite population of N users communicating with each other or with a central station over a packet-switched multiple-access radio channel of limited bandwidth.

In order to analyze the multiplexing techniques that we consider throughout the dissertation and to compare them to the existing techniques described in Chapter 1, we develop a general channel configuration model.

First we specify some of the simplifying assumptions made about the system. Second we characterize the traffic model and describe the features which are common to all schemes presented in the dissertation.

2.1 System Assumptions

We adopt the following assumptions:

(A.1) All packets are of constant length. Let b_m denote the number of bits in each packet. If W denotes the bandwidth of the radio channel modulated at 1 bit per Hz, then each packet requires a transmission time of $T = b_m/W$ seconds.

(A.2) The channel is noise-free: errors in packet reception caused by random noise are not addressed in the dissertation.

(A.3) Acknowledgement traffic is carried over a separate channel. In other words, acknowledgements arrive reliably and at no cost.

Let T_a denote the transmission time of the acknowledgement packet on the acknowledgement channel and τ_m the propagation delay from the transmitter to the receiver. We assume that no processing time is required at the receiver to generate an acknowledgement. Then the period for receiving the acknowledgement is simply equal to $T_a + 2\tau_m$ following the end of the transmission of the (information) packet at the transmitter.

(A.4) There is no multipath effect: The effect of multipath on any type of signal is to spread the signal duration by echoing it.

(A.5) There is no FM-capture effect: When two signals with different powers (say $P_1 > P_2$) are received simultaneously, some receivers can correctly receive the stronger signal. An FM receiver is characterized by its capture ratio:

$$CR = 10 \log \frac{P_1}{P_2} .$$

We assume a non-capture system, i.e., the overlap of any fraction of two packets results in destruction of both.

(A.6) The propagation delay between any source-destination pair is very small compared to the packet transmission time: If the maximum propagation delay is denoted by τ, we have:

$$\tau << T$$

Consider, for example, 1000 bit packets transmitted over a channel operating at a speed of 100 kilobits per second. The transmission time of a packet is then 10 msecs. If the maximum distance between the source and the destination is 10 miles, then the packet propagation delay is of the order of 54 μsecs. Thus the propagation delay is a very small fraction

$$a \triangleq \frac{\tau}{T} = .005$$

of the transmission time of a packet. On the contrary, when one considers satellite channels [KLEI 73] the propagation delay is a relatively large multiple of the packet transmission time (a >> 1).

In subsequent chapters additional assumptions on the system will be introduced as needed.

2.2 Traffic Model and Some Protocols' Features

The traffic model is defined as follows.

Traffic Source

Except in Chapter 6 (where the traffic source will be redefined) we consider a finite number N of buffered users, with unlimited buffer size. Each user generates traffic independently of the others according to a homogeneous Poisson process. The aggregate packet generation rate is denoted by λ (packets/second). If N is not too large, each user may generate packets frequently enough so that the interarrival time between successive packets at a given user is less than the delay incurred by a packet from arrival to the end of transmission. Thus, each user may have more than one packet requiring transmission at any time, which will be transmitted on a first-come-first-served basis. We further assume

(A.7) All Users are within range and in line of sight of each other. Therefore any user has the ability to sense the carrier of any other transmission on the channel.

Channel Throughput

Each packet has been assumed to be of constant length, requiring T seconds for transmission. Let $S = \lambda T$. S is the average number of packets generated per transmission time, i.e., the input rate normalized with respect to T. Under equilibrium, S can also be referred to as the channel throughput rate [LAM 74, TOBA 74].

Channel Utilization

The channel throughput rate S is less than or equal to one. If we were able to perfectly schedule the packets into the available channel space with no overlap or space between packets, we could achieve a maximum throughput of one. Therefore S is also referred to as the channel utilization [LAM 74, TOBA 74].

Slotted Transmission

The time axis is slotted. All users are synchronized and are forced to start transmission only at the beginning of a slot of duration denoted by s. When the users are not an equal distance apart, their synchronization is not simple. We assume:

(A.8) *This synchronization is feasible, and in this dissertation we will not address the problems this synchronization arises.*

Two slot sizes are considered, giving rise to different protocols, and will be defined in the subsequent chapters.

Control of Transmission

In the previous chapter we discussed the following:

(1) Classical techniques like TDMA and FDMA are quite

different in philosophy than random access modes such as CSMA or reservation schemes such as SRMA. While with the former we observed a low channel utilization and therefore a discrepancy in delay at low input rate due to the burstiness of traffic, the latter provided a good channel utilization at low input rate, at the price of collisions increasing with the input rate; this results in lower channel utilization at high input rate and in unstability.

(2) Another alternative to fixed assignment (TDMA, FDMA) and to random access (CSMA, SRMA) is provided by Polling, where a central station is controlling the transmission, therefore avoiding collisions and minimizing the time when the channel is wasted for the users when the number of users is not too large.

The purpose of the techniques studied in the subsequent chapters is to increase the channel utilization at low input rate without any control from a central station (rather we use self-control of the transmission by the users). In addition, collisions are avoided, and hence the channel is stable.*

In the following we choose:

1) a dynamic assignment of the channel to each user for a certain time duration. In other words, if user i is presently transmitting a packet over the channel, an assignment scheme designates a user j (possibly = i) to transmit the next packet. At each beginning of a packet transmission, all terminals know to which user the channel

*Stable and unstable channels are defined as follows (see [LAM 74] and [KLEI 75A]): In stable channels, the steady-state throughput-delay results are achievable over an infinite time horizon, while in unstable channels, such channel performance is achievable only for some finite time period before the channel goes to saturation, i.e., a zero throughput is observed because of increasing collisions.

is assigned.

2) as in CSMA, to use the ability of each user to detect the presence or absence of a carrier τ seconds after the beginning of the transmission. All other users know after τ seconds whether user i had a packet to transmit (carrier present) or not (carrier absent). In case the carrier is absent, they all know which user (user j) is chosen to start transmission immediately. They all listen to the carrier for the next τ seconds, after which, if the carrier is once more absent, a third user may start transmitting a packet, etc.

An access mode is characterized by the maximum number of attempts M to find a busy user (a user who has at least one packet to transmit).

Obviously, we have

$$1 \leq M \leq N$$

If after M attempts we find no busy users, various actions can be taken, giving rise to various protocols which will be defined in the subsequent chapters.

If M = 1: we have a fixed assignment, an example of which is TDMA. We will not consider this value of M.

If M = N: even though only one user has a packet to transmit, this user will eventually be chosen after M = N (worst case) attempts.

In subsequent chapters, the parameter M will be defined precisely.

Carrier Detection

After a maximum time of τ seconds, any user may start detecting the carrier of any transmission. We assume that

(A.9) The time required to detect the carrier due to packet transmissions is negligible with respect to τ.

<u>Channel Capacity</u>

Because we waste part of the available channel space (between the successive transmission of two packets, there may be an integer number of τ seconds wasted for sensing the carrier), the achievable throughput may be less than one. The maximum achievable throughput for an access mode is called the <u>capacity</u> of the channel under that mode, denoted C:

$$C = \max S$$

<u>Expected Packet Delay</u>

Together with the channel capacity, the expected packet delay D is an important performance measure. D is defined as the average time elapsing from the generation of a packet until the end of its transmission, normalized with respect to T (the delay in seconds is DT).

We have so far defined the following variables:

N: number of users

M: maximum number of attempts to find a busy user

b_m: number of bits per (information) packet

W: channel bandwidth

T: b_m/W, transmission time of a packet in seconds

τ: maximum propagation delay of any transmission

a: τ/T, maximum propagation delay normalized with respect to T ($a \ll 1$)

s: slot duration

λ: aggregate (total) mean packet generation rate (packets per second)

S: normalized channel throughput rate: $S = \lambda T$, channel utilization

D: expected packet delay normalized with respect to T

C: channel capacity

CHAPTER 3

HEAD OF THE LINE, ALTERNATING PRIORITIES, ROUND ROBIN AND RANDOM ORDER

Four protocols are described and analyzed in this chapter: Head of the Line (HOL), Alternating Priorities (AP), Round Robin (RR) and Random Order (RO). They differ by their dynamic assignment schemes; however, in each of them the slot size is the same and a slot is never wasted as long as at least one user has a packet to transmit, i.e.,

$$M = N$$

where M is the maximum number of attempts to find a busy user per slot.

In Section 1, we define the slot configuration. The various protocols are described in Section 2. The first important performance measure (the channel capacity) is shown in Section 3 to be the same under the four protocols and to be better than existing techniques only when the number of users, N, is small (≤ 20). In Section 4, we establish a conservation law which is of great use in analyzing and comparing the delay performance under those protocols. The main result is that the AP, RR and RO protocols all have the same average delay when all users have identical average input rates.

The next two sections are devoted to analytical results concerning delay under HOL and AP protocols (the latter for N = 2). Since we cannot compare AP, RR and RO in terms of average delay (they are the same), it is necessary to investigate higher moments. It is shown in

Section 8 that AP provides the smallest delay variance and RR the largest, although they are close to each other.

Finally, in Section 9 we compare the throughput delay performance of those three protocols to some of the existing techniques discussed in Chapter 1. For a small number of users, AP, RR and RO provide a delay throughput performance close to that obtained with Polling and are particularly suitable for the multiple access of a small number of buffered users. When the number of users increases, the performance degrades. However if all users are very close to each other, with a significant number of users (N up to 50) AP, RR and RO perform better than CSMA and CS SRMA under heavy traffic conditions. Like CSMA, the new schemes have the advantage of not requiring the control from a central station, while Polling and CS SRMA do.

3.1 Slot Configuration and Operational Features

Consider the slot configuration shown in Figure 3.1. The size of a slot is

$$s = (N - 1)\tau + T + \tau = T[Na + 1] \qquad (3.1)$$

where τ is the maximum propagation time, T is the transmission time of an (information) packet and N, the number of users. Thus a slot consists of three parts:

1) an overhead of (N - 1) "minislots," each of duration τ, followed by
2) the packet transmission time of length T, followed by
3) one minislot (propagation delay: the last bit of a packet is received at most one minislot after it was transmitted).

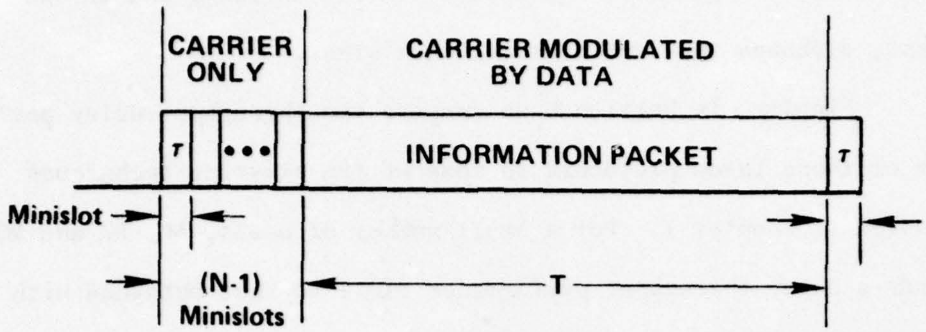


Figure 3.1. Slot Configuration.

All users are synchronized and may start transmission of the carrier (with no modulation) only at the beginning of a minislot, and all users may start transmission of the (data) packet (carrier modulated with data) only at the beginning of the second part of the slot, i.e., (N - 1) minislots after the beginning of a slot. In each slot an assignment scheme (see below) orders the N users.

If the highest order user, say u_1, has a packet to transmit, he starts sending the carrier (with no modulation) at the beginning of the slot, i.e., at the beginning of the first minislot. Within the first τ seconds, i.e., after at most one minislot, all other users will detect the presence of the carrier and remain quiet until next slot.

If u_1 has no packet to transmit then he remains quiet and after one minislot all other users know that the channel is idle (carrier absent). If the next user in order, say u_2, has data to send, he will start sending the carrier (without modulation) at the beginning of the second minislot.

If u_2 has no packet to transmit, then he remains quiet and the next user in order, say u_3, starts transmitting the carrier (with no modulation) at the beginning of the third minislot if he has data (u_3 transmits if the carrier has been absent for the first two minislots), etc.

We have already assumed (Assumption 9, Chapter 2) that the carrier detection time is negligible with respect to a minislot. We further assume that

(A.10) The time for choosing a user by the assignment scheme is negligible with respect to τ.

We assume that the assigned sequence $u_1, \ldots, u_N$ is known by all users at the beginning of the slot since the assignment scheme is run by all users before the end of the previous slot.

With Assumptions 8, 9 and 10, the protocols as described above and using such a slot configuration are considered as feasible; in this dissertation we do not address the practical problems of feasibility and implementation such as synchronization, implementation of sensing and of assignment algorithms. One way to satisfy Assumption 8 (synchronization problems due to the fact that users are not all equally distant) and Assumption 9 (carrier detection time negligible) is to increase the size of a minislot (previously chosen as equal to the maximal propagation time τ), or equivalently to take a larger value for the parameter a. We defined a in Chapter 2 as the ratio of τ over the packet transmission time T. We now define:

$$a \triangleq \frac{\text{minislot size}}{T} << 1 \tag{3.2}$$

3.2 The Protocols

The four protocols considered below differ in the way the assignment scheme orders the N users in each slot.

3.2.1 Head of the Line (HOL)

This protocol, named after the priority queueing system first studied by Cobham [COBH 54], is devised for a population of N users on which a fixed priority structure is imposed. In this system, queue i ($i = 1, 2, \ldots, N$) is served only if all queues of higher priority than queue i are empty, and before all queues with lower priority than queue i. In each queue, customers are served on a first-come-first-served basis. The priority among queues remains constant in time. *Without loss of generality*, we denote by u_1 the user with the highest priority, by u_2 the user with the next highest priority, ..., and by u_N the user with the lowest priority; and we write $u_1 > u_2 > \ldots > u_N$ in each slot. At each slot, u_i will transmit his data, if any, only if $u_1, u_2, \ldots, u_{i-1}$ are idle. If all users are idle, then the slot is unused.

Protocol:

(1) The highest order user (u_1) need never sense the channel and synchronizes his packet's transmission, if any, as follows:

(i) At the beginning of the slot he begins transmission of the carrier.

(ii) ($N - 1$) minislots later he transmits the packet.

(2) The i^{th} user in order (u_i)($1 < i \leq N$) senses the channel for ($i - 1$) minislots.

(i) if no carrier is detected after (i - 1) minislots, at the beginning of the i^{th} minislot, he sends the carrier and (N - i) minislots later, he transmits the packet.

(ii) Otherwise (u_i idle or carrier detected earlier) u_i waits for the next slot and then operates as before.

3.2.2 Alternating Priorities (AP)

This protocol, named after the priority queueing system studied by Miller [MILL 64], obeys the following rule:

(1) Assign the slot to that user (say user u_i) who transmitted the last packet, if possible. Otherwise (if there are no more packets from this user)

(2) Assign the slot to the next user in sequence (i.e., user $u_{i \bmod N+1}$)*.

(i) If this next user is busy a packet is transmitted in this slot, and in the following slot, operate as above.

(ii) If this next user is idle, then repeat step 2 until either a busy user is found or the N users have been scanned. In this latter case (all users idle), the slot is unused and in the following slot, operate as above. (The following slot is assigned to user u_i)

Protocol

(1) The user to which the slot is assigned does not sense the channel. If he has a packet to transmit, he transmits the carrier only, from the beginning of the slot and (N - 1)

* Users are ordered in a given sequence which rotates as new users gain access.

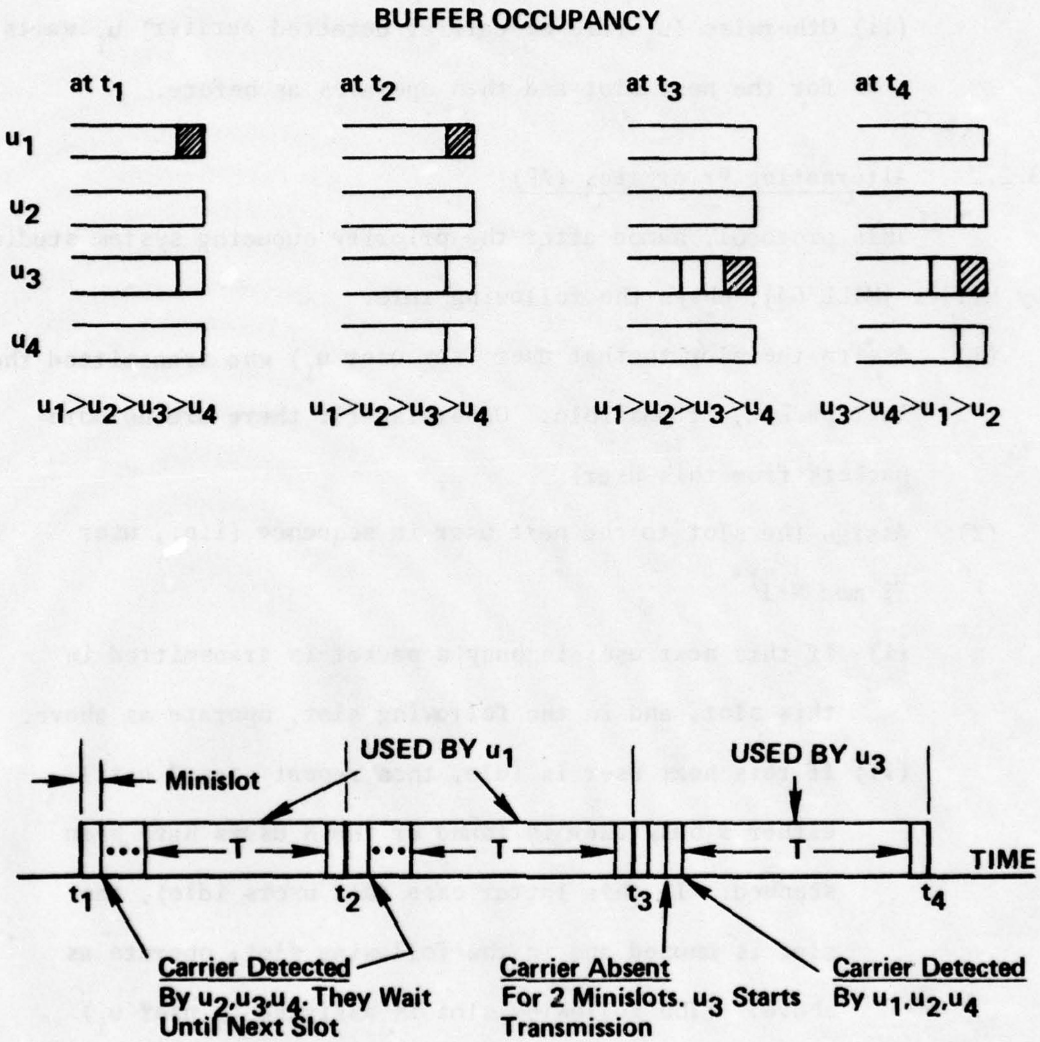


Figure 3.2 Alternating Priorities (N=4) (Cross-Hatching Indicates a Transmission).

minislots later he starts transmitting his packet. Otherwise (he has no packet to transmit) he remains quiet.

(2) The user who is currently i^{th} in sequence $(1 < i \leq N)$ senses the carrier for $(i - 1)$ minislots. If the carrier is absent and if he has data, he transmits the carrier at the beginning of the i^{th} minislot and $(N - i)$ minislots later, he transmits his data packet. Otherwise (carrier present or buffer empty) he remains quiet.

(3) Before the end of the slot, if the channel was switched from one user to another, all users update their priority, i.e., the number of minislots they will have to sense the carrier during the following slot before transmitting (if the carrier is absent). All users then operate as above.

In Figure 3.2 we consider an example with N = 4 queues. At time t_1, priority is given to user 1 who transmitted in the previous slot. User 1 has one packet in his buffer. He therefore transmits his packet in the slot starting at t_1. At the beginning of the next slot, at time t_2, his buffer contains a packet generated between t_1 and t_2. User 1 still has priority and transmits his packet. At time t_3, user 1's buffer is empty and priority is given to user 2 whose buffer is empty. Next in priority is user 3 who has 3 packets to transmit. User 3 transmits a packet in the slot starting at t_3 and will keep transmitting until his buffer is empty.

3.2.3 Round Robin (RR)

As in TDMA, each user is assigned one slot in a round robin fashion. After one user's slot has elapsed, the channel is switched to another user according to a fixed order, say u_1, u_2, ..., u_N, u_1, ... ; but if the user to whom the current slot is assigned is idle, the (N - 1) remaining users are scanned in sequence according to the same order* until a busy user is found. The protocol works as follows:

(1) If the user to whom the current slot is assigned has a packet to transmit, he transmits the carrier from the beginning of the slot and (N - 1) minislots later he transmits his data packet. If he is idle he remains quiet until the end of slot.

(2) The i^{th} user in sequence senses the carrier for i-1 minislots. If the carrier is absent and if he has a packet to transmit, he transmits the carrier from the beginning of the i^{th} minislot and (N - i) minislots later he transmits his data packet. Otherwise (carrier present or buffer empty) he remains quiet until the end of the slot.

(3) No matter who uses the current slot, slots are assigned to users in sequence: if the current slot is assigned (not necessarily used) to u_i, the next slot is assigned to $u_{i \bmod N+1}$.†

In Figure 3.3, we take an example with N = 3 queues. The slot starting at t_1 is assigned to user 1. Users 1 and 2 are idle, user 3 has 2 packets to transmit. User 3 transmits in this slot. The next

---

*If the slot is assigned to u_i ($1 \le i \le N$), the next user in sequence is $u_{i \bmod N + 1}$, ...; the last in sequence is u_{i-1} (or u_N if i = 1).

†<u>Variant</u>: If the current slot is assigned to u_i, and u_j uses the current slot (j = i, i+1, ..., N, 1, ..., i-1), then the next slot is assigned to $u_{j \bmod N+1}$. If the current slot is not used (all users idle) then the next slot is assigned to $u_{i \bmod N+1}$.

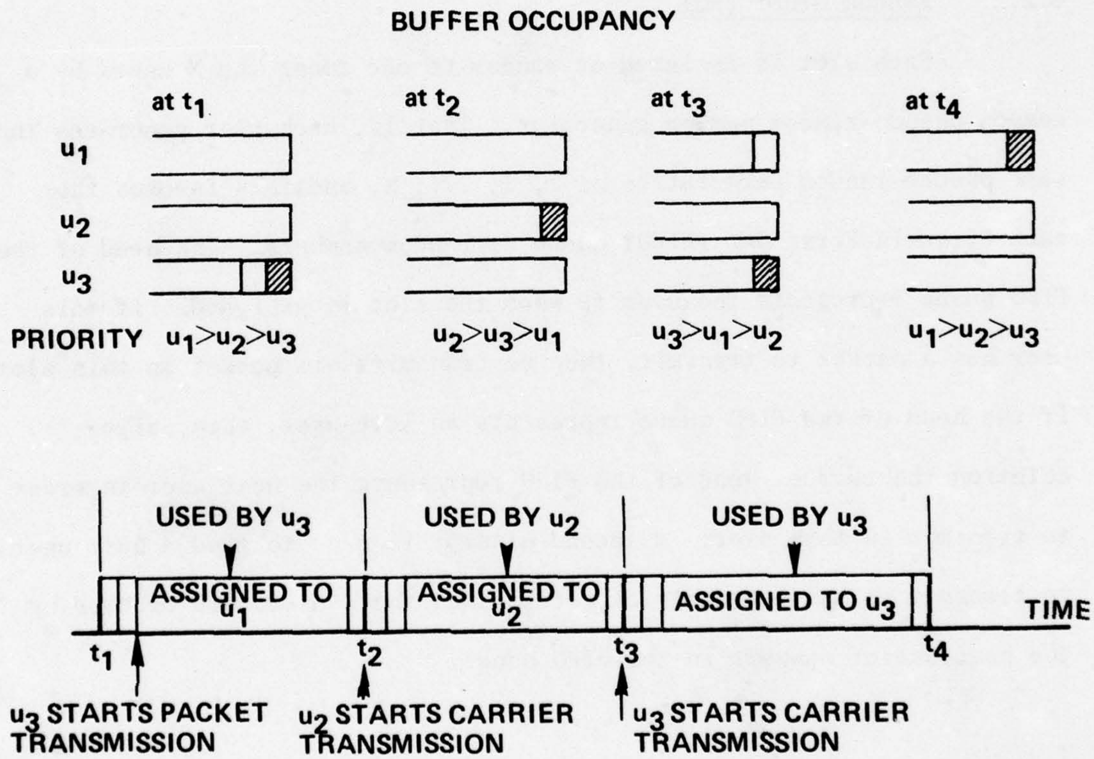


Figure 3.3. Round Robin (N=3) (Cross-Hatching Indicates a Transmission.

slot is assigned to user 2 who transmits the packet generated in the previous slot. The following slot is assigned to user 3 who still has a packet to transmit. The following slot assigned to user 1 will be used by user 1 who has meanwhile generated a packet.

3.2.4 Random Order (RO)

Each slot is assigned at random to one among the N users by a common pseudo-random number generator. That is, each user generates the same pseudo-random permutation of 1, 2, ..., N, and this invokes the same First-In-First Out (FIFO) queue of random numbers. The head of the FIFO queue represents the user to whom the slot is assigned. If this user has a packet to transmit, then he transmits his packet in this slot. If the head of the FIFO queue represents an idle user, then, after deletion the current head of the FIFO represents the next user in order to transmit in this slot: a second attempt is made to find a busy user to transmit in the current slot, etc. Thus the N users are ordered by the sequence of numbers in the FIFO queue.

Protocol:

(1) At the beginning of a slot, if the user to whom the slot is assigned has a packet to transmit, he sends the carrier at the beginning of the slot and (N - 1) minislots later, he transmits his packet. Otherwise (idle user) he remains quiet.

(2) The i^{th} user in sequence (represented by the i^{th} number in the FIFO queue from the head of the queue) senses the carrier for the (i-1) first minislots, after which he transmits (the

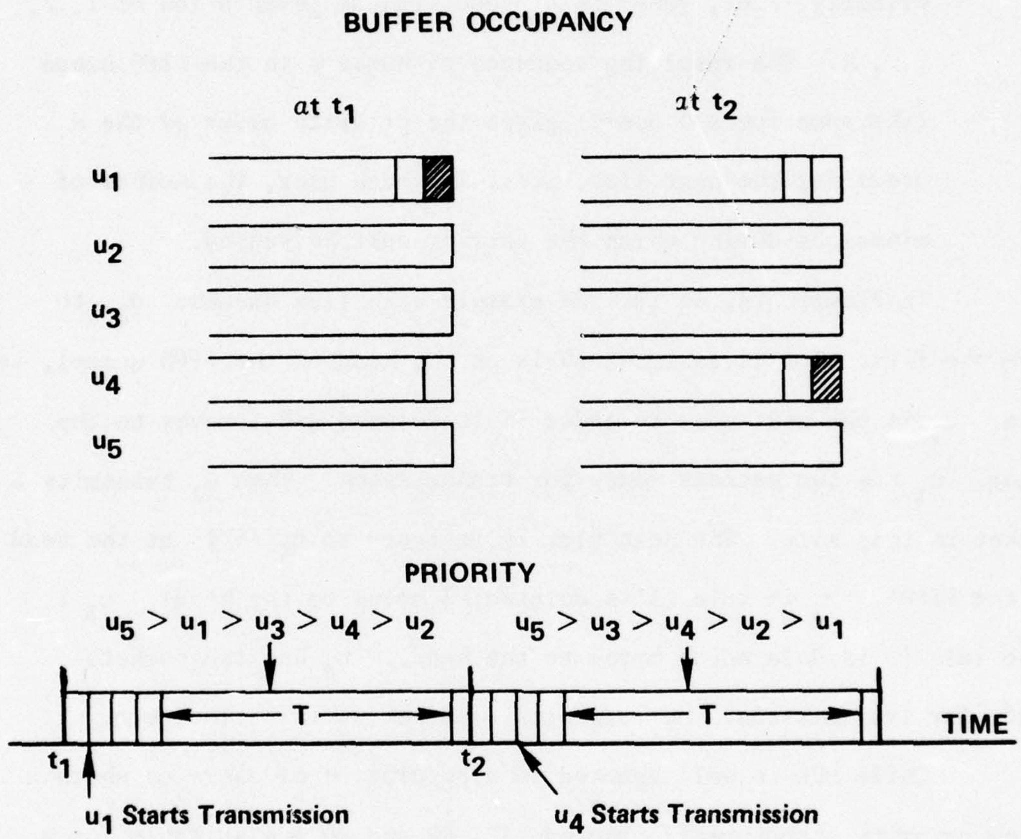


Figure 3.4. Random Order (N=5) (cross-hatching indicates a transmission)

carrier) if he has a packet to transmit and if the carrier were absent. Otherwise he remains quiet.

(3) Before the end of the current slot, all users update their priority, i.e., generate a pseudo-random permutation of 1, 2, ..., N. The resulting sequence of numbers in the FIFO queue (the same for all users) gives the priority order of the N users for the next slot, i.e., for each user, the number of minislots during which the carrier must be sensed.

In Fiqure 3.4, we take an example with five queues. u_5, to whom the first slot is assigned (5 is at the head of the FIFO queue), is idle. u_1 is the next user in order (5 is deleted and 1 moves to the head). u_1 has two packets ready for transmission. Then u_1 transmits a packet in this slot. The next slot is assigned to u_5 (5 is at the head of the FIFO). u_5 is idle (5 is deleted, 3 moves to the head). u_3 is also idle (3 is deleted, 4 moves to the head). u_4 has two packets ready for transmission. He transmits a packet in this slot, etc.

While HOL is well adapted to a population of users on which a fixed priority structure is imposed, AP, RR and RO are suitable for a population of users identical in terms of average input rate and priority.

The scheduling algorithm of the RO protocol is obviously more expensive to implement than are AP and RR.

Channel capacity and delay are the two important performance measures, the analysis of which will allow us to decide which protocol is best and to compare these techniques to existing techniques. First, we consider channel capacity.

3.3 Channel Capacity

The channel capacity C, under a given protocol, has been defined in Chapter 2 as the maximum achievable throughput for this protocol. Between the successive transmission of two packets, there are (N - 1) minislots wasted for carrier sensing, and therefore whatever protocol we use with the slot configuration defined in Section 3.1, the maximum achievable throughput is less than 1. Since within each slot (of size s defined by Eq. (3.1)), NaT seconds are wasted, where a is defined by Eq. (3.2); the channel capacity is

$$C = \frac{T}{s} = \frac{1}{1 + Na} \tag{3.3}$$

In Figure 3.5, we plot the capacity C versus the number of users N for various values of a.

On the same figure are plotted the capacity of Polling, CSMA and slotted ALOHA. With Polling [KOHN 72] one can always achieve a theoretical throughput of 1, since when one user transmits over the channel, he keeps transmitting at a rate of one packet per packet transmission time. If his buffer never empties there is no waste of the channel due to switching to (polling) another user. The capacity of slotted ALOHA is known to be 1/e [KLEI 75A]. The slotted non-persistent CSMA protocol provides the highest capacity among all CSMA protocols [KLEI 75B]; the CSMA capacity is plotted for a = .01, .05, and .001.

The capacity of HOL, AP, RR and RO is observed to decay very fast below the CSMA capacity when a is not too small (N = 18 for a = .01, N = 10 for a = .05) and is worse than the slotted ALOHA mode for N > 172 if a = .01 (or N > 34 if a = .05). However, when a is very

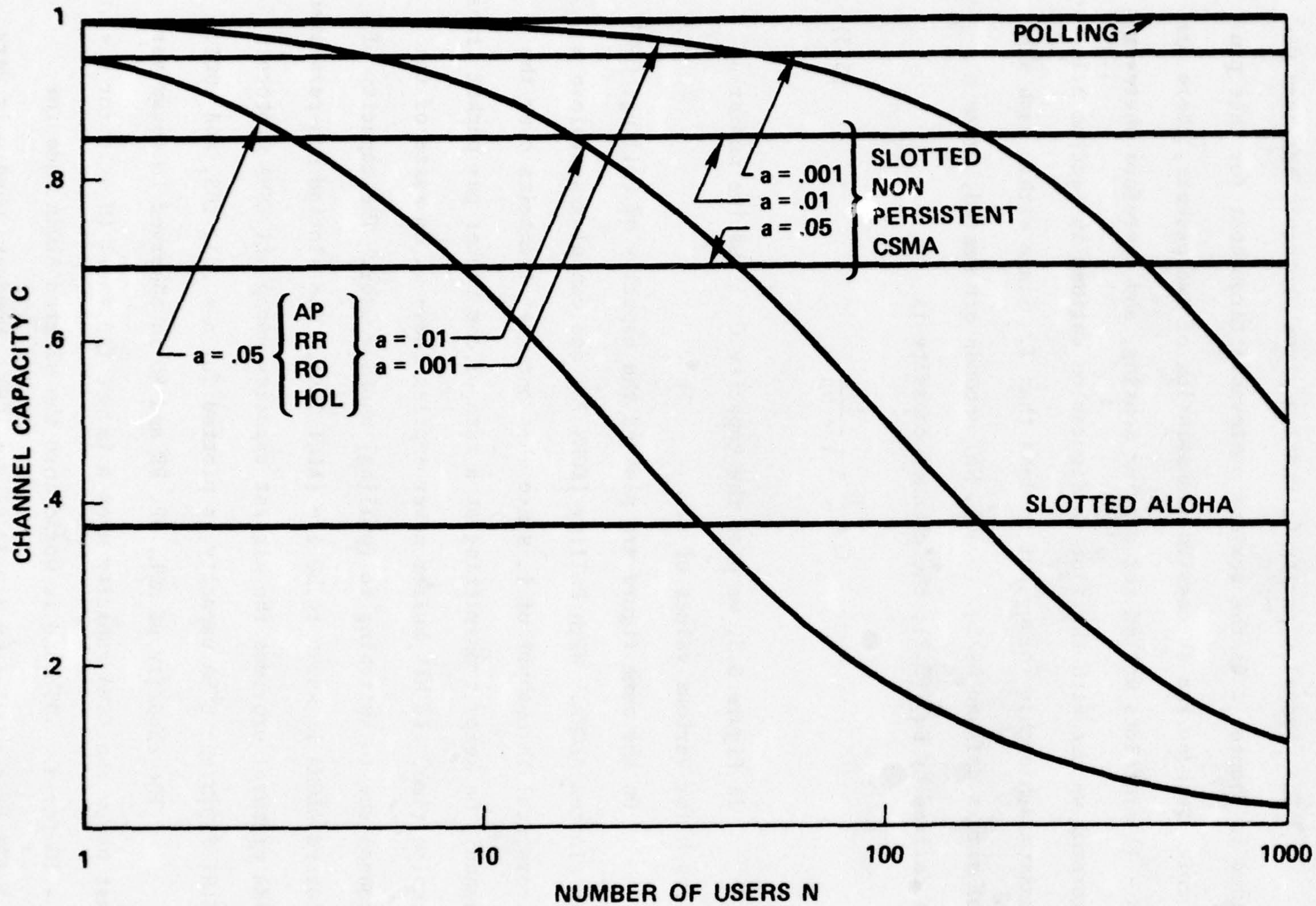


Figure 3.5. Effect of Number of Users on Channel Capacity.

small, say a = .001*, the capacity stays high (> 90%) for N $\leq$ 110.

In Figure 3.6 we plot the channel capacity C versus a for various values of N, and compare it to the channel capacity of CSMA (slotted non-persistent) [KLEI 75B] and slotted ALOHA [KLEI 75A]. For a small number of users (N = 10), the protocols studied in this chapter give a higher channel capacity than all CSMA protocols for values of a not larger than .038, and is fairly good when a $\leq$ .02 ($\geq$ 83%). When the number of users is larger, C quickly decays as a increases; if N = 50 for values of a > .35, the capacity drops below that of slotted ALOHA.

After channel capacity, the second important performance measure we analyze is the delay incurred by a packet from its generation to the end of its transmission. We first establish a conservation law which gives us, as a main result, the average delay under a broad class of protocols, whatever the number of users is, when all users have the same input rate.

3.4 A Conservation Law

Multiple access to a radio channel from a finite number of buffered users can be modeled as a priority queueing system with one server (service corresponds to the transmission of a packet over one single channel). Customers (packets) arrive at multiple queues to be served according to a queueing discipline which is nothing more than a protocol (i.e., a means for choosing which customer in which queue is to be served next). Since we assumed Poisson generation of the packets

* a = .001; if for example, all users are less than 2 miles apart and transmit 1000 bit packets over a channel operating at a speed of 100 kilobits per second, or if all users are less than 10 miles apart and transmit 5000 bit packets over a channel operating at a speed of 100 kilobits per second.

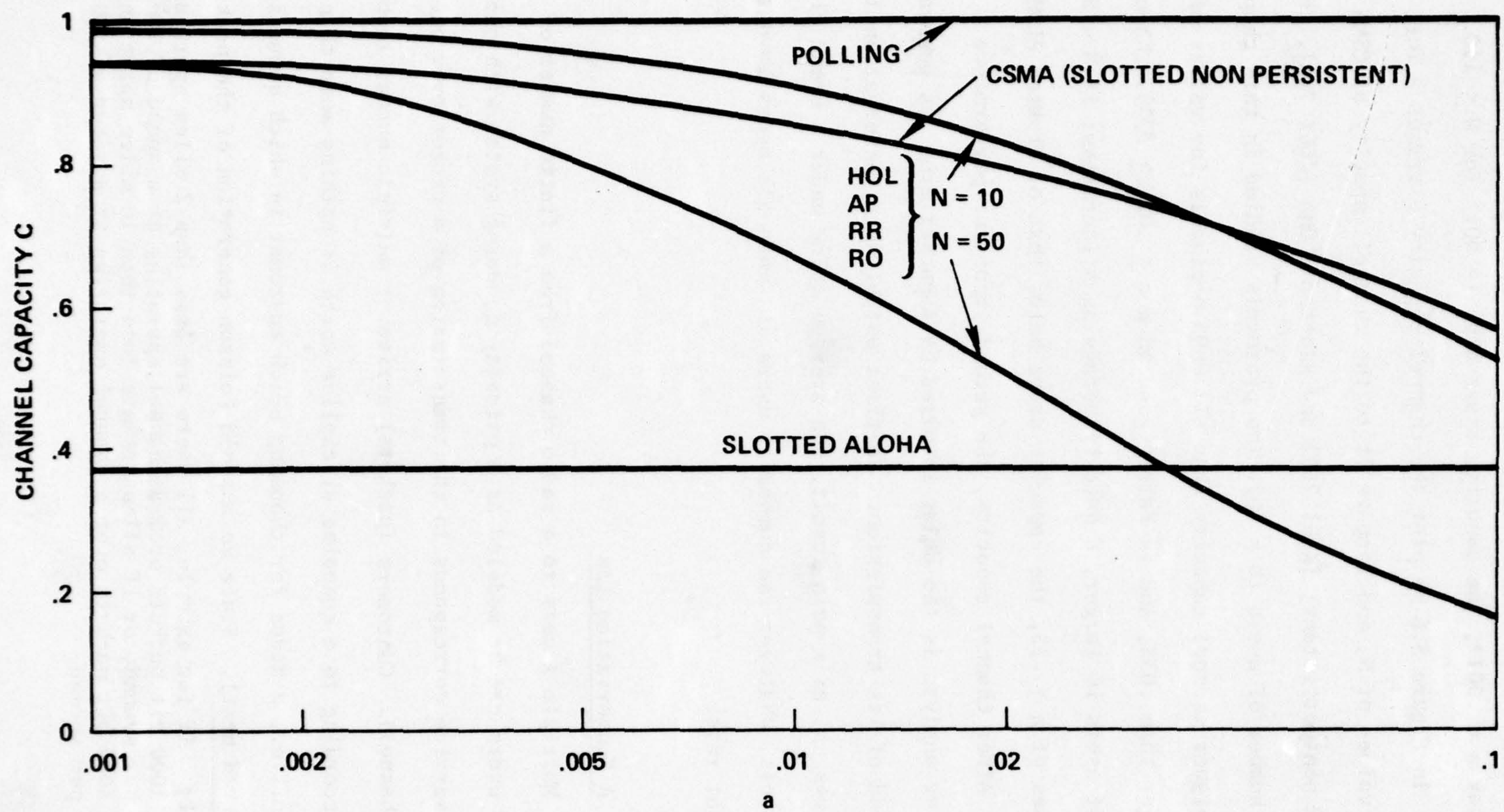


Figure 3.6. Effect of Propagation Delay on Channel Capacity.

we could apply all the results known about M/G/1 priority queueing systems (see for example [KLEI 76], or for a more complete work on the subject see [JAIS 68]). However, in any M/G/1 queueing system, a customer who, upon arrival, finds the system idle (no customer in any queue or in service) is immediately serviced. In any conservative* priority queueing system [KLEI 65], a customer who, upon arrival finds the system idle (no customer in any of the queues or in service) is immediately serviced. This is not true in our multiple access schemes since the transmission is slotted. Whatever protocol we use, a packet which upon generation finds the system empty (no packet waiting for transmission at any user; no packet being transmitted), has to wait until the beginning of the following slot to be a candidate for transmission. Therefore we model our access schemes as work-conserving priority queueing systems with <u>rest period</u>. When the server goes idle (upon service completion of the last customer) he goes for a "vacation" with an arbitrary distribution function. At the end of this vacation, he starts serving any customers who arrived during this vacation, or if no one arrived, he goes for another vacation.

Our slotted system can be modeled as a queueing system with rest period where both service time and rest period are deterministic with the same length (slot size). The queue M/G/1 with rest period has been studied by Miller [MILL 64]. Using another approach, in Appendix A we give some useful results concerning delay in such a queueing system.

For any M/G/1 system and a non-preemptive work-conserving

---

*By "conservative" or "work-conserving," we mean [KLEI 65] that no work is created (example of creation: server standing idle in the face of a non-empty queue) or destroyed (example of destruction: customer leaving the system before service completion).

queueing discipline, Kleinrock first stated and proved the following conservation law [KLEI 65]:

$$\sum_{p=1}^{P} \rho_p \overline{W}_p = \frac{\rho \overline{W}_0}{1 - \rho} \qquad \rho < 1 \tag{3.4}$$

where arriving customers belong to one of a set of P different priority classes, customers from priority group p arrive in a Poisson stream at rate λ_p customers per second, each customer from this group has a mean service time $\overline{x}_p$ and a service time second moment $\overline{x_p^2}$. Customers from group p incur an average waiting time $\overline{W}_p$, and ρ_p, ρ and $\overline{W}_0$ are defined as follows:

$$\rho_p = \lambda_p \overline{x}_p \tag{3.5}$$

$$\rho = \sum_{p=1}^{P} \rho_p \tag{3.6}$$

$$\overline{W}_0 = \sum_{p=1}^{P} \rho_p \frac{\overline{x_p^2}}{2\overline{x}_p} = \sum_{p=1}^{P} \lambda p \frac{\overline{x_p^2}}{2} \tag{3.7}$$

$\overline{W}_0$ represents the expected residual life of the customer found in service upon an arrival's entry.

Thus this weighted sum of the average waiting times $\overline{W}_p$ never changes, whatever the queueing discipline is.

The purpose of this section is to extend this conservation law to the M/G/1 queue with rest period; by the same argument, this law can be extended to the M/G/1 queue with initial set-up time (see Appendix D).

3.4.1 M/G/1 with Rest Period Conservation Law

The following law holds for queueing systems with rest period under the following restrictions:

1. Arrival statistics are all Poisson; service statistics are arbitrary; and arrival and service statistics are all independent of each other.

2. Preemption* is not allowed and the queueing discipline is work-conserving.

3. If there are no more customers in the system so that the server goes idle for lack of work, he will be withdrawn from the system for some time T_0 (rest period) with an arbitrary distribution function. At the end of the rest period, the server will return and begin to serve the customers that have accumulated during his absence. If there is no backlog, he will take another independent rest period which begins immediately.

It is assumed throughout that the systems under consideration are in the steady-state equilibrium. In general, this is equivalent to requiring that the system has been operating for a long time and that $\rho < 1$, where ρ (Eq. (3.6)) is, as usual, the product of the average arrival rate of customers per second times their expected service time:

$$\lambda = \sum_{p=1}^{P} \lambda_p \tag{3.8}$$

$$\overline{x} = \sum_{p=1}^{P} \frac{\lambda_p}{\lambda} \overline{x}_p \tag{3.9}$$

$$\rho = \sum_{p=1}^{P} \rho_p = \lambda \overline{x} \tag{3.10}$$

*"If a customer in the process of being served is liable to be ejected from service and returned to the queue whenever a customer with a higher value of priority appears, the system is said to be preemptive." [KLEI 65].

Theorem

For any queueing discipline and any given arrival and service time parameters subject to restrictions 1 through 3 above,

$$\sum_{p=1}^{P} \rho_p \overline{W}_p = \begin{cases} \text{constant with respect to variation} \\ \text{of the queueing discipline} \end{cases} \qquad (3.11)$$

where $\overline{W}_p$ is the expected waiting time of the p^{th} priority group, and where P represents the total number of groups to be distinguished. In particular,

$$\sum_{p=1}^{P} \rho_p \overline{W}_p = \frac{\rho}{1-\rho}\overline{W}_0 + \rho\tau_0 \qquad \rho < 1 \qquad (3.12)$$

where $\overline{W}_0$ (Eq. (3.7)) represents the expected residual life of the customer found in service upon an arrival entry, and τ_0 represents the expected residual life of the rest period upon an arrival of a customer who arrives during a rest period:

$$\tau_0 = \frac{\overline{T_0^2}}{2\overline{T}_0} \qquad (3.13)$$

Proof:

The proof follows the argument by Kleinrock almost exactly. Let us define U(t) as the total unfinished work present in the system at time t. "In particular, U(t) represents the time that it would take to empty the system of all customers present at time t, if no new customers were allowed to enter the system after time t." [KLEI 65]. A typical section of U(t) might look like the graph shown in Figure 3.7.

The instants t_i are the times of arrival of new customers to the system, each customer requiring a service (work) of x_i seconds. At t_i, U(t) increases by an amount x_i. During the rest period, U(t) cannot decrease (i.e., it may only jump up at t_i and remain constant between successive t_i's). When the server ends its rest period, U(t) decreases at a steady rate of 1 sec/sec as long as U(t) is positive; as *before, it jumps by* x_i *at the times* t_i, and once having reached zero, it remains there until the next customer's arrival.

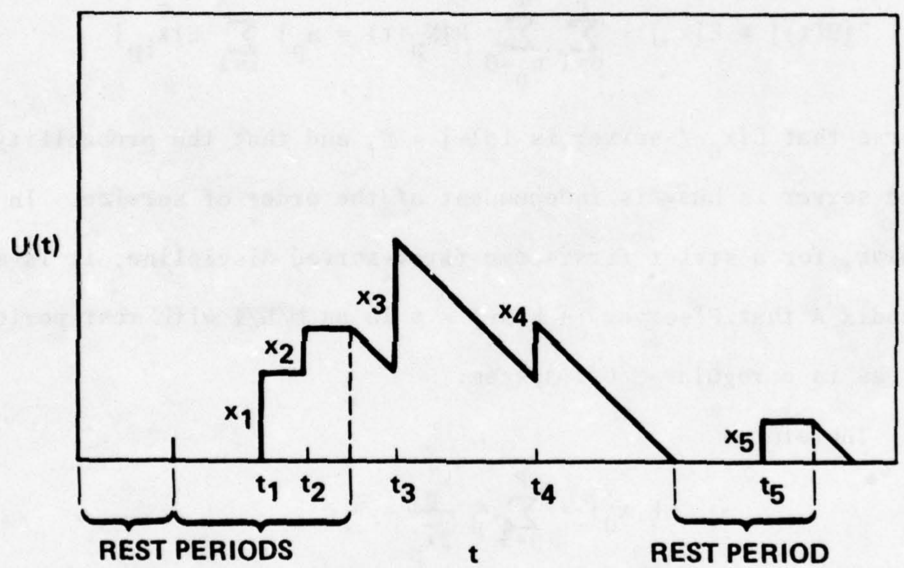


Figure 3.7. Unfinished Work, U(t).

It is clear that regardless of the order of service (no matter what conservative queueing discipline is used), the function U(t) will not change. It is also clear that "no matter which U(t) function turns up, as long as the same statistics are used for the t_i and x_i, the expected value $\overline{U}$ of the unfinished work will be the same" [KLEI 65].

If at time t there are $N_p(t)$ customers from group p in the queue, and if the i^{th} of these (i = 1, 2, ..., $N_p(t)$) is to have a service time x_{ip} and if x_0 represents the work yet to be done on the man in service (if there is any), then we may say

$$U(t) = x_o + \sum_{p=1}^{P} \sum_{i=1}^{N_p(t)} x_{ip}$$

regardless of the order of service.

Then taking expectations on both sides, we have

$$E[U(t)] = E[x_0] + \sum_{p=1}^{P} \sum_{n_p=0}^{\infty} P[N_p(t) = n_p] \sum_{i=1}^{n_p} E[x_{ip}] \quad .$$

We observe that $E[x_0$ / server is idle] = 0, and that the probability that the server is busy is independent of the order of service. In particular, for a strict first-come-first-served discipline, it is shown in Appendix A that P[server is busy] = ρ in an M/G/1 with rest period system, as in a regular M/G/1 system.

Therefore

$$E[x_0] = \sum_{p=1}^{P} \rho_p \frac{\overline{x_p^2}}{2\overline{x}_p} = \overline{W}_0$$

since $E[x_{ip}^n] = \overline{x_p^n}$ is independent of the index i. With the time t taken at random (and large from beginning), we may write

$$\overline{U} \triangleq \lim_{t\to\infty} E[U(t)]$$

$\overline{U}$ is the limiting average unfinished work. Thus we have:

$$\overline{U} = \overline{W}_0 + \lim_{t\to\infty} \sum_{p=1}^{P} \sum_{n_p=0}^{\infty} n_p P[N_p(t) = n_p]\overline{x}_p$$

$$\overline{U} = \overline{W}_0 + \sum_{p=1}^{P} \overline{x}_p E[N_p]$$

However, $E[N_p] = \lambda_p \overline{W}_p$ by Little's result [LITT 61]. Thus, we conclude

$$\overline{U} = \overline{W}_0 + \sum_{p=1}^{P} \rho_p \overline{W}_p \qquad (3.14)$$

We get the same expression as the one that was obtained for the average unfinished work in a regular M/G/1 priority queueing system (see Eq. (13) in [KLEI 65]). This is not surprising since in both systems the increase of work is the same and, on the average, the fraction of time when the server is busy is ρ (and we know that in both cases when the server is busy, the unfinished work decreases at a rate of 1 sec/sec.).

Now since $\overline{U}$ is independent of the order of service, we may as well calculate $\overline{U}$ for a strict *first-come-first-served* discipline.

Looking at the graph shown in Figure 3.7, one recognizes that for an arrival occuring at time t:

i) if the arrival occurs during a period when the server is busy (with probability ρ), then

$$W_{FCFS}(t) = U(t)$$

where $W_{FCFS}(t)$ is the waiting time of a customer arriving at time t in a first-come-first-served discipline. Indeed, if service is given in order of arrival, the time a customer has to wait for service if he arrived at time t is precisely the backlog of work at time t, i.e., U(t).

ii) if the arrival occurs during a rest period (with probability $(1 - \rho)$), then

$$W_{FCFS}(t) = U(t) + \theta_0$$

where θ_0 is the residual life of the rest period upon arrival.

Taking expectations we may write:

$$\overline{W}_{FCFS} = \overline{U} + (1 - \rho)\tau_0 \tag{3.15}$$

where τ_0 is the expected residual life of the rest period: $\tau_0 = \dfrac{\overline{T_0^2}}{2\overline{T}_0}$.

If we use the value of $\overline{U}$ drawn from Eq. (3.15) in Eq. (3.14) we have

$$\overline{W}_{FCFS} - (1 - \rho)\tau_0 = \overline{W}_0 + \sum \rho_p \overline{W}_p \tag{3.16}$$

Eq. (3.16) is true regardless of the order of service. In particular, if the queueing discipline is FCFS, $\overline{W}_p = \overline{W}_{FCFS}$ for all p, and since $\sum\rho_p = \rho$, we finally obtain from Eq. (3.16)

$$\overline{W}_{FCFS} = \frac{\overline{W}_0}{1 - \rho} + \tau_0 \tag{3.17}$$

Eq. (3.17) is consistent with the expression for the average waiting time in an M/G/1 with rest period (Appendix A: Eq. (A.27)).

$$\overline{W}_{FCFS} = \frac{\lambda\overline{x^2}}{2(1 - \lambda\overline{x})} + \tau_0 \tag{3.18}$$

Indeed, Eq. (3.17) becomes

$$\overline{W}_{FCFS} = \frac{\sum_{p=1}^{P} \lambda_p \overline{x_p^2}}{2(1 - \rho)} + \tau_0$$

and observing that $\overline{x^2} = \sum_{p=1}^{P} \frac{\lambda_p}{\lambda} \overline{x_p^2}$ and that $\rho = \lambda\overline{x}$, we finally get Eq. (3.18).

Substituting the value of $\overline{W}_{FCFS}$, as given by Eq. (3.17), into Eq. (3.16) we have the conservation law given in Eq. (3.12).

Now, in the special case where $\overline{x}_p = \overline{x}$ for all p, then the conservation law gives

$$\sum_{p=1}^{P} \lambda_p \overline{W}_p = \frac{\lambda \overline{W}_0}{1 - \rho} + \lambda \tau_0 \qquad \rho < 1 \tag{3.19}$$

However, from Little's result we have

$$\sum_{p=1}^{P} E[N_p] = \frac{\lambda \overline{W}_0}{1 - \rho} + \lambda \tau_0$$

But this sum is merely the average total number in queue, denoted by $\overline{N}_q$

$$\overline{N}_q = \frac{\lambda \overline{W}_0}{1 - \rho} + \lambda \tau_0 \qquad \rho < 1 \tag{3.20}$$

And applying once more Little's result, we have

$$\overline{W} = \frac{\overline{W}_0}{1 - \rho} + \tau_0 \qquad \rho < 1 \tag{3.21}$$

Thus in the special case where $\overline{x}_p = \overline{x}$, then the average total number in queue and the average waiting time in queue are independent of the queue discipline:

$$\sum_{p=1}^{P} \lambda_p \overline{W}_p = \lambda \overline{W} \qquad \rho < 1 \tag{3.22}$$

Furthermore, one can easily show that when the order of service is independent of service time, then the distribution of total number of customers in system, and thus the average waiting time, are both independent of the queueing discipline (note that $\overline{x}_p = \overline{x}$ is a less strong assumption than "order of service independent of service time"). The approach for showing this statement is exactly the same used in a regular M/G/1 priority queueing system [KLEI 76].

When the order of service is independent of service time, in particular we have

$$\overline{x_p^2} = \overline{x^2} \quad \text{for all } p \qquad \text{and} \qquad \overline{W}_0 = \frac{\lambda \overline{x^2}}{2}$$

Thus, Eq. (3.22) becomes

$$\sum \frac{\lambda_p}{\lambda} \overline{W}_p = \frac{\lambda \overline{x^2}}{2(1 - \rho)} + \tau_0 \qquad \rho < 1 \tag{3.23}$$

$$\text{with} \quad \begin{cases} \tau_0 = \dfrac{\overline{T_0^2}}{2\overline{T}_0} \\ \rho = \lambda \overline{x} \end{cases}$$

The right-hand side of Eq. (3.23) is merely the average waiting in an M/G/1 queue with rest period (Eq. (3.18)).

Thus the conservation law puts a linear equality constraint on the set of average waiting times $\overline{W}_p$; the weighted sum of the average waiting time $\overline{W}_p$ is equal to the waiting time under FCFS discipline. Any attempt to modify the queueing discipline so as to reduce one of the $\overline{W}_p$'s will force an increase in some of the other $\overline{W}_p$'s in a way which balances the result.

3.4.2 Average Packet Delay in AP, RR, RO

We shall model our multiple access schemes as priority queueing systems with rest period where the service time and rest period have the same deterministic distribution of length s = [1 + Na]T, and where, to each user, corresponds a priority group, with a total of N priority groups.

Let us denote by Δ_i the average delay normalized with respect to s (i.e., expressed in slots) incurred by a packet generated at user i, and W_i denote the average waiting time in queue (user's buffer) of the packet generated at user i, normalized with respect to s. Then

$$\Delta_i = W_i + 1$$

Any packet requires one slot of service (all packets have the same size and thus require the same amount of channel time) whatever user it is generated at.

We may then apply the conservation law, by means of Eq. (3.23) which becomes

$$\sum_{i=1}^{N} \frac{\lambda_i}{\lambda} W_i = \frac{\rho}{2(1-\rho)} + \frac{1}{2} = \frac{1}{2(1-\rho)} \qquad \rho < 1 \qquad (3.24)$$

where $\rho = \lambda\overline{x} = \lambda s$ can be viewed as the total input rate normalized with respect to s, and where λ_i is the generation rate (packets/second) at user i.

Also, we may write

$$\sum_{i=1}^{N} \frac{\lambda_i}{\lambda} \Delta_i = \frac{1}{2(1-\rho)} + 1 \qquad \rho < 1 \qquad (3.25)$$

Eq. (3.24) and Eq. (3.25) are true whatever protocol is used!

Furthermore, the total average number of packets is independent of the protocol used and is given by

$$\overline{N} = \frac{\rho}{2(1-\rho)} + \rho \qquad \rho < 1 \qquad (3.26)$$

Let us define a <u>symmetric protocol</u> as one under which, when the input rate is the same at all users, the average delay of a packet is independent of the user at which it was generated:

if $\lambda_i = \lambda/N$ for all i

then $\Delta_i = \Delta_j$ for all i and j between 1 and N.

Therefore, one corollary of the conservation law is: when the packet generation rate is the same at all users, the average packet delay for each group is independent of the (symmetric) protocol, and given by

$$\Delta_i = \frac{1}{2(1 - \rho)} + 1 \qquad i = 1, N$$

$$\rho = \lambda T(1 + Na) < 1 \tag{3.27}$$

In particular, AP, RR and RO are three symmetric schemes which give the same packet delay given by Eq. (3.27). In order to compare AP, RR and RO and decide which one is the best in terms of delay, we will have to compare the delay variance under these various protocols.

First however, let us try to solve for the average delay in the general case, i.e., when the input rate is not the same at all queues. For HOL a mean value analysis is available which we will present in the next section. For AP, RR and RO, the problem is not easy; we will present the special case of N = 2 users and give an exact analysis of AP. We have found a light traffic approximation for the analysis of RR in the special case of (N = 2) users (unpublished note). No analysis is available for RO.

3.5 Head of the Line (HOL)

We will show that the average delay of a packet generated at queue p and expressed in slots is given by

$$\Delta_p = 1 + \frac{1}{2(1 - \sigma_p)(1 - \sigma_{p+1})} \tag{3.28}$$

where

$$\sigma_p = \sum_{i=p}^{N} \rho_p \tag{3.29}$$

$$\rho_p = \lambda_p s$$

We choose, without loss of generality, an external priority structure such that queue p, p = 2, ..., N, has higher priority than queue (p - 1).

For such a discipline in a regular (no rest period) M/G/1, the average waiting time at queue p is (see [KLEI 76], Chapter 3)

$$\overline{W}_p = \frac{\overline{W}_0}{(1 - \sigma_p)(1 - \sigma_{p+1})} \tag{3.30}$$

where $\overline{W}_0$ is given by Eq. (3.7)

Using the same mean value analysis as for the regular HOL M/G/1, and the same notations as used in [KLEI 76], we show that for an M/G/1 system with rest period and a HOL queueing discipline, the average waiting time at queue p is given by

$$\overline{W}_p = \frac{\overline{W}_0 + \tau_0(1 - \rho)}{(1 - \sigma_p)(1 - \sigma_{p+1})} \tag{3.31}$$

$$p = 1, 2, \ldots, N$$

where τ_0 is given by Eq. (3.13)

From Eq. (3.31) we then easily get Eq. (3.28) which also has been directly derived by a different approach [SPRA 72].

Consider a customer newly generated at queue p. The first part of the delay of this customer referred to as the "tagged customer" is due to the customer he finds in service, or if there is nobody in service, the first part of the delay is the residual life of the rest period. The second component of delay is due to customers found in the queues by our tagged customer and who receive service before he does. Finally the third part of the delay is due to later arrivals than he.

Consequently, the total average delay in queue for our tagged customer may be written as

$$\overline{W}_p = \overline{W}_0 + \tau_0(1 - \rho) + \sum_{i=1}^{N} \overline{x}_i\overline{N}_{ip} + \sum_{i=1}^{N} \overline{x}_i\overline{M}_{ip} \qquad (3.32)$$

The first two terms of the right-hand side correspond to the first component of the delay; the third and fourth terms correspond respectively to the second and third components of delay, where $\overline{N}_{ip}$ and $\overline{M}_{ip}$ are defined as follows.

$\overline{N}_{ip}$ represents the average number of customers of queue i found by our tagged customer upon arrival and who receive service before he does. Therefore, we have $\overline{N}_{ip} = 0$ for $i = 1, \ldots, p - 1$.

$\overline{M}_{ip}$ represents the average number of customers of queue i who arrive at the system while our tagged customer is in the queue and who receive service before he does. Therefore, we have $\overline{M}_{ip} = 0$ for $i = 1, \ldots, p$.

From Little's result, we also have

$$\overline{N}_{ip} = \lambda_i\overline{W}_i \qquad i = p, p + 1, \ldots, N \qquad (3.33)$$

Indeed, on the average, there will be $\lambda_i\overline{W}_i$ customers present at queue i when our tagged customer arrives.

Similarly, since he spends on the average $\overline{W}_p$ seconds in queue, and since each queue's arrival process is independent of queue size, there will be on the average $\lambda_i\overline{W}_p$ customer arrivals at queue i while our tagged customer waits on queue. Therefore

$$\overline{M}_{ip} = \lambda_i \overline{W}_p \qquad i = p+1, p+2, \ldots, N \tag{3.34}$$

Thus Eq. (3.32) becomes

$$\overline{W}_p = \overline{W}_0 + \tau_0(1 - \rho) + \sum_{i=p}^{N} \overline{x}_i \lambda_i \overline{W}_i + \sum_{i=p+1}^{N} \overline{x}_i \lambda_i \overline{W}_p \tag{3.35}$$

$$p = 1, 2, \ldots, N$$

Solving Eq. (3.35) for $\overline{W}_p$, we have

$$\overline{W}_p = \frac{\overline{W}_0 + \tau_0(1 - \rho) + \sum_{i=p+1}^{N} \rho_i \overline{W}_i}{1 - \sigma_p} . \tag{3.36}$$

$$p = 1, 2, \ldots, N$$

Solving recursively this triangular set of equations, that is, starting with $\overline{W}_N$ and from this finding $\overline{W}_{N-1}$ etc., we obtain Eq. (3.31)

We first observe that the conservation law stated in the previous section must hold. One can easily verify this by substituting $\overline{W}_p$ as given by Eq. (3.31) into Eq. (3.12).

We now pose the following optimization problem which has been stated and solved in [KLEI 76]:

How should we assign external priorities to customers, given there is a cost ratio of C_p dollars for each second of delay suffered by each customer from queue p?

Clearly the average cost per second to the system we denote by C must be

$$C = \sum_{p=1}^{N} C_p \overline{N}_p \tag{3.37}$$

where $\overline{N}_p$ is the average number of customers in the system generated at queue p. From Little's result we know that regardless of the queueing discipline, we have

$$\overline{N}_p = \lambda_p \overline{T}_p = \lambda_p(\overline{W}_p + \overline{x}_p)$$

and so

$$C = \sum_{p=1}^{N} \rho_p C_p + \sum_{p=1}^{N} C_p \lambda_p \overline{W}_p$$

To minimize C in an M/G/1 queue with rest period, which queueing discipline (non-preemptive and work-conserving) should we choose? The solution of this optimization problem (see [KLEI 76], Chapter 3) is that of all the possible non-preemptive work-conserving disciplines, the HOL discipline with the ordering given in Eq. (3.39) below is that which minimizes the average cost given in Eq. (3.37).

If we order without loss of generality the customers such that

$$\frac{C_1}{\overline{x}_1} \leq \frac{C_2}{\overline{x}_2} \leq \cdots \leq \frac{C_N}{\overline{x}_N} \tag{3.38}$$

then the optimal ordering is:

$$\text{queue } p > \text{queue } (p - 1) \qquad p = 2, \ldots, N \tag{3.39}$$

where the sign > denotes "higher priority than."

By the same argument, one can easily show that a HOL discipline with queue p < queue (p + 1), p = 1, 2, ..., N - 1, maximizes C.

In particular, choose $C_p = 1/\lambda_p$; the higher the traffic is, the lower is the cost of delay in the system.

From Little's result, C is merely the sum of the average times in system:

$$C = \sum_{p=1}^{N} \overline{T}_p$$

If we want to minimize this sum, we must then choose a HOL discipline that gives highest priority to 1, and lowest priority to N, if the following inequalities hold:

$$\rho_1 \leq \rho_2 \leq \cdots \leq \rho_N$$

and a HOL discipline with queue p < queue (p + 1), p = 1, 2, ..., N - 1 will maximize C.

As another example, choose $C_p = \overline{x}_p$; C becomes

$$C = \sum_{p=1}^{N} \rho_p \overline{W}_p + \sum_{p=1}^{N} \rho_p \overline{x}_p$$

Any queueing discipline solves minimum C, since by the conservation law, C is constant with respect to the queueing discipline. This is consistent with the fact that the inequalities (3.38) become

$$\frac{C_1}{\overline{x}_1} = \cdots = \frac{C_N}{\overline{x}_N} = 1$$

3.6 Alternating Priorities, N = 2 Users

We wish to solve for the delay of a packet generated at user i when the rule of selection among users is AP, whose protocol has been defined in Section 3.2.2. When the traffic is equally distributed among all users ($\lambda_i = \frac{\lambda}{N}$, i = 1, ..., N), the average delay is given by Eq. (3.27). When the rates of packet generation, λ_i, i = 1, ..., N are not equal, solving for delay is a more complicated problem and does not seem to result in explicit or closed forms, for N > 2. Below we analyze the particular case of N = 2 users.

The Model

The scheme defined in Section 3.2.2 is modeled as follows: We deal with a single server queueing system with arbitrary rest period denoted by T_0, with distribution function $R_0(\cdot)$, independent of service time and arrival processes. This queueing system serves several classes of customers. Customers of class i arrive at points of time generated by a Poisson process with intensity λ_i. A service time distribution function $B_i(\cdot)$ is associated with class i. Within a class, service times are identically distributed and are independent of each other and of other class service time distribution functions and of the arrival processes. Customers are selected in first-come-first-served order within classes.

The alternating priorities discipline decides from which class the next customer is to be selected, as follows:

i) Choose the next customer from the same class as was the customer whose service was just completed, if possible.

ii) If there are no more customers of the same class, the server then chooses one from a different class (see Section 3.2.2) and continues working on that class until the system is empty of this class members. If there are no customers at all from which to select, the server becomes idle and goes for a vacation (rest period) of length T_0. At the end of this rest period, the server begins to serve the customers that have accumulated during his absence. If there is no backlog, he will immediately take another rest period. If there is a backlog, the first arrival during the rest period obtains the attention of the server for members of his class.

For the case of (N = 2) classes, there is no need for an additional rule to select the next class when the switch is made.

For the case of a fixed size packet switching slotted system the rest period and service time of any class are identically distributed according to a deterministic distribution function.

Protocols for Starting a Busy Period

The model described above differs from the protocol defined in Section 3.2.2 by the rule of selecting a class when the server finds members of different classes at the end of a rest period.

While the model selects the first arrival, the protocol chooses a customer of class 1 (or a customer of the same class as the one who ended the previous busy period). Clearly, the former rule (first-come-first-served) is not easy to implement in our distributed environment. However, let us divide a rest period into minislots (of length τ) and make the following assumption, consistent with the nature of the arrival process (Poisson process) when the minislot is small:

(A.11) At most, one arrival may occur at any queue during one minislot.

With this assumption, the server can choose the class of the first arrival to start the busy period. Indeed if at the end of the minislot during which a packet is generated at user i, the carrier is absent then queue i knows it has priority (the packet generated at queue i was the first to be generated during the rest period). User i starts transmitting the carrier so that other users upon packet generation (if any) will detect the presence of the carrier and therefore know they do not have priority.

Average Delay: N = 2

The following notation will be used.

$\overline{x}_i$: average service time at queue i, i = 1, 2.

$\overline{x_i^2}$: second moment of service time distribution at queue i.

$B_i^*(s)$: Laplace-Stieljes Transform of service time at queue i.

$\overline{T}_i$: Average delay at queue i (i.e., sum of the average service time $\overline{x}_i$ and the average waiting time $\overline{W}_i$)

$S_i^*(s)$: Laplace-Stieljes transform of delay (service time plus waiting time) at queue i.

and as usual,

λ_i: input rate at queue i.

$\rho_i = \lambda_i \overline{x}_i$.

$\overline{T}_0$, $\overline{T_0^2}$: first and second moments of rest period distribution $R_0(\cdot)$.

$R_0^*(s)$: Laplace-Stieljes transform of the rest period.

The Laplace-Stieljes transform of time in system, and therefore the average delay, have been solved by Miller (MILL 64) for a system identical in all respects to our model, except that there is no rest period. (The idle period ends with the arrival of the first customer who starts service immediately.)

Using a similar approach, we solve for the Laplace-Stieljes transform of delay (or time in system) at each queue $S_i^*(s)$ and obtain the first moments, i.e., the average delay $\overline{T}_i$ at each queue.

Definitions

A cycle is defined as the length of time which elapses between the beginning of a rest period and the beginning of the following rest period. The length of a cycle is denoted by T_c. The busy period (time between the end of the rest period and the end of the cycle) has a length (which may be zero) denoted by T_b. Type i cycles are the cycles where the busy period starts with a class i customer (type i busy period); the length of a type i cycle is denoted by T_{ci} and the length of a type i busy period is denoted by T_{bi}.

Furthermore a busy period is divided into intervals called phases, the length of which is denoted by T_{ikj}, with Laplace-Stieljes transform $R^*_{ikj}(s)$ where T_{ikj} is, for class i customers, the length of the k^{th} sub-busy period within a busy period which started with the service of class j customers. We have

$$\left.\begin{aligned} T_{cj} &= T_0 + T_{bj} \\ T_{bj} &= \sum_{k=0}^{\infty} \sum_{i=1}^{2} T_{ikj} \end{aligned}\right\} \quad j = 1, 2 \tag{3.40}$$

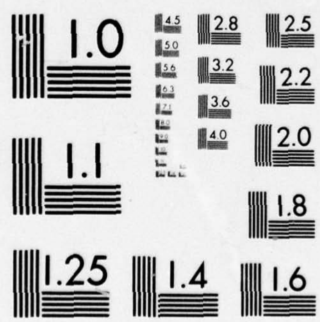
As an example, consider a type 1 cycle. As depicted in Figure 3.8, the busy period starts with a phase of length T_{111}; this phase is a busy period for class 1 customers who arrived during the rest period. This phase is followed by a phase of length T_{211}, which is a busy period for class 2 customers initiated by customers which arrived during the "rest period" of length $T_0 + T_{111}$.

AD-A039 018 CALIFORNIA UNIV LOS ANGELES DEPT OF COMPUTER SCIENCE F/G 9/2
ADVANCED TELEPROCESSING SYSTEMS.(U)
DEC 76 L KLEINROCK DAHC15-73-C-0368

UNCLASSIFIED NL

2 OF 4
AD
A039018

390



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

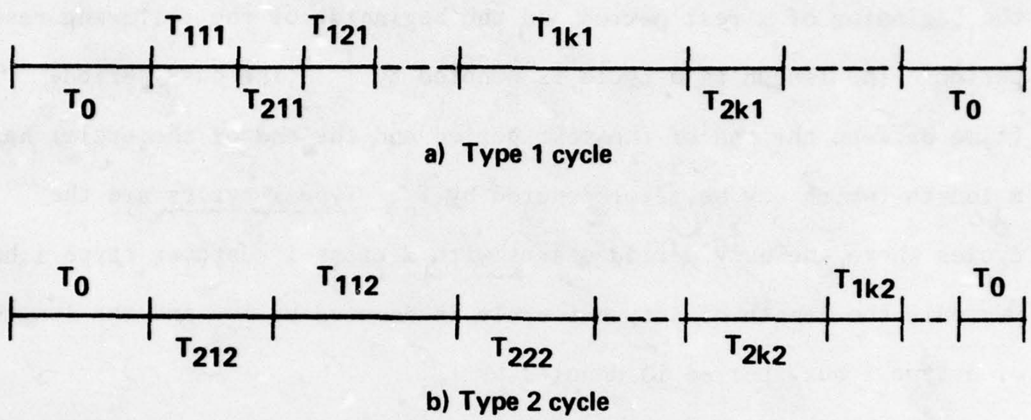


Figure 3.8 Examples of Cycles

The next phase, of length T_{121}, is a busy period for class 1 customers initiated by customers who arrived during the "rest period" of length T_{211}, etc.

To solve for the transform of time in system for class 1 customers, $S_1^*(s)$, it is enough to solve for the transform of time in system given they arrive in a type 1 cycle, denoted by $S_{11}^*(s)$ and for the transform of time in system given they arrive in a type 2 cycle, denoted by $S_{12}^*(s)$.

$S_1^*(s)$ is given by

$$S_1^*(s) = S_{11}^*(s)P\{\text{Class 1 customer arrived during a type 1 cycle}\} + S_{12}^*(s)P\{\text{Class 1 customer arrived during a type 2 cycle}\} \quad (3.41)$$

We have so far defined the following variables:

T_c: length of a cycle.

T_{ci}: length of a type i cycle (i = 1,2).

T_b: length of a busy period.

T_{bi}: length of a busy period in a type i cycle (i = 1,2).

T_{ikj}: length of the k^{th} phase of class i customers (i = 1,2). serviced within a type j cycle (j = 1, 2)

$R^*_{ikj}(s)$: Laplace-Stieljes transform of T_{ikj}.

$S^*_i(s)$: Laplace-Stieljes transform of delay at queue i (i = 1,2).

$S^*_{ij}(s)$: Laplace-Stieljes transform of delay at queue i (i = 1,2), given that the customer arrived during a type j cycle (j = 1, 2).

Laplace-Stieljes Transform of Delay at Queue i

Consider a type 1 cycle and a tagged customer arriving at queue 1 during a type 1 cycle.

For simplicity, we will denote by $\{T_{ikj}\}$, the interval of length T_{ikj} as long as there is no ambiguity about the instant when this interval starts.

If our tagged customer arrives during $\{T_0 + T_{111}\}$ he will be served during $\{T_{111}\}$. More generally if the tagged customer arrives during $\{T_{2k1} + T_{1,k+1,1}\}$ he will be served during $\{T_{1,k+1,1}\}$, for $k = 0, 1, 2, \ldots$ where of course $\{T_{201}\} = \{T_0\}$.

Now since the arrival process is Poisson, we have:

$$P\left\{\begin{array}{l}\text{customer arrives in } \{T_{2k1} + T_{1,k+1,1}\} \text{ given} \\ \text{he arrived in a type 1 cycle}\end{array}\right\}$$

$$= \frac{E\{T_{2k1} + T_{1,k+1,1}\}}{E\{T_{c_1}\}}$$

We therefore have the following relationship:

$$S^*_{11}(s) = \sum_{k=0}^{\infty} S^*_{11}(s/k) \cdot \frac{E\{T_{2k1} + T_{1,k+1,1}\}}{E\{T_{c_1}\}} \tag{3.42}$$

where $S^*_{11}(s/k)$ denotes the transform of our tagged customer delay, given he arrived in $\{T_{2k1} + T_{1,k+1,1}\}$. But $S^*_{11}(s/k)$ is the transform of time in system in a queue with rest period which we have solved in Appendix A. (See Eqs. (A.23) and (A.39)). Therefore we have

$$S^*_{11}(s/k) = \frac{B^*_1(s)\,[1 - R^*_{2k1}(s)]}{E\{T_{2k1} + T_{1,k+1,1}\}[\lambda_1 B^*{}_1(s) + s - \lambda_1]} \tag{3.43}$$

Substituting Eq. (3.43) into Eq. (3.42) we get

$$S^*_{11}(s) = \frac{B^*_1(s)}{E\{T_{c_1}\}[\lambda_1 B^*_1(s) + s - \lambda_1]} \sum_{k=0}^{\infty} [1 - R^*_{2k1}(s)] \tag{3.44}$$

where $R^*_{201}(s) \triangleq R^*_0(s)$

by similar derivations, we get for type 2 cycles:

$$S^*_{12}(s) = \frac{B^*_1(s)}{E\{T_{c_2}\}[\lambda_1 B^*_1(s) + s - \lambda_1]} \sum_{k=1}^{\infty} [1 - R^*_{2k2}(s)] \tag{3.45}$$

where R^*_{212} is defined as the transform of $T_0 + T_{212}$, with second moment $E[(T_{212})^2] \triangleq E[(T_0 + T_{212})^2]$.

Since we will never have to consider either the transform of the interval of length T_{212} alone or its second moment, we choose this rather ambiguous notation. However, we keep $\overline{T_{212}}$ as the average length of the first phase of service of type 2 customers in a type 2 cycle.

It remains to evaluate:

P{class 1 customer arrived during type j cycle}

Since the arrival process is Poisson, this must be equal to the fraction of time represented by type j cycles, divided by the expected length of a cycle $E\{T_c\}$. But the fraction of time represented by type j cycles is equal to the expected length of a type j cycle multiplied by the frequency of occurrence of type j cycles.

Thus, we have

$$P\{\text{class 1 customer arrived during type j cycle}\} = \frac{\frac{\lambda_j}{\lambda_1 + \lambda_2} E\{T_{c_j}\}}{E\{T_c\}} \quad j=1,2 \tag{3.46}$$

Substitution of Eqs. (3.44), (3.45) and (3.46) into Eq. (3.41) gives an expression for the transform of time in system of our tagged customer:

$$S_1^*(s) = \frac{B_1^*(s)}{(\lambda_1 + \lambda_2)E\{T_c\}[\lambda_1 B_1^*(s) + s - \lambda_1]}\left(\lambda_1 \sum_{k=0}^{\infty} [1 - R_{2k1}^*(s)] + \lambda_2 \sum_{k=1}^{\infty} [1 - R_{2k2}^*(s)]\right) \tag{3.47}$$

To evaluate $E\{T_c\}$ we observe that this is merely the expected length of a cycle for a queue with a rest period. We therefore have (see Eq. (A.39))

$$E\{T_c\} = \frac{\overline{T}_0}{1 - \rho} \tag{3.48}$$

where $\rho = \lambda\overline{x} = (\lambda_1 + \lambda_2)\left(\frac{\lambda_1}{\lambda_1 + \lambda_2}\overline{x}_1 + \frac{\lambda_2}{\lambda_1 + \lambda_2}\overline{x}_2\right) = \rho_1 + \rho_2$

The final expression for $S_1^*(s)$ is given by

$$S_i^*(s) = \frac{B_1^*(s)(1 - \rho)}{\lambda\overline{T}_0[\lambda_1 B_1^*(s) + s - \lambda]}\left(\lambda_1 \sum_{k=0}^{\infty} [1 - R_{2k1}^*(s)] + \lambda_2 \sum_{k=1}^{\infty} [1 - R^*_{2k2}(s)]\right) \tag{3.49}$$

We have been unable to find a closed expression. The sums appearing in Eq. (3.49) will converge if $\rho < 1$, since then the busy period will end with probability one.

Taking the derivative at $s = 0$ of the right-hand side of Eq. (3.49) term by term, we have, after some algebraic manipulation:

$$\overline{T}_1 = \overline{x}_1 + \frac{\lambda_1 \overline{x_1^2}}{2(1 - \rho_1)} + \frac{1 - \rho}{2\lambda\overline{T}_0(1 - \rho_1)}\left[\lambda_1 \sum_{k=0}^{\infty} \overline{(T_{2k1})^2} + \lambda_2 \sum_{k=1}^{\infty} \overline{(T_{2k2})^2}\right] \tag{3.50}$$

where the bar is used for the expected value of a random variable.

This formula takes the form of the Pollaczek-Khinchin relation plus an additional quantity which remains to be evaluated.

$\overline{(T_{2k1})^2}$ is the square moment of the k^{th} phase of service for class 2 customers given a type 1 cycle. These customers arrive either in the "rest period" $\{T_{1k1}\}$ or during the busy period $\{T_{2k1}\}$. From Appendix A we know the expression of the second moment of the busy period in a queue with rest period, in terms of the parameters of the system (Eq. (A. 42)). We then have

$$\overline{(T_{2k1})^2} = \frac{\lambda_2 \overline{x_2^2}}{(1-\rho_2)^3} \overline{T_{1k1}} + \frac{\rho_2^2}{(1-\rho_2)^2} \overline{(T_{1k1})^2} \qquad k \geq 2$$

$$\overline{(T_{211})^2} = \frac{\lambda_2 \overline{x_2^2}}{(1-\rho_2)^3} \overline{(T_0 + T_{111})} + \frac{\rho_2^2}{(1-\rho_2)^2} \overline{(T_0 + T_{111})^2} \qquad k = 1$$

$$\overline{(T_{201})^2} = \overline{T_0^2} \qquad k = 0 \tag{3.51}$$

Similarly, we have

$$\overline{(T_{2k2})^2} = \frac{\lambda_2 \overline{x_2^2}}{(1-\rho_2)^3} \overline{T_{1,k-1,2}} + \frac{\rho_2^2}{(1-\rho_2)^2} \overline{(T_{1,k-1,2})^2} \qquad k \geq 2$$

$$\overline{(T_{212})^2} = \frac{\lambda_2 \overline{x_2^2}\,\overline{T_0}}{(1-\rho_2)^3} + \frac{\overline{T_0^2}}{(1-\rho_2)^2} \qquad k = 1 \tag{3.52}$$

The last equation is obtained by observing that $\overline{(T_{212})^2}$ is the second moment of the interval which has been defined as the union of the rest period $\{T_0\}$ and the first phase of class 2 customers' service in a type 2 cycle. This is precisely the second moment of the cycle (rest period and busy period) in a queue with rest period that we can get by taking the second derivative of the cycle time transform (see Appendix A Eq. (A.43)). Similarly we get

$$\overline{(T_{1k2})^2} = \frac{\lambda_1 \overline{x_1^2}}{(1-\rho_1)^3} \overline{T_{2k2}} + \frac{\rho_1^2}{(1-\rho_1)^2} \overline{(T_{2k2})^2} \qquad k \geq 2$$

$$\overline{(T_{112})^2} = \frac{\lambda_1 \overline{x_1^2}}{(1-\rho_1)^3} \overline{(T_0 + T_{212})} + \frac{\rho_1^2}{(1-\rho_1)^2} \overline{(T_0 + T_{212})^2} \qquad k = 1$$

$$\overline{(T_{102})^2} = \overline{T_0^2} \qquad k = 0 \tag{3.53}$$

and

$$\overline{(T_{1k1})^2} = \frac{\lambda_1 \overline{x_1^2}}{(1-\rho_1)^3} \overline{T}_{2,k-1,1} + \frac{\rho_1^2}{(1-\rho_1)^2} \overline{(T_{2,k-1,1})^2} \qquad k \geq 2$$

$$\overline{(T_{111})^2} \triangleq E\left[(T_0 + T_{111})^2\right] = \frac{\lambda_1 \overline{T}_0 \overline{x_1^2}}{(1-\rho_1)^3} + \frac{\overline{T_0^2}}{(1-\rho_1)^2} \qquad k = 1$$

(3.54)

We wish to solve for $\sum_{k=0}^{\infty} \overline{(T_{2k1})^2}$ and $\sum_{k=1}^{\infty} \overline{(T_{2k2})^2}$. Eq. (3.51) and Eq. (3.52) summed over the indicated values of k will contain sums of the first moments of T_{ikj}, namely

$$\sum_{k=1}^{\infty} \overline{T}_{1k1} \;, \quad \sum_{k=1}^{\infty} \overline{T}_{1k2}$$

To solve for the first moments of $\overline{T}_{ikj}$, we will once more observe that in a queue with rest period (Eq. (A.40)),

$$\overline{T}_b = \frac{\rho}{1-\rho} \overline{T}_0$$

and apply this last relationship to each phase of service of type 1 customers in either type 1 cycle or type 2 cycle. We get, after summing over k,

$$\sum_{k=1}^{\infty} \overline{T}_{1k1} = \frac{\rho_1}{1-\rho_1} \sum_{k=1}^{\infty} \overline{T}_{2k1} + \frac{\rho_1}{1-\rho_1} \overline{T}_0$$

$$\sum_{k=1}^{\infty} \overline{T}_{1k2} = \frac{\rho_1}{1-\rho_1} \sum_{k=1}^{\infty} \overline{T}_{2k2} + \frac{\rho_1}{1-\rho_1} \overline{T}_0 \qquad (3.55)$$

Similarily we get symmetric expressions for $\sum_{k=1}^{\infty} \overline{T}_{2k2}$ and $\sum_{k=1}^{\infty} \overline{T}_{2k1}$ and by substituting the latter in Eq. (3.55), we have

$$\sum_{k=1}^{\infty} \overline{T}_{1k1} = \sum_{k=1}^{\infty} \overline{T}_{1k2} = \frac{\rho_1 \overline{T}_0}{1 - \rho} \qquad (3.56)$$

which is indeed the fraction of time the server is busy with type 1 customers in a cycle $\left(\overline{T}_C = \frac{\overline{T}_0}{1 - \rho}\right)$.

$$\sum_{k=1}^{\infty} \overline{T}_{2k2} = \sum_{k=1}^{\infty} \overline{T}_{2k1} = \frac{\rho_2 \overline{T}_0}{1 - \rho}$$

This is the fraction of time in a cycle that the server is busy with type 2 customers. Summing over the indicated values of k, Eqs. (3.51), (3.52), (3.53) and (3.54), substituting Eq. (3.56) we get four equations with four unknowns $\sum_k \overline{(T_{ikj})^2}$ (i = 1, 2 ; j = 1, 2). By solving this system in $\sum_k \overline{(T_{2k1})^2}$ and in $\sum_k \overline{(T_{2k2})^2}$ we eventually have:

$$\sum_{k=0}^{\infty} \overline{(T_{2k1})^2} = \frac{\overline{T}_0}{1 - \rho} \left[\frac{\lambda_2 \overline{x_2^2}(1 - \rho_1)^2 + \lambda_1 \rho_2^2 \overline{x_1^2}}{(1 - \rho_1)^2 (1 - \rho_2)^2 - \rho_1^2 \rho_2^2} \right] + \overline{T_0^2} \left[1 + \frac{\rho_2^2}{(1 - \rho_1)^2 (1 - \rho_2)^2 - \rho_1^2 \rho_2^2} \right] \qquad (3.57)$$

and

$$\sum_{k=1}^{\infty} \overline{(T_{2k2})^2} = \frac{\overline{T}_0}{1 - \rho} \left[\frac{\lambda_2 \overline{x_2^2}(1 - \rho_1)^2 + \lambda_1 \rho_2^2 \overline{x_1^2}}{(1 - \rho_1)^2 (1 - \rho_2)^2 - \rho_1^2 \rho_2^2} \right] + \frac{\overline{T_0^2}(1 - \rho_1)^2}{(1 - \rho_1)^2 (1 - \rho_2)^2 - \rho_1^2 \rho_2^2} \qquad (3.58)$$

Substitution of Eqs. (3.57) and (3.58) into Eq. (3.50) finally gives

$$\overline{T}_1 = \overline{x}_1 + \frac{\lambda_1 \overline{x_1^2}}{2(1-\rho_1)} + \frac{\lambda_2(1-\rho_1)^2\overline{x_2^2} + \lambda_1\rho_2^2\overline{x_1^2}}{2(1-\rho_1)(1-\rho)[(1-\rho_1)(1-\rho_2)+\rho_1\rho_2]}$$

$$+ \frac{\overline{T_0^2}}{2\overline{T}_0}\left[\frac{\lambda_1}{\lambda} + \frac{\lambda_2(1-\rho_1) - \lambda_1\rho_2(1-2\rho_2)}{\lambda[(1-\rho_1)(1-\rho_2)+\rho_1\rho_2]}\right] \tag{3.59}$$

and similarily

$$\overline{T}_2 = \overline{x}_2 + \frac{\lambda_2 \overline{x_2^2}}{2(1-\rho_2)} + \frac{\lambda_1(1-\rho_2)^2\overline{x_1^2} + \lambda_2\rho_1^2\overline{x_2^2}}{2(1-\rho_2)(1-\rho)][(1-\rho_1)(1-\rho_2)+\rho_1\rho_2]}$$

$$+ \frac{\overline{T_0^2}}{2\overline{T}_0}\left[\frac{\lambda_2}{\lambda} + \frac{\lambda_1(1-\rho_2) - \lambda_2\rho_1(1-2\rho_1)}{\lambda[(1-\rho_1)(1-\rho_2)+\rho_1\rho_2]}\right] \tag{3.60}$$

It is interesting that these formulas take the form of the relation obtained by Miller [MILL 64] (Alternating Priorities in an M/G/1 queue without rest period) plus an additional term (the fourth term) relating the supplementary delay due to the fact that a busy period may start only at the end of a rest period.

The first two terms are those of the Pollaczek-Khinchin formula, the third term is due to the time periods during which the server is unavailable to customers of the class in question.

We check the conservation law and find that Eq. (3.12) is verified, i.e.,

$$\rho_1\overline{W}_1 + \rho_2\overline{W}_2 = \rho\,\frac{\lambda_1\overline{x_1^2} + \lambda_2\overline{x_2^2}}{2(1-\rho)} + \rho\,\frac{\overline{T_0^2}}{2\overline{T}_0}$$

where $\overline{W}_i = \overline{T}_i - \overline{x}_i$, $i = 1, 2$

which is the average waiting time at queue i.

Furthermore, if $\lambda_1 = \lambda_2 = \lambda/2$, $\overline{x}_1 = \overline{x}_2 = \overline{T}_0 = s$ and $\overline{x_1^2} = \overline{x_2^2} = \overline{T_0^2} = s^2$, then

$$\overline{W_1} = \overline{W_2} = \frac{s}{2(1 - \rho)} \qquad (\rho = \lambda s)$$

which is, as we might have expected, consistent with Eq. (3.27).

3.7 Delay Variance under the Protocols AP, RR and RO

The analysis of Section 3.4 predicted the same average packet delay under the protocols AP, RR and RO, provided that each queue has the same input rate. It is therefore necessary to look for higher moments of the delay distribution in order to compare these three protocols in terms of delay.

The delay variance under these protocols has been obtained from simulation only. It turns out that the variance is only slightly different from one scheme to the other, as shown in Figures 3.9, 3.10 and 3.11 below.

In the first two figures, the delay variance (expressed in $(\text{slots})^2$) is plotted versus the total input rate normalized with respect to a slot (i.e., ρ), respectively for N = 2 users and N = 5 users.

The delay variance under three special disciplines in a slotted system are also plotted:

First-come-first-served (FCFS)

Random order of service (ROS)

Last-come-first-served (LCFS)

The first system is a special case of FCFS M/G/1 queue with rest period. In Appendix A, we give the delay variance in such a queueing

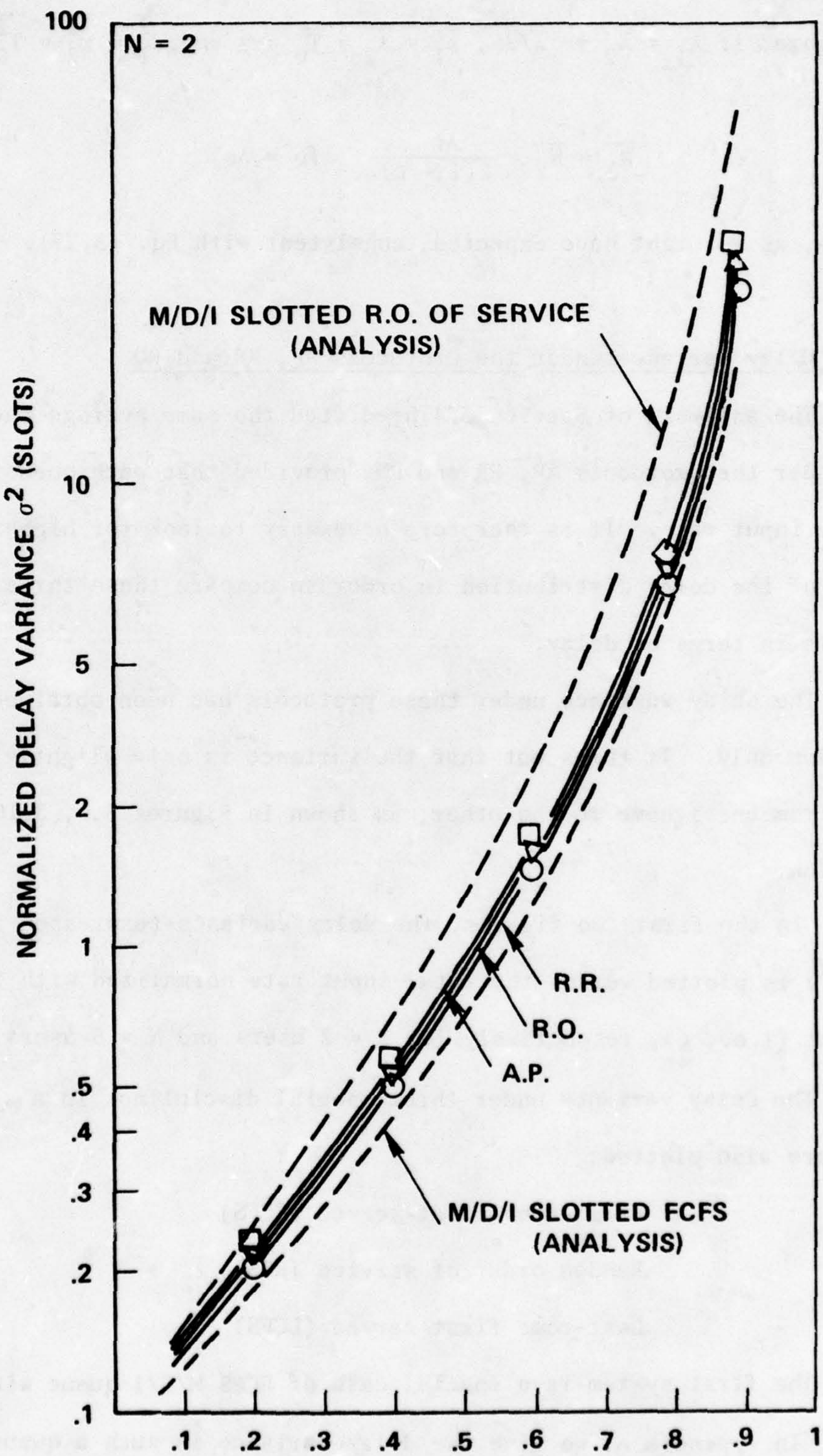


Figure 3.9. σ^2 vs ρ (from simulation).

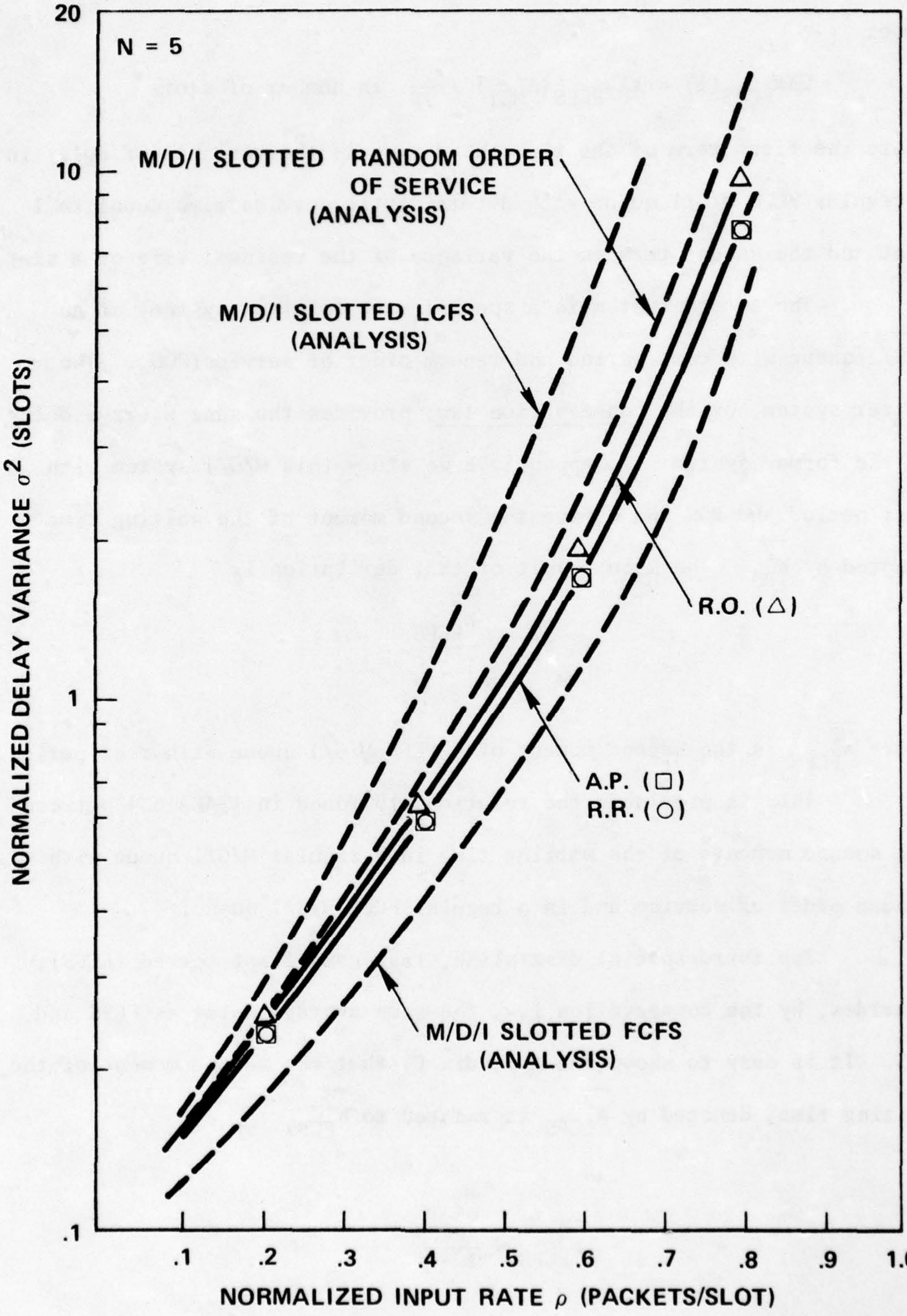


Figure 3.10. σ^2 vs ρ (from simulation)

system [Eq. (A.34)]. In the particular case of a slotted system we have:

$$VAR_{FCFS}(\Delta) = VAR_{FCFS}(\Delta_{MG1}) + \frac{1}{12} \text{ in number of slots}$$

where the first term of the right hand side is the variance of delay in a regular FCFS M/G/1 queue with deterministic service time equal to 1 slot and the second term is the variance of the residual life of a slot.

The second system is a special case (slotted system) of an M/G/1 queue with rest period and random order of service(ROS). The latter system, by the conservation law, provides the same average delay as the former system. In Appendix B we study this M/G/1 system with rest period and ROS and derive the second moment of the waiting time denoted by $\overline{W^2_{RO}}$. The main result of that derivation is

$$\overline{W^2_{RO}} = \frac{\overline{W^2_{FCFS}}}{1 - \frac{\rho}{2}}$$

where $\overline{W^2_{FCFS}}$ is the second moment of a FCFS M/G/1 queue with rest period.

This is precisely the relationship found in [TAKA 63] between the second moments of the waiting time in a regular M/G/1 queue with random order of service and in a regular FCFS M/G/1 queue.

The third special discipline, last-come-first-served (LCFS), provides, by the conservation law, the same average delay as FCFS and ROS. It is easy to show (see Appendix C) that the second moment of the waiting time, denoted by $\overline{W^2_{LCFS}}$ is related to $\overline{W^2_{FCFS}}$ by

$$\overline{W^2_{LCFS}} = \frac{\overline{W^2_{FCFS}}}{1 - \rho}$$

This is precisely the relationship found in [TAKA 63] between the second moments of the waiting time in a regular (no rest period) LCFS M/G/1 queue and in a regular FCFS M/G/1 queue.

In Summary, the following relationship between $\overline{W^2_{FCFS}}$, $\overline{W^2_{RO}}$, $\overline{W^2_{LCFS}}$ holds for the M/G/1 queue with rest period as well as for the regular (without rest period) M/G/1 queue.

$$\overline{W^2_{FCFS}} = (1 - \frac{\rho}{2})\overline{W^2_{RO}} = (1 - \rho)\overline{W^2_{LCFS}} \tag{3.61}$$

Kingman showed that under a work-conserving, non-preemptive queueing discipline (independent of the service time) in a G/G/1 queue (without rest period) the variance of the waiting time is not less than under the FCFS discipline [KING 62]. Recently, Vasicek [VASI 76] showed the following interesting and general result.

Theorem:

The expected value of any convex function of the waiting time (such as the variance) in a general single-server queue under a general queueing discipline (independent of the service time) does not exceed that under the LCFS discipline, and is not less than that under the FCFS discipline.

It is easy to show that the same result holds for an M/G/1 queue with rest period. Let us apply the Kingman argument to an M/G/1 with rest period to show that the variance of the waiting time is not less than under the FCFS discipline. Using the same type of argument we then show that the variance of the waiting time does not exceed that under the LCFS discipline.

Consider the customers c_1, ..., c_n, who are served in any one particular busy period. Suppose c_i arrives at time t_i and has a waiting time w_i; if the instants when the server begins to serve a new customer are s_j, there exists a permutation R(j) of (1, 2, ..., n) such that $c_{R(j)}$ enters service at s_j and then

$$w_{R(j)} = s_j - t_{R(j)}$$

$$t_{R(k)} \leq s_k \qquad k = 1, 2, \ldots, n$$

The effect of the queueing discipline is to determine R(j). The average waiting time over this particular busy period is

$$\overline{w} = \frac{1}{n} \sum_{j=1}^{n} (s_j - t_{R(j)}) = \overline{s} - \overline{t}$$

where $\overline{s}$ and $\overline{t}$ are the averages over the busy period of the instants when the server begins to serve a new customer and of the arrival times.

$$\overline{w^2} = \frac{1}{n} \sum_{j=1}^{n} (s_j - t_{R(j)})^2 = \overline{s^2} + \overline{t^2} - \frac{2}{n} \sum_{j-1}^{n} s_j t_{R(j)}$$

Therefore, $\overline{w}$ is independent of R(j). $\overline{w^2}$ is least when $\sum_{i=1}^{n} s_j t_{R(j)}$ is maximum; $\overline{w^2}$ is maximum when $\sum_{i=1}^{n} s_j t_{R(j)}$ is minimum. It is well known [HARD 34] that this occurs respectively when the sets s_j and $t_{R(j)}$ are respectively similarly ordered and in reverse order. Therefore the R(j) permutation is respectively:

$R_1(j) = (1, 2, \ldots, n)$ which corresponds to FCFS, and

$R_2(j) = \max [i \ni i \neq R_2(k),\ k = 1, 2, \ldots, j - 1;\ t_i \leq s_j],$

$j = 1, 2, \ldots, n$

which corresponds to LCFS.

If we now average over a large number of busy periods, we see that the mean waiting time is independent of the queueing discipline, while the variance attains its minimum when customers are served in order of arrival, and attains its maximum when customers are served in reverse order of arrival.

In Figure 3.11, the delay variance normalized with respect to s and obtained by simulation under AP,RR and RO, is plotted versus the number of users N for a value of $\rho = 0.6$.

When N increases, the smallest variance is to be expected under AP and the largest under RR, the difference between the two being less than 1 $(\text{slot})^2$, while the variance under RO converges to that of a M/G/1 queue (slotted) with random order of service; this result was to be expected since when N is very large, there is at most one packet waiting at each user. Therefore, to select at random which of the (non-idle) users will transmit a packet is equivalent to selecting randomly one packet among all packets present in one queue.

We may thus conclude that the three protocols are quite equivalent in terms of throughput delay performance. This is no longer true, when the input rate is not the same at all users. In Figure 3.12 we take an example of N=2 users and plot the average packet delay (expressed in slots) at user 2 versus user 2's input rate ρ_2 (packets/slot) for various values of user 1's input rate ρ_1 under AP,RR and RO.

First we observe that if $\rho_1 = \rho_2$, the three protocols provide the same delay. This we know from Section 3.4 (see Eq. (3.27)). When $\rho_2 < \rho_1$, RR and AP provide respectively the smallest delay and the largest delay at user 2. But from the conservation law (see Eq. (3.12)) any attempt to modify the queueing discipline so as to reduce W_2 will

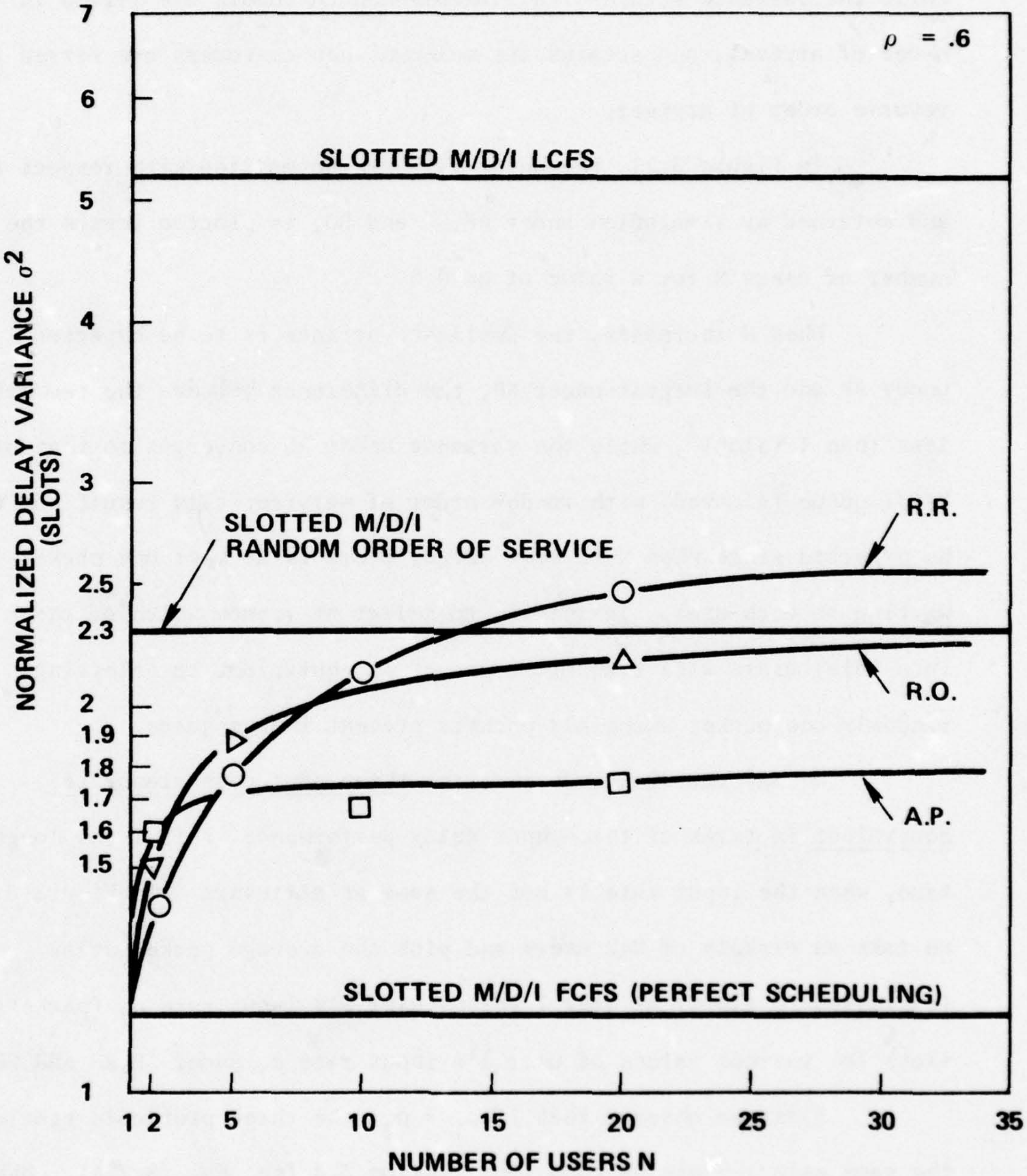


Figure 3.11. Effect of the Number of Users on the Delay Variance.

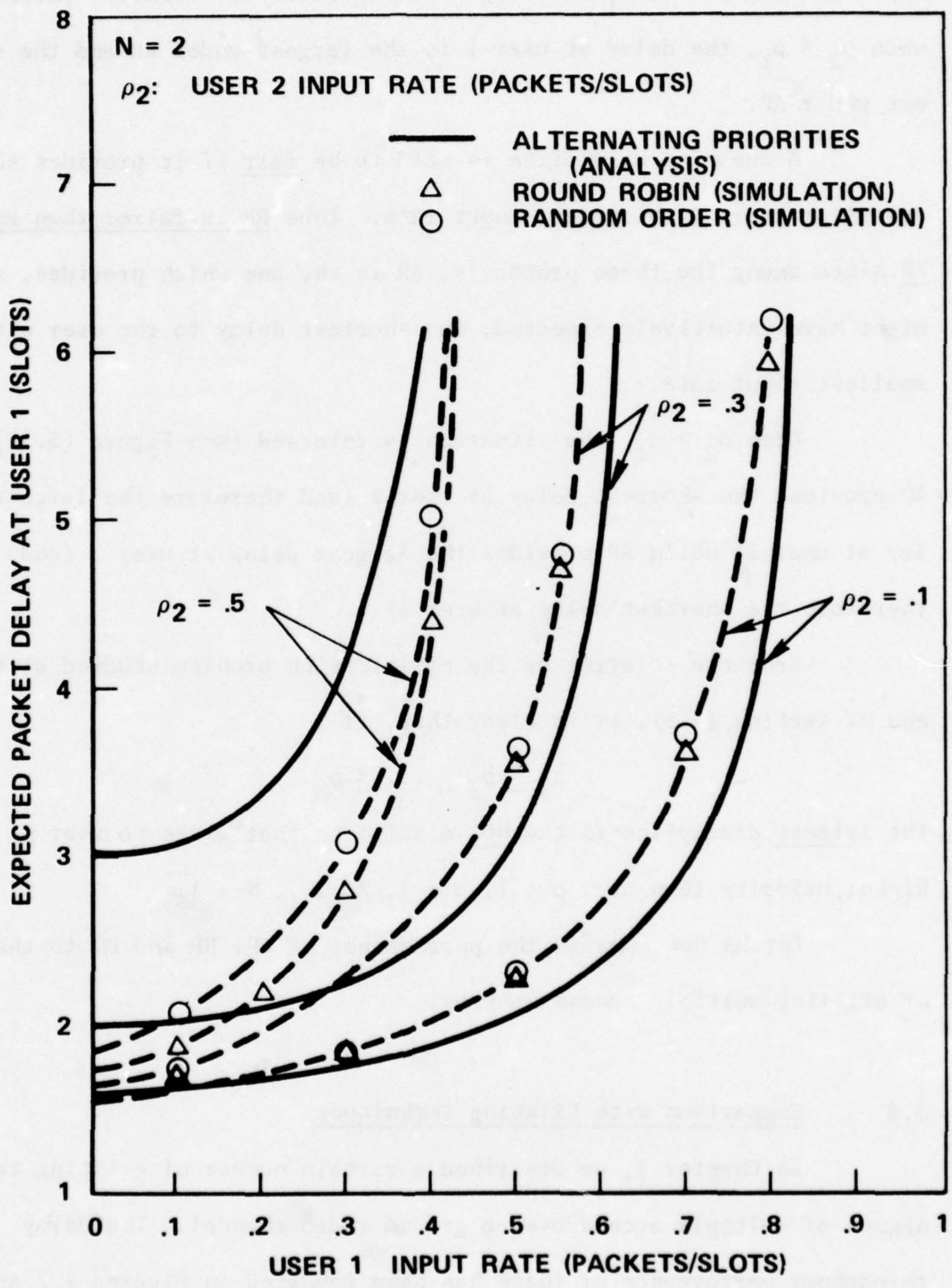


Figure 3.12. N = 2 Users: Effect of User 2's Input Rate on User 1's Performance.

force an increase in W_1 in a way which balances the result. Therefore, when $\rho_2 < \rho_1$, the delay at user 1 is the largest under RR and the shortest under AP.

A queueing discipline is said to be fair if it provides shorter delays to users with smaller input rate. Thus RR is fairer than RO and AP since among the three protocols, RR is the one which provides, as we might have intuitively expected, the shortest delay to the user with the smallest input rate.

When $\rho_2 > \rho_1$, the situation is reversed (see Figure (3.12)). AP provides the shortest delay at user 2 (and therefore the largest delay at user 1) while RR provides the largest delay at user 2 (and therefore the shortest delay at user 1).

From the solution of the optimization problem studied at the end of section (3.5), it is clear that, if

$$\rho_1 \leq \rho_2 \leq \cdots \leq \rho_N$$

the fairest discipline is the HOL discipline that gives to user p higher priority than user p + 1, p = 1, 2, ..., N - 1.

Let us now compare the performance of AP, RR and RO to that of existing multiple access schemes.

3.8 Comparison with Existing Techniques

In Chapter 1, we described a certain number of existing techniques of multiple access over a ground radio channel. The delay throughput performance of these has been compared in Figures 1.2 and 1.3.

The best access modes appeared to be slotted non-persistent CSMA, Polling, and the reservation scheme referred to as CS SRMA.

Depending on the environment and the system parameters, one scheme is more suitable than the others.

In Figures 3.13, 3.14 and 3.15, we plot the average packet delay normalized with respect to the packet transmission time versus the throughput for CSMA, Polling, CS SRMA and for our new schemes (AP, RR and RO).

As in our comparative study of existing techniques (Chapter 1) the following parameters' values have been chosen:

The ratio of the maximum propagation time over the packet transmission time is $a = .01$. The ratio of the request packet length over the information packet length in CS SRMA has been chosen to be $\eta = .01$. One minislot is assumed to be enough to transmit a polling message so that the polling time r (see Chapter 1) is:

$$r = 1 + 2 = 3 \text{ minislots}$$

As an example, consider 1000 bit packets transmitted over a channel operating at a speed of 100 kilobits/second. If the maximum distance between the source and destination is 20 miles, then $a \approx .01$. The request packet in CS SRMA has a length of 10 bits. The slot size in Polling is equal to the maximum propagation time (one minislot), thus the polling message has a length of 10 bits.

Figure 3.13 depicts an example of performance for N = 10 users. The number of users being small, the channel capacity achieved under AP, RR and RO is close to 1 (.91) and the delay under those protocols is slightly larger than that under Polling, for all values of the throughput rate. At very light traffic, however, the delay is larger under AP, RR and RO because of the slot size. Indeed, a packet

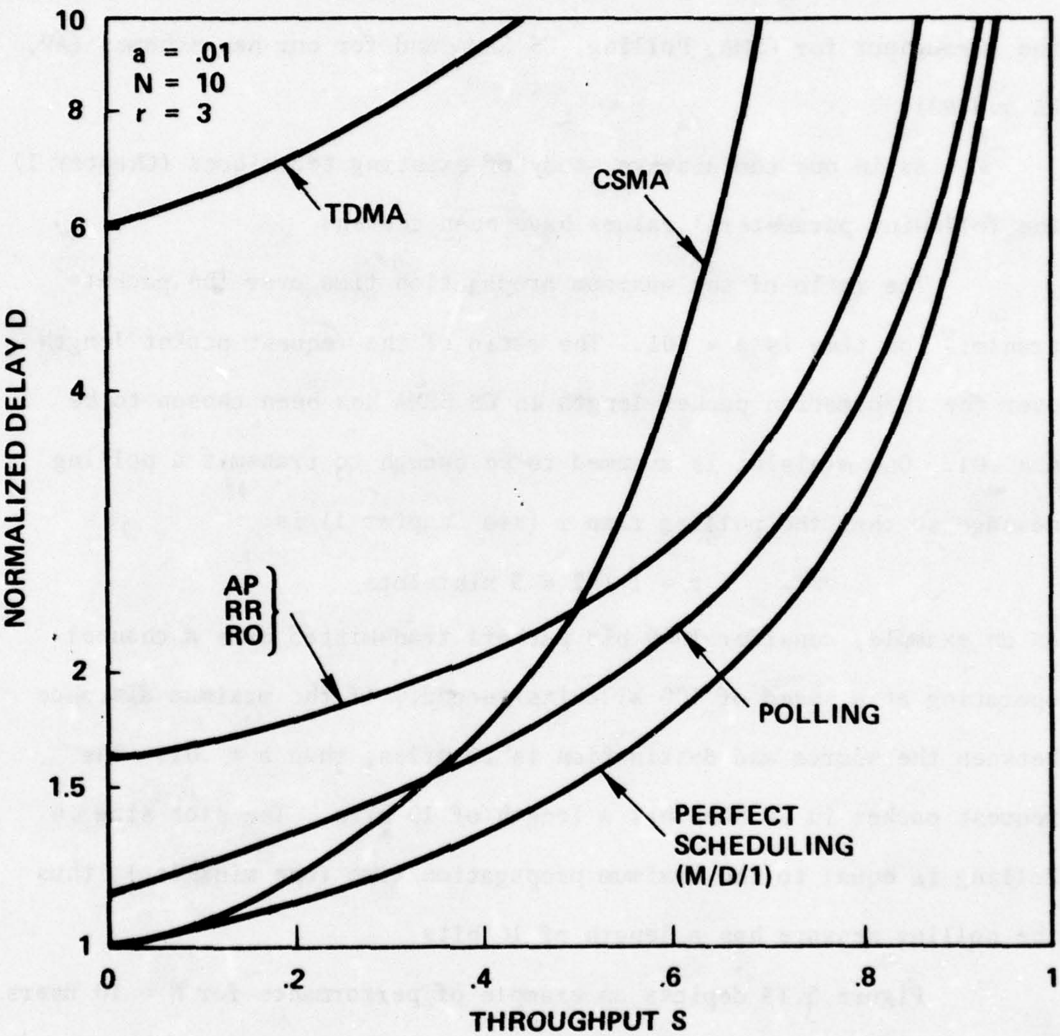


Figure 3.13. Average Packet Delay vs Throughput: Comparison To Existing Techniques (N = 10; a = .01)

which upon generation finds the system empty must wait, on the average, for half a slot before starting transmission, while under Polling such a packet would be delayed, on the average, only by the time taken to poll half the number of queues; this time is much less than half of a large slot (in Polling, the slot size is much smaller, being on the order of the maximum propagation time τ).

Let us now compare AP, RR and RO to CSMA. In Figure 3.13 we plot the CSMA performance predicted by the infinite population model [KLEI 75B]. As mentioned before (see Section 1.2.4), it was shown in [TOBA 76C] that the performance predicted by this model is a very good approximation to the performance of N = 10 buffered users contending for the channel under CSMA. We note from Figure 3.13, that at light traffic, CSMA provides the shortest delays. But when S is greater than .5, the new schemes perform much better than CSMA.

CS SRMA performance has not been plotted in Figure 3.13 since the model that predicted this performance is not suitable for a small number of queues. Indeed when N is small, buffering capabilitites are needed at each user; the CS SRMA model assumes that a user cannot generate a new packet when it is already in the process of transmitting a packet.

In Figure 3.14, a = .001. Even for a more significant number of users (N = 50), AP, RR and RO provide a performance comparable to that under Polling or CS SRMA. We did not plot the performance of CS SRMA predicted by the infinite population model [TOBA 76A] for the sake of clarity, since it lies between that of Polling and that of the new schemes. The performance of CS SRMA (unstable channel: no steady

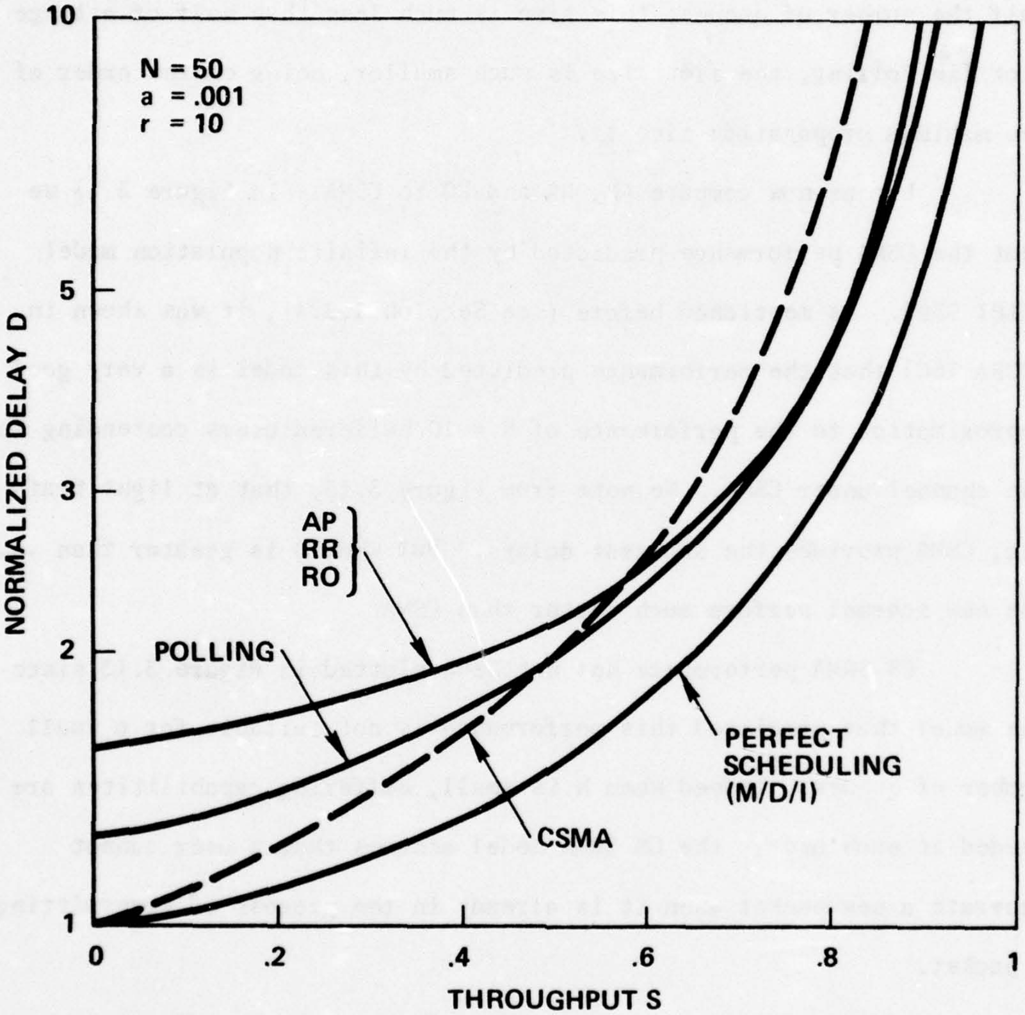


Figure 3.14. D vs S: Comparison To Existing Techniques (N = 50, a = .001)

state over an infinite time horizon) is likely an upper bound for the steady state performance (stable channel) for N = 50 users, which performance has not been studied. However, under heavy traffic conditions, the infinite population model performance is very accurate. The same remark holds for CSMA (the steady state performance of which has been studied [TOBA 76B] but was not available for N = 50 and a = .001).

Now comparing AP, RR and RO to CSMA, we note from Figure 3.14, as might be expected, that at light traffic, CSMA provides the shortest delays. Under heavy traffic conditions, the new schemes perform better than CSMA.

In the last example (Figure 3.15) we choose N = 50 and a = .01. The delay is significantly higher under the new protocols. At very small traffic ($S \simeq 0$), the delay is 1.5 slots, i.e., 2.25 in number of packet transmission times since the overhead in each slot is Na = .5; and this gives 1.5 slots times 1.5 packet transmission times/slot = 2.25. The capacity of the channel is only 2/3, while it is .84 for CSMA [TOBA 76B], greater than .9 for CS SRMA and 1 for Polling. When the traffic is not too high CS SRMA provides a better performance than Polling, for which the delay due to polling, proportional to $\frac{Nra}{2(1 - S)}$ (Eq. (1.3)) is very significant. This trend will increase with N, the number of users. However when N is very large, one may hypothesize higher delays under CS SRMA than that predicted by the infinite population model, since in order to have a stable channel the retransmission time of a conflicted request must be increased.

In <u>conclusion, for a small number of users*, AP, RR and RO</u>

---

*In particular, when the product Na is small. A typical value is $Na \leq .1$, e.g., N = 10, a = .01 or N = 20, a = .005.

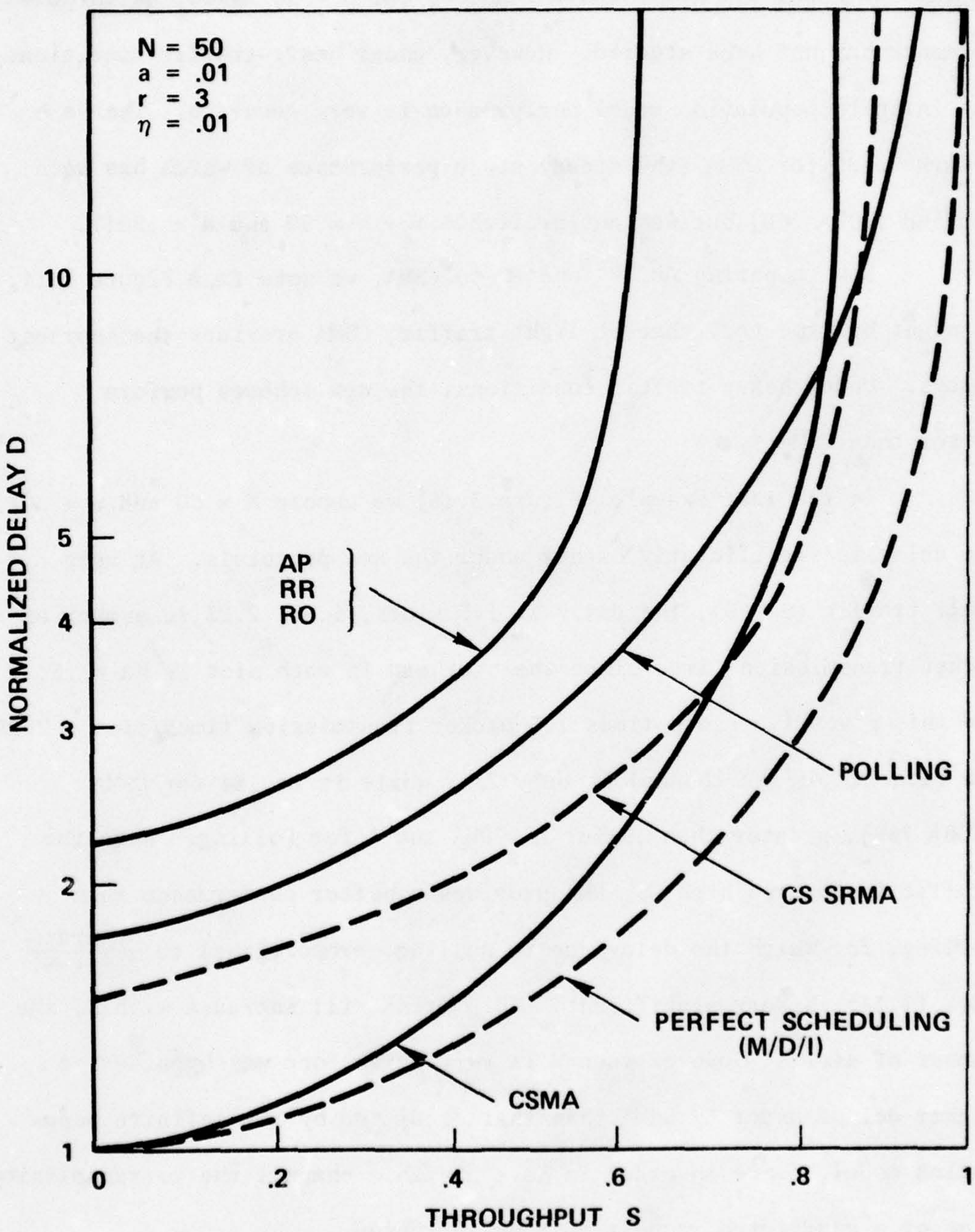


Figure 3.15. D vs S : Comparison to Existing Techniques (N = 50, a = .01)

provide a high channel capacity ($C = \frac{1}{1 + Na} > .9$) and a delay-throughput performance close to that obtained with Polling. The new protocols are particularly suitable for multiple access from a small number of buffered users without the control from a central station (Figure 3.13).

When all users are very close to each other (a small, e.g., a = .001), AP, RR and RO accept a significant number of users ($N \leq 50$) without performance degradation and under heavy traffic conditions they perform as well as CS SRMA and Polling and better than CSMA (Figure 3.14). But Polling does not require all users to be in line of sight and in range of each other. However the new schemes, as CSMA, have the advantage of not requiring the control from a "master" user (central station) while Polling and CS SRMA do.

When a is not too small (a = .01) the performance degrades with the number of users. Indeed in each slot, the overhead is proportional to the number of users. In the following chapter, we modify the protocols so as to decrease the overhead in order to reduce the performance degradation.

CHAPTER 4

RANDOM ORDER WITH ALOHA CONTENTION

When the number of users increases, we observe a degradation of the delay-throughput performance with the multiple access modes studied in Chapter 3. This degradation is due to the large overhead existing in each slot, which is equal to N minislots. This suggests the following modification of the slot configuration and of the protocols described in Chapter 3 so as to reduce the overhead.

If, in each slot, at most M attempts (M < N) are made to find a busy user, then the overhead is *only M minislots long and the channel* capacity is

$$C = \frac{1}{1 + Ma} \tag{4.1}$$

instead of C = 1/1 + Na (Eq. (3.3)). Under heavy traffic conditions, the probability of finding no busy user after M attempts in a given slot is very small, and we get a performance (with M < N) better than that obtained in Chapter 3 since we can achieve a higher throughput (see Eq. (4.1)). However, at light traffic the probability of choosing M idle users is high. Thus, if after M unsuccessful attempts (M idle users), we choose to wait until the next slot (we waste a slot), we penalize any busy users among the (N - M) remaining ones (with the schemes of Chapter 3 one of them could have transmitted a packet in the current slot) and therefore we increase the delay performance under light traffic conditions.

In order to avoid this high delay performance at light traffic, we introduce the following technique: The N users are split in k groups ($M + k < N$). After M unsuccessful attempts, one among the k groups is chosen. All busy users* in this group will transmit their packet in the current slot. As in slotted ALOHA, if more than one user (among the N/k users of this group) is busy, there is a collision, and those (busy) users must retransmit their packet in a later slot. If only one user is busy, his packet is successfully transmitted. Finally, if all users of the group are idle, then we choose another disjoint group of users who operate as above, etc.

With this technique, we obtain a much better throughput-delay performance than that obtained in Chapter 3. In particular, the capacity of the channel is

$$C = \frac{1}{1 + (M + k)a} \qquad (4.2)$$

However, the improved performance is still lower than that of the existing multiple access schemes (e.g., CSMA and Polling) when N is large ($N > 100$).

Since AP, RR and RO are equivalent in terms of delay-throughput performance (same average delay (Eq. (3.27)) and their normalized delay variances are close to each other (Figure 3.11)), we choose, without (much) loss of generality to study the Random Order mode (RO) throughout this chapter. The modified protocol is described in Section 4.1. The same technique could have been applied to AP or RR. In Section 4.2 we present the delay-throughput performance obtained by simulation for

---

*users who have at least one packet ready for transmission various values of M and k (number of groups of users contending for the channel in a slotted ALOHA fashion).

4.1 Protocol

The protocol considered below is not conflict-free, since in a given slot, after M unsuccessful attempts, users contend for the channel in a slotted ALOHA fashion.

Slot Configuration

Each slot consists of three parts (Figure 4.1):

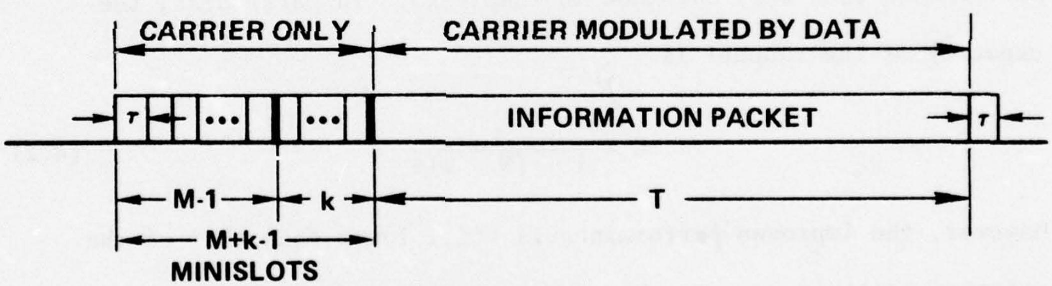


Figure 4.1. Slot Configuration.

1) an overhead of M + k - 1 minislots*, followed by
2) the packet transmission time of length T, followed by
3) one minislot (propagation time).

All users are synchronized and may start transmission of the carrier only at the beginning of a minislot, and all users may start the

---

*as defined in Chapter 3.

transmission of the (data) packet only at the beginning of the second part of the slot, i.e., $M + k - 1$ minislots after the beginning of a slot.

As in Section 3.2.4, in a given slot the N users are randomly ordered in sequence. If the resulting random ordered sequence in the current slot is $u_1, u_2, \ldots, u_M, \ldots, u_N$, then user u_1 has highest priority, and the lowest order user is u_N.

If user u_1 has a packet to transmit, he starts sending the carrier (with no modulation) at the beginning of the slot. If u_1 is idle he remains quiet, and after one minislot user u_2 knows that u_1 is idle (carrier absent). If u_2 has a packet ready for transmission, he starts sending the carrier at the beginning of the second minislot, etc.

Until now, the technique is identical in all ways to the Random Order (RO) multiple access technique studied in Chapter 3; in particular all system assumptions and operational features defined in Chapter 2 and in Sections 3.1 and 3.2.4 hold. The protocol described below differs from RO as follows:

If after M attempts, we do not find a busy user (i.e., $u_1, \ldots, u_M$ are idle) then a group of users is selected randomly among k disjoint groups, the union of which is the set of the N users; e.g., each group contains $\left\lfloor \frac{N}{k} \right\rfloor$ users* except the last one which contains $N - (k - 1) \left\lfloor \frac{N}{k} \right\rfloor$ users. All busy users of this selected group know that the channel is idle (carrier absent for $(M - 1)$ minislots), and transmit their packet in the given slot. (If more than one user transmits,

*where $\left\lfloor \frac{N}{k} \right\rfloor$ represents the integer part of $\frac{N}{k}$.

they collide and must retransmit their packet the next time they are selected.) If none of these $\left\lfloor \frac{N}{k} \right\rfloor$ users has a packet ready for transmission, another group is chosen, the users of which know that the channel is idle (carrier absent for M - 1 + 1 = M minislots) and may transmit in this slot, etc. If after k successive attempts (i.e., a total of M + k attempts in this slot), the carrier is absent, then all users are surely idle; the slot is unused and all users wait until the next slot.

Let us denote by $g_1, g_2, \ldots, g_k$ the k groups into which the N users are split, e.g.,

$$g_1 = \{u_1, \ldots, u_{\lfloor N/k \rfloor}\} \ldots g_k = \{u_{(k-1)\lfloor N/k \rfloor + i}, \ldots, u_N\}$$

In each slot, the k groups are randomly ordered as were the N users (see Section 3.2.4), i.e., each user generates the same sequence of pseudo-random numbers between 1 and k, and this invokes the same FIFO queue of random numbers; we now have at each user two random numbers queues: one for the users as in Section 3.2.4, called the "users-queue" and one for the k groups, called the "groups-queue." The head (i) of the groups-queue represents the group g_i to whom the slot is assigned (if the M first users selected are idle). If all users of group g_i are idle, then i is deleted from the groups-queue, and the new head j represents the group g_j of users to whom the slot is assigned if the first M users and all users of group g_i are idle, etc.

Protocol:

(1) The highest order user* in the current slot need not sense the channel and synchronizes his packet's transmission, if any, as follows:

(i) At the beginning of the slot he begins transmission of the carrier.

(ii) $(M + k - 1)$ minislots later he transmits the packet.

(2) The i^{th} user in sequence† $(1 \leq i \leq M)$ in the current slot, senses the channel for the first $(i - 1)$ minislots.

(i) If no carrier is detected after $(i - 1)$ minislots, he synchronizes his packet's transmission, if any, as follows:

a) At the beginning of the i^{th} minislot he begins transmission of the carrier,

b) $(M + k - i)$ minislots later he transmits the packet.

(ii) Otherwise (users idle or carrier detected earlier) he waits for the next slot and then operates as above.

(3) All users of the j^{th} group in sequence§ $(1 \leq j \leq k)$ in the current slot, sense the channel for the first $(M - 1 + j)$ minislots.

(i) If no carrier is detected after $(M - 1 + j)$ minislots, all busy users of the j^{th} group synchronize their

---

*represented by the head of the (FIFO) users-queue

†The i^{th} user in sequence is represented by the i^{th} number in the (FIFO) users-queue, starting from the head of the queue.

§The j^{th} group in sequence is represented by the j^{th} number in the FIFO groups-queue, starting from the head of the queue.

packet's transmission as follows:

a) At the beginning of the $(M + j)^{th}$ minislot*, they begin transmission of the carrier,

b) $(k - j)$ minislots later* they transmit their packet.

(ii) All idle users, as well as all busy users if the carrier was detected earlier, wait for the next slot and then operate as above†.

(4) Before the end of the current slot, all users update their priority, i.e., take the following actions:

(i) Generate a pseudo-random permutation of 1, 2, ..., N; The resulting sequence in the FIFO users-queue (the same for all users) gives the priority order of the M users for the next slot.

(ii) Generate a pseudo-random permutation of 1, 2, ..., k; The resulting sequence of numbers in the FIFO groups-queue (the same for all users) gives the priority order of the k groups for the next slot.

Before studying the throughput-delay performance obtained under this protocol, let us evaluate the channel capacity.

Observe first, that despite the possible collisions among users of a same group (when the first M selected users are idle), the channel never drifts into saturation, and a maximum throughput $S_{max} = 1$ packet

---

* The k^{th} group's users transmit their packet right away (if no carrier is detected after $(M + k - 1)$ minislots).

† Among the users of the j^{th} group in sequence, some (idle) users may have been selected in Step 2. If meanwhile a packet was generated at those users, they compete with other users of the j^{th} group.

per slot is achievable (with infinite delays). Indeed, when the input rate S is close to one packet per slot, the probability that the first M selected users are idle is close to zero; a busy user will eventually be selected within the M attempts. This user will successfully transmit his packet.

Then the channel capacity is

$$C = \frac{S_{max}}{1 + (M + k)a} = \frac{1}{1 + (M + k)a}$$

since in each slot, an overhead of (M + k) minislots is wasted for packet transmission. By setting k = 0, we obtain the channel capacity (Eq. (4.1)) of the protocol mentioned at the beginning of this chapter. With this protocol, a given slot is wasted if after M attempts no busy user was found (no ALOHA contention).

4.2 Throughput-Delay Performance

In the introduction of this chapter, it was pointed out that by reducing the number of attempts M to find a busy user in a given slot, the overhead in each slot is reduced and therefore the channel capacity is increased (Eq. 4.1). However, it was mentioned that one consequence of this overhead reduction was an increase of the packet delay under light traffic conditions. The smaller M is, the larger is the channel capacity, but the larger is the packet delay under light traffic. Let us evaluate the packet delay D at any user when the total input rate S is close to zero (for k = 0). We easily show that

$$D = \frac{N}{M} + \frac{1}{2} \quad \text{(slots)} \qquad (4.3)$$

when $S = 0$, assuming as in Chapter 3 that all user packet arrival processes are statistically identical (Poisson with intensity λ/N).

To obtain Eq. (4.3) let us evaluate (when the input rate λ is very small) the probability P_L that a slot is wasted, given there is at least one busy user. We have

$$P_L = \sum_{i=1}^{N-M} P\{\text{i users are busy and the M selected users are chosen out of the } (N-i) \text{ idle users}\}/P\{\text{at least one user is busy}\}$$

$$P_L = \frac{\sum_{i=1}^{N-M} \times P\left\{\begin{matrix}\text{the M users are chosen out of the } (N-i) \\ \text{idle users / i out of N users are busy}\end{matrix}\right\} \times P_i}{\sum_{i=1}^{N} P_i}$$

where $P_i \triangleq P\{\text{i out of N users are busy}\}$

$$P_L = \sum_{i=1}^{N-M} \frac{N-i}{N} \times \frac{N-i-1}{N-1} \times \cdots \times \frac{N-i-M+1}{N-M+1} P_i \Big/ \sum_{i=1}^{N} P_i$$

since the users are ordered randomly. Then

$$P_L = \sum_{i=1}^{N-M} \frac{\binom{N-M}{i}}{\binom{N}{i}} P_i \Big/ \sum_{i=1}^{N} P_i$$

Assuming that at very light traffic ($\lambda \cong 0$), the events {user j busy} $j = 1, \ldots, N$ are independent of each other, we obtain the following expression for P_i

$$P_i = \binom{N}{i} \rho^i (1-\rho)^{N-i}$$

where ρ is the probability that any user is busy.

The smaller λ is, the better is the previous assumption. Then,

$$P_L = \sum_{i=1}^{N-M} \binom{N-M}{i} \rho^i (1-\rho)^{N-i} \Big/ 1 - (1-\rho)^N$$

$$P_L = (1-\rho)^M \sum_{i=1}^{N-M} \binom{N-M}{i} \rho^i (1-\rho)^{N-M-i} \Big/ 1 - (1-\rho)^N$$

$$P_L = \frac{(1-\rho)^M - (1-\rho)^N}{1-(1-\rho)^N}$$

Since λ is close to zero, so is ρ and we can write

$$P_L = \frac{1 - M\rho - (1 - N\rho) + 0(\rho)}{N\rho + 0(\rho)}$$

$$P_L = 1 - \frac{M}{N} + 0(\rho) \qquad (4.4)$$

where $0(\rho)$ is such that

$$\lim_{\rho \to 0} \frac{0(\rho)}{\rho} = 0$$

Therefore, in the limit, when $\lambda = \rho = 0$, the probability of loss P_L is

$$P_L = 1 - \frac{M}{N} \qquad (4.5)$$

Eq. (4.5) may be directly obtained by observing that the probability that a packet which upon arrival finds the system empty (all users idle except one user who has one packet ready for transmission) will not be transmitted in the next slot, with probability

$$P_L = \frac{N-1}{N} \times \frac{N-2}{N-1} \times \cdots \times \frac{N-M}{N-M+1}$$

where the i^{th} factor is the probability that at the i^{th} attempt

($1 \leq i \leq M$), an idle user is selected. Therefore

$$P_L = 1 - \frac{M}{N}$$

Knowing P_L, we easily get the expected delay D when $\lambda \cong 0$. The latter is composed of two parts: The first part is 1/2 slot, which is the average time elapsing from the packet arrival instant until the beginning of the next slot; the second part, denoted by $\bar{x}$, is the expected value of the number of slots $\tilde{x}$ it takes for the user j (where the packet was generated) to be selected. Recall that since $\lambda \cong 0$, the packet generated at user j will eventually be transmitted before any new packet is generated. Then

$$P\{\tilde{x} = k\} = P_L^{k-1}(1 - P_L) \qquad k = 1, 2, 3, \ldots$$

and

$$E\{x\} = \bar{x} = \frac{1}{1 - P_L} = \frac{N}{M}$$

then

$$D = \frac{1}{2} + \bar{x} = \frac{1}{2} + \frac{N}{M}$$

In conclusion, at very light traffic, the closer M is to N, the lower is the delay. But to increase the channel capacity, we require a small value of M, and thus we expect a large value of D. This is why the protocol with ALOHA contention described in the previous section was introduced. With this protocol, even with a very small value of M, we now have, at $\lambda = 0$:

$$D = 1 + \frac{1}{2} = \frac{3}{2} \qquad (4.6)$$

Indeed, with this protocol, when $\lambda = 0$, a packet finding an empty system upon arrival will eventually be transmitted in the slot following its arrival. With increasing values of the input rate λ, how is the average delay expected to behave with the new protocol?

If M is small, at larger values of the input rate (e.g., N = 50, M = 5, S = 0.5 packets/slot), the probability is intuitively high that in a given slot, no busy user is found after M attempts and that $\left\lfloor \frac{N}{k} \right\rfloor$ users will contend for the channel (after M unsuccessful attempts, a first group of $\left\lfloor \frac{N}{k} \right\rfloor$ users is allowed to transmit in this slot. If none of these $\left\lfloor \frac{N}{k} \right\rfloor$ users is busy, another group will contend for the channel in this slot, etc.).

In order to transmit a packet successfully in this slot, then, one expects no collision among users of the same group. The smaller the group size $\left\lfloor \frac{N}{k} \right\rfloor$ is, the less likely is the collision, and therefore a value of k not too small (e.g., k = 20 when N = 50) should be chosen. But if k is too large, the increase in the channel capacity with this protocol is not significant (see Eq. (4.2)): <u>The smaller (M + k) is, the higher is the channel capacity.</u>

No analytical optimal value was found for M and k (when N is given).

However, by simulating the system, we may choose for a given N, the values of M and k such that, at all values of the input rate, we significantly improve the expected delay. The simulation verifies the expected following trends:

(a) <u>M-effect</u>: For a given k, <u>smaller values of M provide higher delays</u> at all values of the input rate not too close to one packet/slot, but provide a larger maximum achievable throughput. When M is very small the delay is not significantly decreased, even with larger values of k.

(b) k-effect: Small values of k provide low delays only at small input rate values, $\lambda < \lambda_0$. For a given M, the larger k is, the higher is the input rate range ($0 < \lambda < \lambda_0$) for which the delay is significantly low.

The tradeoff will be to choose between a small value of (M + k) (high channel capacity but high delays at most values of λ) and values of M and k large enough (low delays at all values of the throughput not too close to one packet/slot, but low achievable throughput).

In Figures 4.2 and 4.3, the expected packet delay, obtained by simulation, is plotted versus the throughput for N = 10 (Figure 4.2) and N = 50 (Figure 4.3), for various values of M and k. Delay and throughput are normalized with respect to the packet transmission time T.

For small values of N (N = 10), the protocol RO, defined in Chapter 3, is close to the optimum. For the sake of clarity, in Figure 4.2 we did not plot the delay contour for (M = 7, k = 1) which is very slightly below the delay contour of RO (M = 10). With M = 5, k = 1, we obtain a channel capacity of .95 (RO's capacity is .92), and the delay is very close to the delay provided by RO when the throughput is less than .85. For very small values of M (M = 1), even if k is large, (k = 5), although we obtain a large capacity (.95) for throughputs larger than .75, the delay is higher than that provided by TDMA. When k = 0, the delay is very large for most values of throughput, e.g., for M = 3 and k = 0, at zero throughput the normalized delay (Eq. (4.3)) is but provide a larger maximum achievable throughput. When M is very small the delay is not significantly decreased, even with larger values of k.

(b) k-effect: Small values of k provide low delays only at small input rate values, $\lambda < \lambda_0$. For a given M, the larger k is, the higher is the input rate range $(0 < \lambda < \lambda_0)$ for which the delay is significantly low.

The tradeoff will be to choose between a small value of (M + k) (high channel capacity but high delays at most values of λ) and values of M and k large enough (low delays at all values of the throughput not too close to one packet/slot, but low achievable throughput).

In Figures 4.2 and 4.3, the expected packet delay, obtained by simulation, is plotted versus the throughput for N = 10 (Figure 4.2) and N = 50 (Figure 4.3), for various values of M and k. Delay and throughput are normalized with respect to the packet transmission time T.

For small values of N (N = 10), the protocol RO, defined in Chapter 3, is close to the optimum. For the sake of clarity, in Figure 4.2 we did not plot the delay contour for (M = 7, k = 1) which is very slightly below the delay contour of RO (M = 10). With M = 5, k = 1, we obtain a channel capacity of .95 (RO's capacity is .92), and the delay is very close to the delay provided by RO when the throughput is less than .85. For very small values of M (M = 1), even if k is large, (k = 5), although we obtain a large capacity (.95) for throughputs larger than .75, the delay is higher than that provided by TDMA. When k = 0, the delay is very large for most values of throughput, e.g., for M = 3 and k = 0, at zero throughput the normalized delay (Eq. (4.3)) is

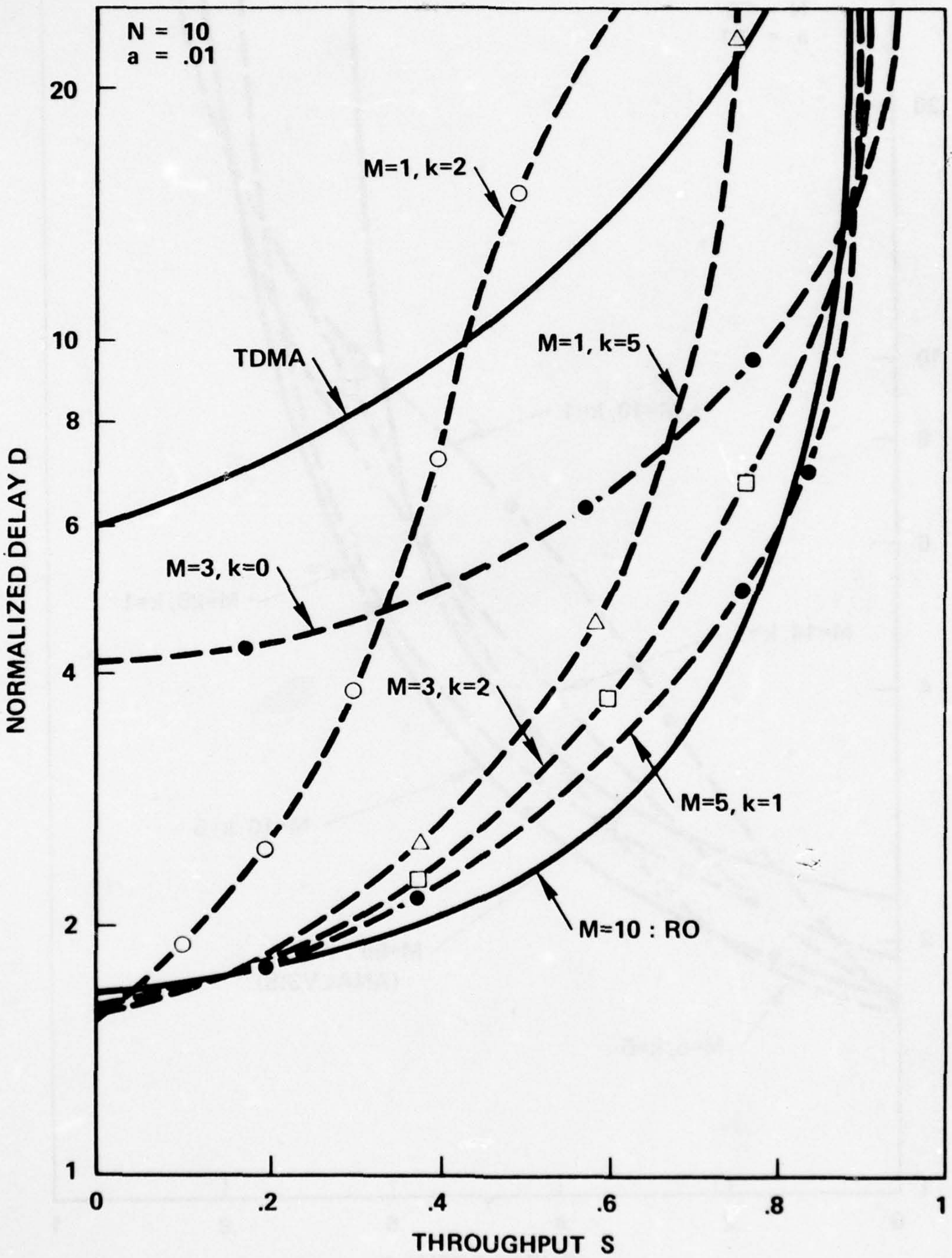


Figure 4.2. New Scheme: D versus S. (N = 10).

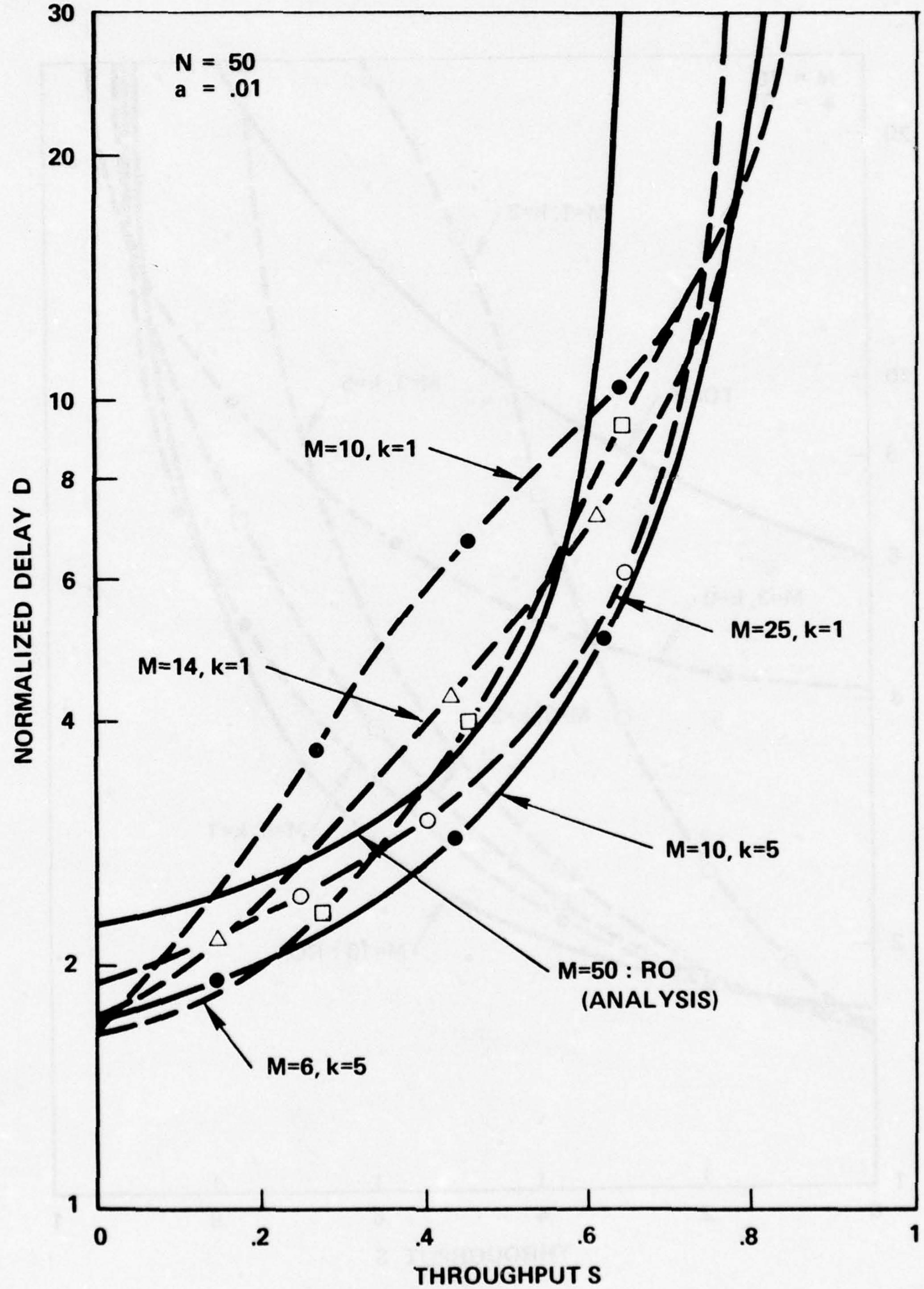


Figure 4.3. New Scheme: D versus S. (N = 50).

$3.9\left(=\left(\frac{N}{M}+\frac{1}{2}\right)(1+Ma)\right)$. In conclusion, with the new protocol we do not get a significant improvement over RO for small values of N ($N \leq 10$).

In Figure 4.3, the expected delay D is plotted versus the throughput S for a value of N = 50. The improvement obtained with the new protocol is important: First, we note the very good delays obtained at low traffic for any value of M and k ($\neq 0$); e.g., if M = 10 and k = 1, then D = 1.65 at S = 0 (with k = 0, D = 6 at S = 0). Second, when M is small (e.g., M = 10, k = 1) we note the large delays (k-effect) for intermediate values of S ($.2 \leq S \leq .8$) leading to a choice of k large (e.g., M = 10, k = 5). Third, we note that when M is very small (e.g., M = 6), even with a value of k = 5, D increases fast when S increases (M-effect). Finally, keeping (M + k) constant, we observe the improvement in delay obtained by decreasing M and increasing k (e.g., i) M = 10, k = 1 and M = 6, k = 5; ii) M = 14, k = 1 and M = 10, k = 5) for the same channel capacity. The values M = 10, k = 5 provide a performance close to the optimum at all values of throughput and a channel capacity of .88 instead of .66 obtained with RO (M = 50).

In Figure 4.4 the delay-throughput performance is compared between RO (M = 50), our new protocol with M = 10 and k = 5, Polling, CSMA and CS SRMA. As previously mentioned (Figure 4.3), the delay is lower with the new scheme (M = 10; k = 5) and the capacity is much higher than with RO (M = 50). At low traffic ($S \leq .4$), Polling and the new scheme (M = 10, k = 5) provide the same delay, but with Polling we may achieve a throughput of 1, while the maximum achievable throughput with the new protocol is .88. We plotted the CSMA performance as obtained in [TOBA 76B] and the CS SRMA performance predicted by the

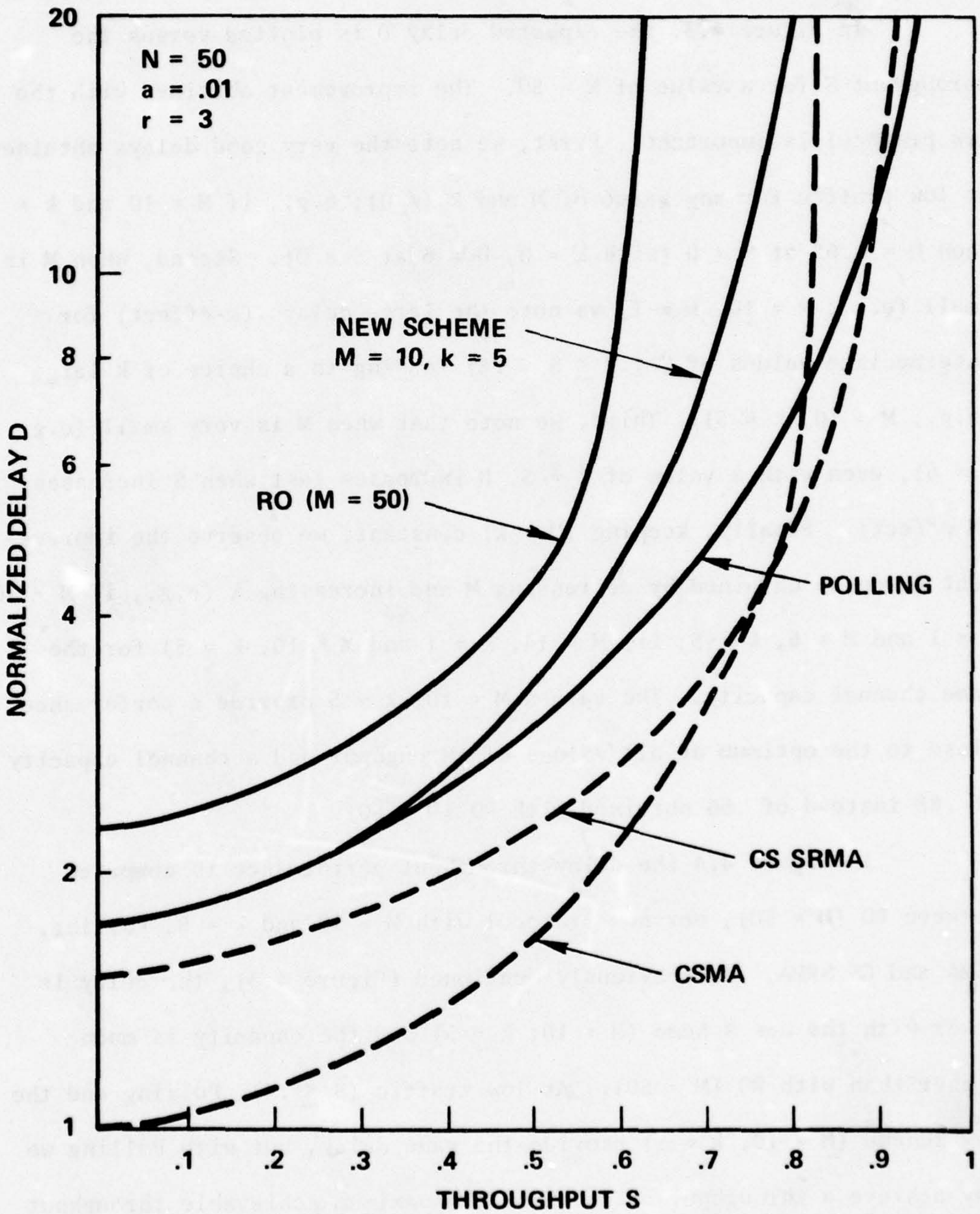


Figure 4.4. D vs S : Comparison to Existing Techniques

infinite population model [TOBA 76A] (unstable channel performance, upper bound for the stable channel performance: see Chapter 1). Both CSMA and CS SRMA provide a lower channel capacity than our new protocol, but perform much better over a large traffic range.

Until now, the ratio a of the maximum propagation time τ to the packet transmission time T has been chosen

$$a = .01$$

as a typical value for ground radio communications. We already showed in Chapter 3 (Figure 3.6) how the channel capacity of AP, RR and RO decreases when a increases under CSMA channel capacity and ALOHA channel capacity.

In Figure 4.5, the channel capacity C is plotted versus a, with N = 50, for Polling (C = 1 for all a); CSMA [TOBA 74]; RO ($C = \frac{1}{1 + Na}$) our new protocol ($M = 10$, $k = 5$, $C = \frac{1}{1 + (M + k)a}$); and slotted ALOHA ($C = 1/e$).

For $a < .018$, the new scheme provides a better channel capacity than CSMA (instead of $a < .001$ with RO). For $a < .11$, the new scheme's channel capacity is larger than $1/e$ (instead of $a < .035$ with RO).

In conclusion, the protocol introduced in this chapter as an extension of RO has been shown to perform much better than RO, and to provide, under light traffic conditions, the same performance as Polling, and a better performance than CSMA under heavy traffic conditions when N is not too large ($N \leq 100$).

When the number of users N is very large ($N > 100$), we can still find values of M and k for which the new protocol

a) performs much better than RO, and

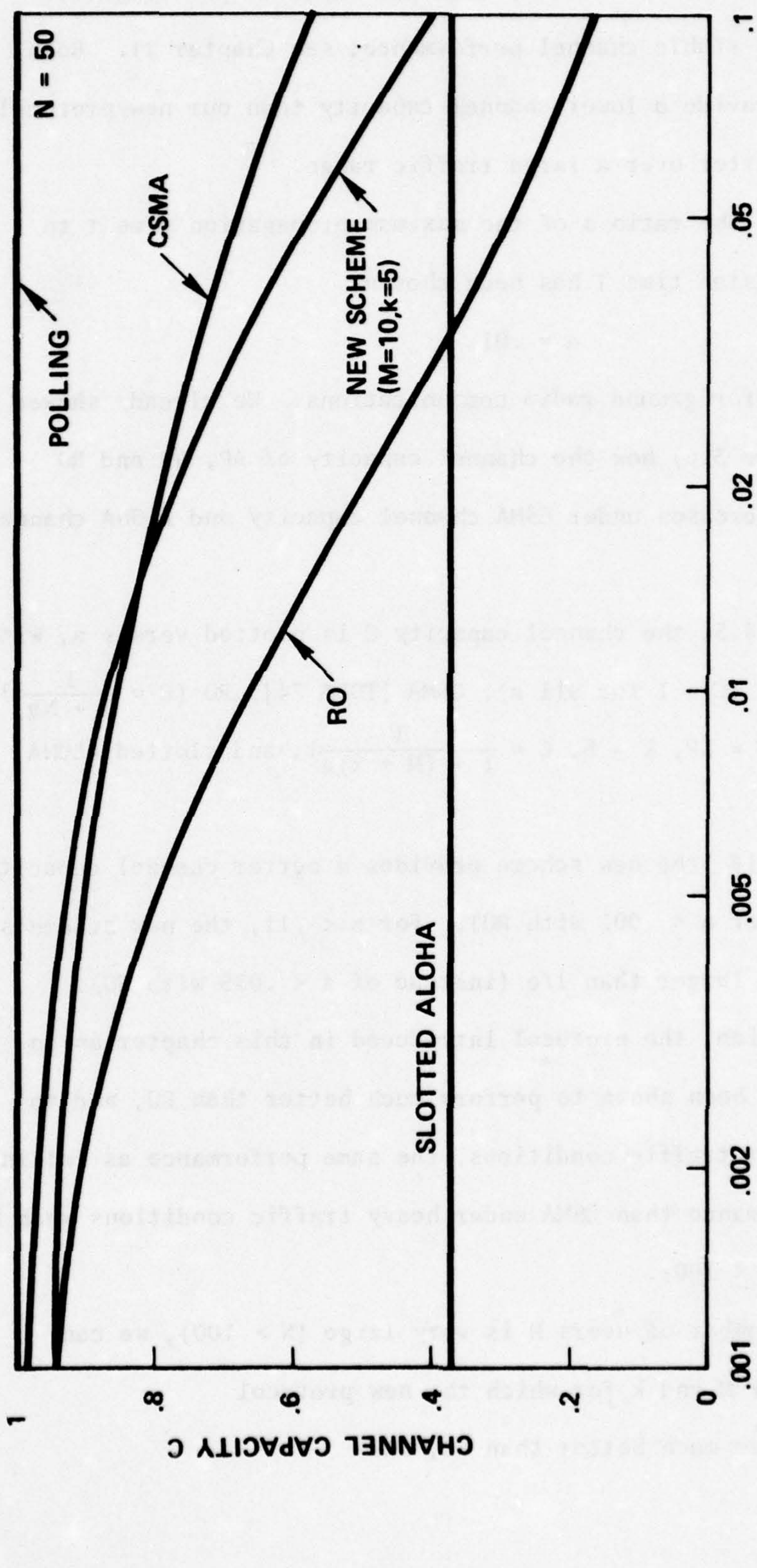


Figure 4.5. Channel Capacity vs a.

b) provides the same delay performance as Polling at low traffic.

However, the throughput-delay performance for such values of M and k will be lower than that obtained with CSMA.

In the next chapter, we introduce another multiple access scheme which provides a better throughput-delay performance for all values of N and a than Polling does, and which performs better than the random access techniques (CSMA, CS SRMA) under heavy traffic conditions, since this new technique, like Polling, allows a maximum achievable throughput of one.

CHAPTER 5

MINI-SLOTTED ALTERNATING PRIORITIES

The major limitation of the schemes studied in Chapter 3 has been shown to be the large overhead (increasing proportionally with the number of users N), rendering those schemes suitable only for a population of users composed of a rather small number of users ($N \leq 20$), and providing a performance far below that obtained with CS SRMA, CSMA or Polling when N is large ($N \geq 50$) and a is not too small ($a = .01$). In order to decrease this overhead, the protocols of Chapter 3 are modified in Chapter 4. This results in a significant improvement of the performance which, however, stays below that obtained with existing techniques for large values of N ($N \geq 50$). Below we introduce and analyze a conflict-free scheme referred to as Mini-Slotted Alternating Priorities (MSAP), which also allows buffering capabilities and also does not require control of a central station. We shall easily solve for the average packet delay under MSAF and show that MSAP performs better than CSMA (and CS SRMA) under heavy traffic conditions, and performs better than Polling for all traffic levels and all numbers of users.

5.1 Protocol

The major difference with the schemes AP, RR, RO and HOL studied in Chapter 3 comes from the slot size which is now taken as equal to the maximum propagation delay τ, i.e., what we called a minislot in Chapter 3 is now referred to as a slot. As with the former schemes, we use the carrier sense capability of each user. However we now try to minimize the channel time lost for packet transmission, i.e., the overhead due to carrier sensing in order to "steal" a slot assigned to an idle user. The protocol obeys the Alternating Priorities rule as follows:

(1) Assign the channel to that user (say i) who transmitted the last packet, if possible; otherwise,

(2) Assign the channel to the next user in sequence*.

By carrier sensing, at most one slot later, all users detect the end of transmission of user i (absence of carrier†); in particular, so does the next user in sequence (user (i mod N + 1)). Then

i) either: User (i mod N + 1) starts transmission of a packet; in this case, one slot after the beginning of transmission, all others detect the carrier. They wait until the end of this packet's transmission and then operate as above.

ii) or: User (i mod N + 1) is idle; in this case, one slot later, all the other users do not detect the carrier, they then know that it is the turn of the next user in sequence, i.e., user (i mod N + 2), and operate as above.

When all users are idle, the "turn" keeps changing at each slot until it is the turn of a non-idle user.

---

* Users grasp the channel according to a fixed order, say 1, 2, ..., N, without loss of generality.

† The carrier detection time is assumed to be negligible (Assumption 8). All users know whose turn has come at most one slot after the end of transmission of a packet.

In Figure 5.1 we consider an example with four users. Two siots after the end of user 3's transmission, user 4 being idle, user 1 starts transmission. He transmits three packets, followed by user 2 and then user 3.

Two remarks are noteworthy:

a) We could have chosen the Random Order Rule or the Round Robin rule. The latter is suitable for unbalanced traffic; i.e., when some users have a smaller input rate than others. Then the Round Robin rule provides to users with small input rate a more frequent access to the channel than the Alternating Priorities rule does. However, the Alternating Priorities rule is chosen here, in order to minimize the "changeover" time between users. This changeover time, which is lost for packet transmission, is shorter with Alternating Priorities than it is with Round Robin (or Random Order)*. The maximum channel utilization is obtained with the Alternating Priorities rule which allows the system to achieve full utilization of the channel. When one queue is saturated and keeps the channel for its own use, there is no changeover and therefore the throughput is S = 1 packet/packet transmission time. Thus, the capacity of MSAP is equal to 1.

b) In Polling, the channel is assigned to the users according to the same rule. The only difference is that the polling time or changeover time between the two users is equal to the polling message transmission time (of length b ($\geq$ 1) slots) plus twice the propagation

* This overhead (one slot per switchover from one user to another) is very small, compared to that incurred with the schemes studied in Chapter 3 (for which N minislots are lost at each packet transmission time).

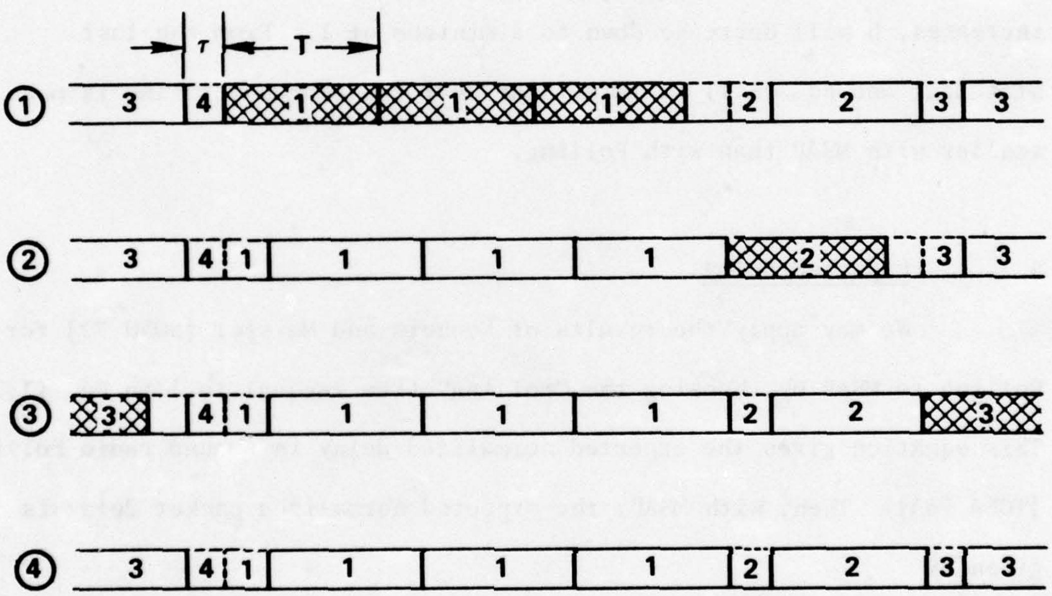


Figure 5.1. Minislotted Alternating Priorities (MSAP) Example of 4 Users (Cross-Hatching Indicates a Transmission).

time between users and station.

If we denote this changeover time by r, we then have for

$$\begin{aligned} &\text{a) } \underline{\text{Polling}} \quad r = b + 2 \\ &\text{b) } \underline{\text{MSAP}} \quad r = 1 \end{aligned} \tag{5.1}$$

Since the polling message contains the identification of the user which is polled, b will increase with N (in particular, it must grow in proportion to log N). Also, b depends on the parameter a. If a increases, b will decrease down to a minimum of 1. From the last statement and Eq. (5.1) it is evident that the changeover time is much smaller with MSAP than with Polling.

5.2 Expected Delay

We may apply the results of Konheim and Meister [KONH 72] for Polling to MSAP by choosing the "polling" time r equal to 1 in Eq. (1.3). This equation gives the expected normalized delay in ground radio Polling [TOBA 76A]. Then, with MSAP, the expected normalized packet delay is given by

$$D = 1 + \frac{S}{2(1 - S)} + \frac{a}{2}\left(1 - \frac{S}{N}\right)\left(1 + \frac{N}{1 - S}\right) \tag{5.2}$$

Therefore, from Eqs. (1.3), (5.1) and (5.2), with Polling, a packet incurs an average delay which exceeds the one he would incur with MSAP by an amount Δ, namely,

$$\Delta = \frac{a}{2}\left(1 - \frac{S}{N}\right)\left(\frac{N(b + 1)}{1 - S}\right) \tag{5.3}$$

When N is large (N > 10),

$$\Delta \cong \frac{Na(b + 1)}{2(1 - S)}$$

Consider, for example, 1000 bit packets transmitted over a channel operating at a speed of 100 kilobits per second. The transmission time of a packet is then T = 10 mseconds. If the maximum distance between the source and the destination is 20 miles, then τ = 108 μseconds. Thus

$$a = \frac{\tau}{T} \cong .01$$

To poll N = 50 users, one slot (10 bits) is enough: b = 1 and $\Delta = \frac{1}{2(1 - S)}$. To poll N = 1000 users, two slots are necessary*, and then: b = 2 and $\Delta = \frac{30}{2(1 - S)}$. As N increases, MSAP by far exceeds the performance of Polling (Figure 5.2).

Let us now compare MSAP to CSMA and CS SRMA. In Figure 5.2 we plot the CSMA performance as obtained in [TOBA 76B], for N = 50 and N = 1000. In the same figure, we also plot the CS SRMA performance predicted by the infinite population model (unstable channel) [TOBA 76A]. As mentioned before (Chapters 1 and 3), it is likely that this performance is an upper bound for the steady state performance (stable channel) in the case of N = 50 users. However when N = 1000, the need to increase retransmission delays in order to have a stable channel may lead to a performance worse than the one predicted by Figure 5.2.

In comparing MSAP to CSMA and CS SRMA, we note from Figure 5.2 that, at light traffic, the larger is N, the more do CSMA (and SRMA) exceed the performance of MSAP; but under heavy traffic conditions, MSAP always performs better than CSMA. For N = 50, the delay with MSAP is better than that obtained with CSMA, for a throughput equal to 0.6 or higher.

---

*since we need 10 bits for identifying the 1000 users and at least 1 bit for error detection purposes.

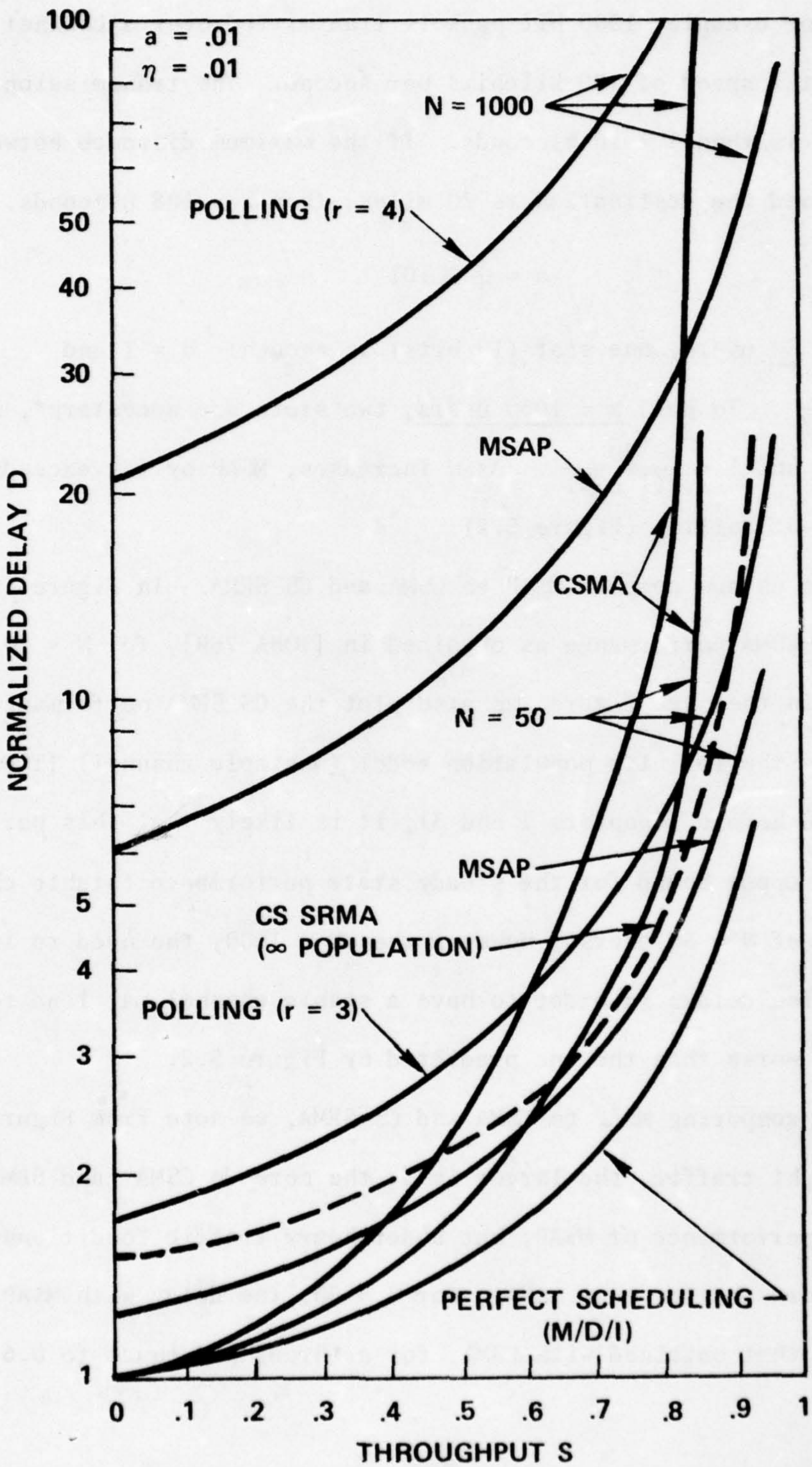


Figure 5.2. D vs S : Comparison to Existing Techniques

CHAPTER 6

MIXED MULTIPLE ACCESS MODES FOR ONE LARGE USER AND A POPULATION OF SMALL USERS

With a large population of bursty users (referred to as <u>small users</u>), we already pointed out in Section 1.2 that the use of slotted ALOHA provides flexibility, simplicity (of implementation) and a more efficient channel utilization with much lower delays at light traffic than does TDMA or FDMA. However the maximum achievable throughput with slotted ALOHA is 1/e. This suggests that we include a <u>large user</u> on the same channel in order to increase the channel capacity. We assume that this large user does not require a short response time (packet delay) as does the background of small users. As an example, we might consider a background of bursty interactive users, together with a user transmitting a large amount of data (file user). Several papers have suggested the use of a radio channel in an environment including small users and a large user. L. Roberts did so in an unpublished note; Abramson [ABRA 73] provided a model where all users (including the large user) use the channel in a slotted ALOHA fashion and determined the channel capacity; Lam [LAM 74] simulated the packet delay at the large user. Gitman et al. [GITM 74] studied such a model to determine whether one should share the radio channel by the large (or possibly by a finite number of large users) and the small users or whether one should split the channel so that one part is used by the large user and the other part is used by the small users.

Binder [BIND 75A] introduced and simulated a technique for regulating file traffic (large user) in a slotted or unslotted ALOHA channel when interactive users' (small users) traffic is also present.

Throughout this chapter, we assume the large user to be within range and in line of sight of all small users. This large user will sense the carrier and will not transmit a packet unless all small users are quiet. We give priority to the small users and since they are controlling the entire bandwidth, the small users perform better than if they were dedicated only a part of the available bandwidth. However, since the large user has lower priority, he may incur higher delays and achieve less throughput than if he was dedicated a part of the channel (of smaller bandwidth). We study two cases which differ with respect to the nature of the background of small users:

a) A large population of small users who contend for the channel in a slotted ALOHA fashion.

In Section 6.1, the Mixed ALOHA Carrier Sense (MACS) mode is analyzed. The MACS performance is shown to be better than the performance of the "large user model" mentioned above [ABRA 73] and also better than the performance obtained when the available bandwidth is optimally split in two parts, one for the large user, the other one for the small users. With MACS, a good channel utilization is obtained (greater than 2/e for all small users' input rates). The large user may achieve a significant throughput (> 1/e). In addition, by providing the entire available bandwidth to all users with MACS, we obtain a throughput delay performance which is better for both the large user and the small users than that obtained when the large user and the background use separate channels of smaller bandwidth.

b) A small number of buffered users within range and in line of sight of each other sharing the channel according to the AP access mode studied in Chapter 3.

In Section 6.2, the Mixed Alternating Priorities Carrier Sense (MAPCS) mode is analyzed. This mode is conflict free*. The MAPCS performance is shown to be better than that predicted by the slotted ALOHA mode mentioned above [ABRA 73]. With MAPCS, a total channel utilization of $\frac{1}{1 + (N + 1)a}$ is obtained where N is the number of small users and a is the ratio of the maximum propagation time to packet transmission time. This channel utilization is fairly good when N is not too large ($N \leq 30$). The throughput-delay performance of the small users is better with MAPCS than when the channel is optimally split. At the large user, however, the higher the small users' traffic, the larger is the degradation of the delay-throughput performance compared to the "split channel" mode where the large user and the small users have dedicated separate channels.

Finally, in Section 6.3 we compare MACS and MAPCS. When the number of small users is not too large ($N \leq 30$), one may achieve a higher total channel throughput with the second scheme (MAPCS). When N increases, the first scheme (MACS) turns out to be better. The performance at the large user is shown to be better with the first scheme (MACS). In addition, MACS does not require that all small users "hear" each other while MAPCS does; however MAPCS allows a high throughput rate for the small users.

*In MACS, small users conflict (in a slotted ALOHA fashion).

6.1 Mixed ALOHA and Carrier Sense Multiple Access (MACS)

We consider a large user and a large number of small users, all competing for the same channel. The access to the channel is modeled as follows.

6.1.1 The Model

We include on the same single channel (of bandwidth W) an infinite population contending for the channel in a slotted ALOHA fashion (small users) and a buffered user (a large user) with an infinite buffer size and a Poisson arrival process (intensity λ_2)* independent of the small users' (Poisson) arrival process. The large user and small users transmit packets of fixed length b_m bits. The large user is in line of sight and within range of all small users. The slot configuration is given by Figure 6.1.

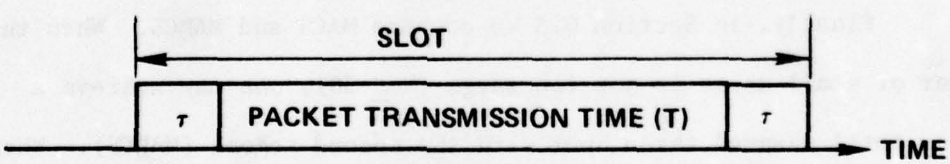


Figure 6.1. Slot Configuration.

*Packets generated at the large user queue up and are served on a first-come-first-served basis.

When a small user has a packet ready for transmission (newly generated or previously collided packet (see [KLEI 75A])), he transmits the carrier for the first τ seconds of the slot and then transmits the (information) packet. τ represents the maximum propagation time from a small user to the large user; τ is considered to be very small in comparison with a packet's transmission time, $T = \frac{b_m}{W}$; $a = \frac{\tau}{T}$ is chosen to be equal to .01 in the numerical calculation throughout this chapter.

When the large user has a packet ready for transmission, he senses the carrier at the beginning of the slot during τ seconds. If the carrier is not detected after τ seconds, the large user immediately transmits his packet. If the carrier is present (one or more small users are transmitting in the current slot) the large user stays quiet until the beginning of the next slot and then operates as above.

The last τ seconds of a slot account for the (maximum propagation) delay between the end of packet transmission and the end of packet reception.

The delay-throughput performance of the ALOHA population is not affected by the presence of the large user*; an analysis of the small users performance in a ground radio environment ($a \ll 1$) can be found in [KLEI 75B] which is similar to the satellite treatment by Kleinrock and Lam (see [LAM 74] and [KLEI 75A]). On the other hand, the large user's transmission is very sensitive to the ALOHA traffic: the higher the ALOHA traffic is, the lower the large user's throughput will be.

* except that the slot size is $(1 + 2a)T$, instead of T when there is no large user

6.1.2 Large Users's Throughput and Channel Capacity

In Chapter 1, we described the slotted ALOHA access mode which has been thoroughly studied in [LAM 74] and [KLEI 75A]. Three kinds of slots can be identified in a slotted ALOHA mode.

i) "successful" slots: slots in which one and only one user of the ALOHA population is transmitted. One packet is successfully transmitted.

ii) "conflicting" slots: more than one ALOHA users are transmitting a packet in this slot. Packets "collide" and must be retransmitted.

iii) "idle" slots: No ALOHA user is transmitting in this slot; each user either has no packet to transmit or has rescheduled the transmission of a previously collided packet for some later time.

In the first two cases (successful or conflicting slot) the channel is sensed busy by the large user. In the third case (idle slot) the channel is sensed idle. Thus, the large user may "steal" these idle slots for transmitting his own packets. By allowing this large user to steal the "idle" slots of an ALOHA population, we increase the total channel utilization; a packet is successfuly transmitted in a given slot if this slot is "successful" or "idle." If the proportion of idle slots is not too small, we may achieve a significant throughput, S_2, for the large user. We easily show below that

$$S_2 = e^{-G} \tag{6.1}$$

where G (average number of packets per slot) is the offered traffic rate (newly generated plus previously collided packets) of the ALOHA population, and where S_2 is the throughput in number of packets per slot for the large user. The ALOHA channel traffic (with mean G) is a random variable representing the total number of packets transmitted by all ALOHA users into a slot. It is assumed that (see [LAM 74])

(A.12) The ALOHA channel traffic is Poisson distributed.

The accuracy of Assumption A.12 has been examined in [LAM 74] through simulations and has been shown to be fairly good.

To evaluate S_2, let us call the "service time" of a packet at the large user (denoted by $\tilde{x}$), the number of slots elapsing between the end of the slot during which this packet was generated and the end of the (successful) packet transmission. From Assumption A.12, we have for a given slot:

$$P[\text{a packet is sent from the large user}] = P[\text{idle slot}] = e^{-G}$$

Then

$$P[\tilde{x} = k] = (1 - e^{-G})^{k-1} e^{-G} \qquad k \geq 1 \qquad (6.2)$$

Therefore, the "service time" as defined above is geometrically distributed with parameter e^{-G}, and

$$E(\tilde{x}) = \overline{x} = e^{G}$$

S_2 may be defined as the expected value of the random variable $\tilde{s}$, the number of packets generated at the large user that are allowed to get "through" the channel in a given slot. Then

$$S_2 = E(\tilde{s}) = 1 \times P\{\tilde{s} = 1\} + 0 \times P\{\tilde{s} = 0\} .$$

Observing that $P\{\tilde{s} = 1\} = P\{\text{idle slot}\} = e^{-G}$, we get Eq. (6.1)

$$S_2 = e^{-G}.$$

Note that under steady state conditions, this maximum throughput S_2 is achieved at the large user with infinite delays. For finite delays, the normalized input rate, S_2' at the large user must be such that

$$S_2' = \lambda_2(1 + 2a)T < S_2 = e^{-G}$$

for a given ALOHA traffic rate G.

If we denote by S_1 the total ALOHA channel throughput rate, we know that under steady state conditions (see, for example [KLEI 75A])

$$S_1 = Ge^{-G} \quad (6.3)$$

From Eqs. (6.2) and (6.3) we obtain the total achievable throughput of the channel:

$$S = S_1 + S_2 = (G + 1)e^{-G} \text{ (packets/slot)} \quad (6.4)$$

The maximum value of S is achieved (with infinite delays at the large user) when G = 0 (no traffic from the ALOHA background):

$$S_{max} = 1 \text{ packet/slot}$$

We then may obtain the channel capacity C. C is the maximum number of packets that can "get through" the system (see Chapter 2) during one packet transmission time T

$$C = \frac{S_{max}}{1 + 2a}$$

$$C = \frac{1}{1 + 2a} \quad (6.5)$$

Eq. (6.5) illustrates the fact that for each packet transmitted, a part ($a \ll 1$) of the channel is wasted (in each slot, the first τ seconds and the last τ seconds cannot be used for data transmission).

In Figure 6.2, we plot the total channel throughput S and the ALOHA throughput S_1 (normalized with respect to a slot) versus the ALOHA traffic rate G. S decreases with increasing values of G, from 1 ($G = 0$) to 0 ($G = \infty$). Because of the ALOHA population, the channel eventually drifts into saturation, i.e., the throughputs (S_1 and S) will go to zero, while the channel load will increase without any bound (see Chapter 1 and [KLEI 75A]). But by applying dynamic control policies [KLEI 75A], we can get a stable channel with a bounded ALOHA traffic. Therefore, the probability of an "idle slot" is greater than zero and we can achieve a throughput at the large user S_2 which is greater than zero.

Since the performance obtained for slotted ALOHA by applying control policies (stable channel) have been shown by Kleinrock and Lam [KLEI 75A] to be close to the quasi-stationary performance (unstable channel), clearly the same will be true when we include the large user as well as the ALOHA background which controls the channel. Throughout this chapter we will, as above (Eqs. (6.1) to (6.4)), use the ALOHA results achievable only for a finite period of time (unstable channel) as approximative results for a stable channel and assume $G \leq 1$.

When $G = 1$ (Figure 6.2) the ALOHA background achieves a maximum throughput

$$S_1 = 1/e$$

the channel throughput is $S = 2/e$ and the large user throughput is also

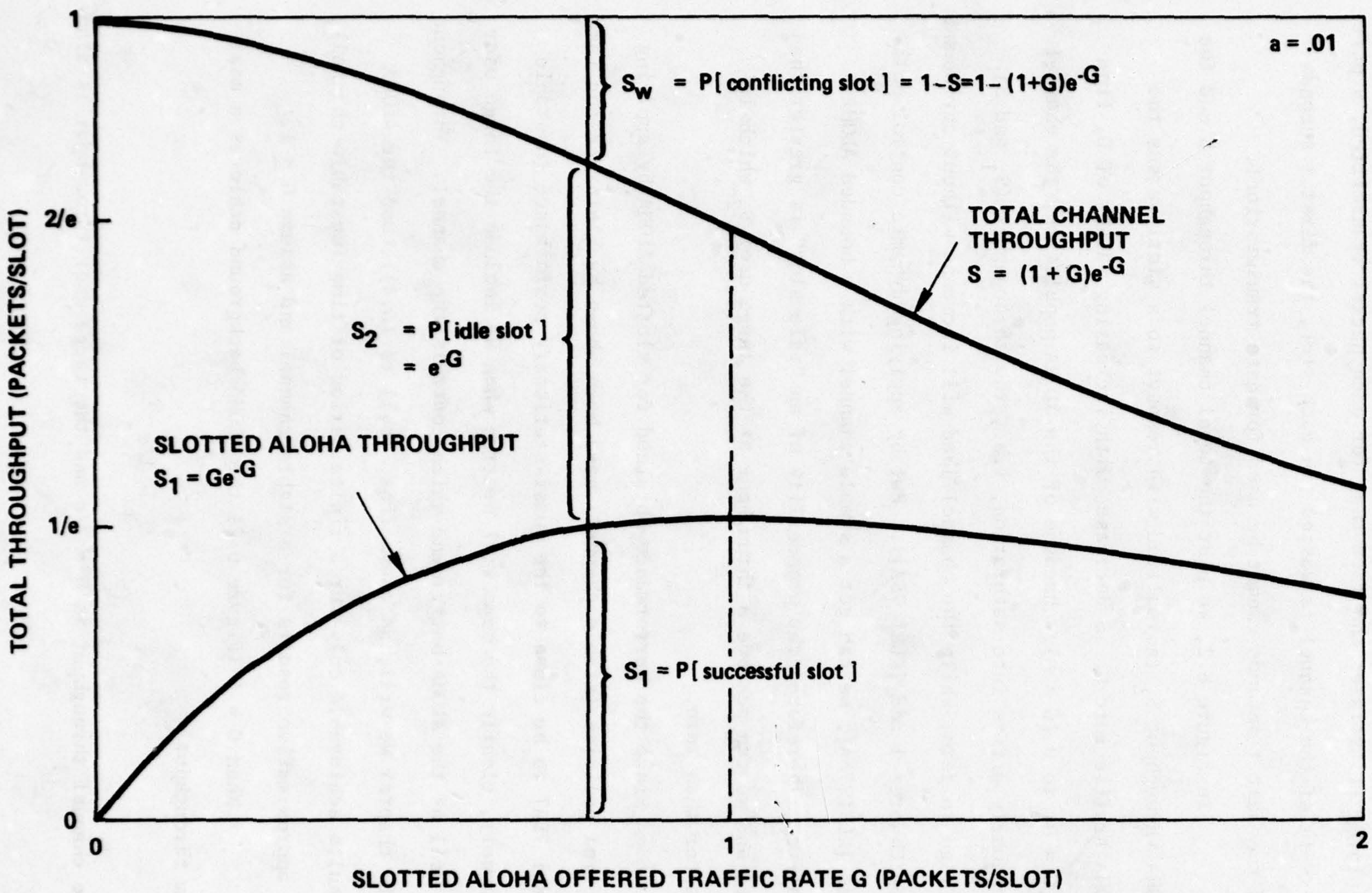


Figure 6.2. MACS: Total Channel and Aloha Throughputs versus G.

$$S_2 = 1/e$$

When G goes to zero, so does S_1, but S_2 increases to one.

From Figure 6.2, it is clear that the probability of a conflict in a given slot (conflicting slot) between more than one user (equal to $1 - 2/e$ when $G = 1$) goes to zero as G goes to zero; meanwhile, the probability that the slot is idle (large user throughput S_2) increases up to one when G goes to zero. (Figure 6.2)

The probability that a given slot is "conflicting" is

$$S_W = 1 - S$$

and represents the wasted channel capacity. In Figure 6.3, S_W, S_1 and S_2 are plotted versus G.

Before solving for the packet delay at the large user, let us compare the large user and the total throughputs predicted with our model (Eqs. (6.1) and (6.4)) with the throughput predicted by the model called the "large user model" that we mentioned above [ABRA 73].

Large User Model:

In this model, first studied by L. Roberts in an unpublished note and later generalized by Abramson [ABRA 73] and Lam [LAM 74], we consider a large buffered user and a population of small users (modelled by an infinite population as described above). The large user and the small users compete on the same channel in a slotted ALOHA fashion. We define by Λ_1 and Λ_2 the throughput rates (normalized with respect to $T = \frac{b_m}{W}$) of the small users' population and the large user, respectively.

$$\Lambda_1 = \lambda_1 T$$

$$\Lambda_2 = \lambda_2 T$$

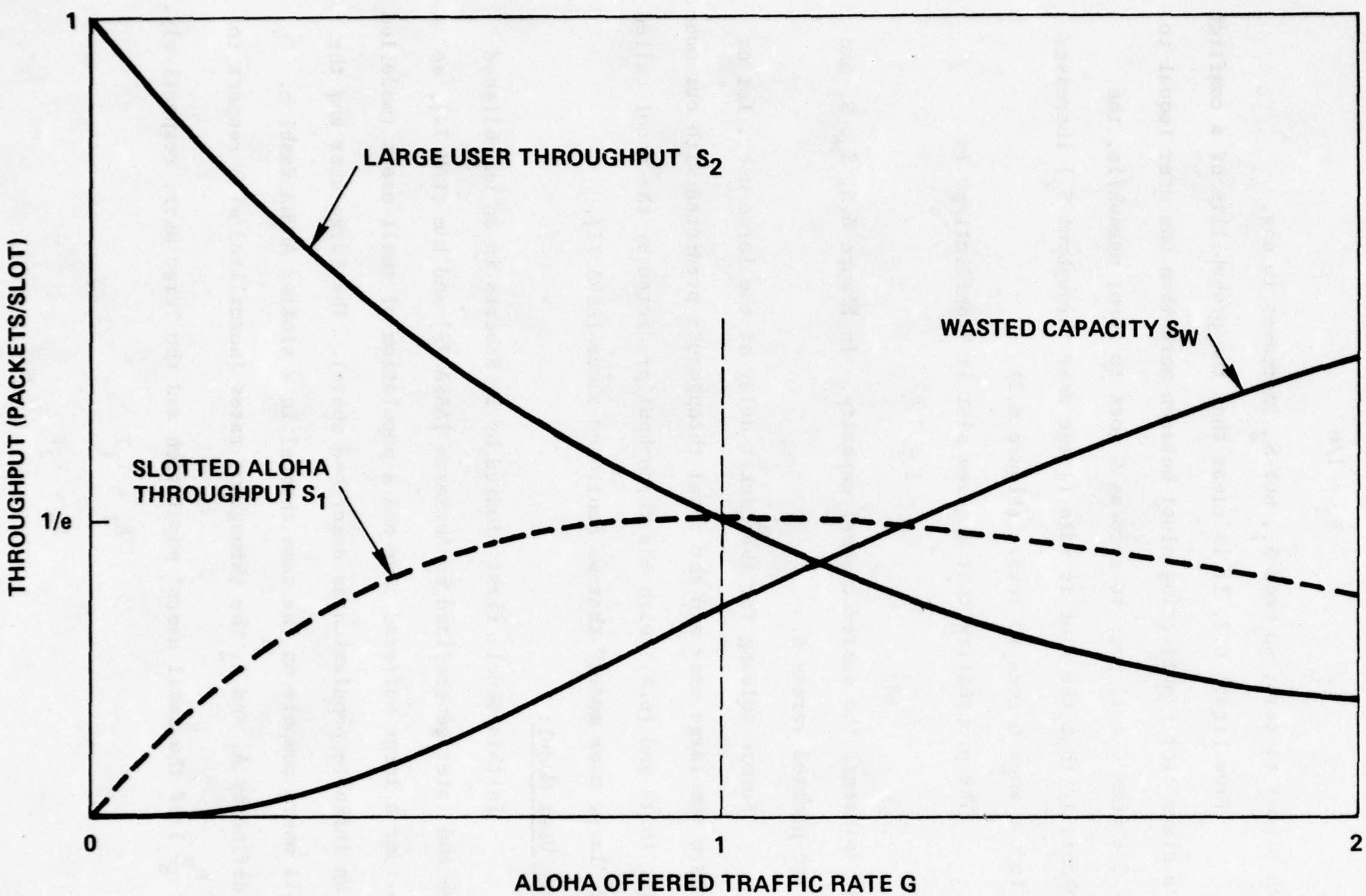


Figure 6.3. MACS: Wasted Capacity and Large User Throughput versus G.

The large user's packets conflicting with small users' packets must be retransmitted, and the offered traffic rate at the large user G_2 is grater than Λ_2. G_1 represents the offered traffic rate from the small users. G_1 and G_2 are normalized with respect to T. It can be shown [ABRA 73], [LAM 74] that the limiting throughputs are

$$\Lambda_1 = G_1 e^{-G_1}(1 - G_2) \tag{6.6}$$

$$\Lambda_2 = G_2 e^{-G_1} \tag{6.7}$$

The limiting channel throughput rate is then

$$\Lambda = \Lambda_1 + \Lambda_2 = (G_1 + G_2 - G_1 G_2)e^{-G_1}$$

Then, given either Λ_1 or Λ_2, Λ is maximized if the condition

$$G_1 + G_2 = 1$$

is satisfied [LAM 74].

The interest of this approach is that we can exceed the 1/e limitation on the capacity of a slotted ALOHA channel. (If the small users' traffic G_1 goes to zero, then the channel is dedicated to the large user and a channel throughput rate arbitrarily close to unity can be achieved--this is then simply an M/D/1 system).

However, the throughput performance predicted by this model is much lower than the performance predicted by our model, as can be seen in Figure 6.4 where we plot Λ_2 and Λ versus the ALOHA traffic rate G_1 for both systems.

6.1.3 Packet Delay at the Large User

In this section we wish to solve for the average delay incurred by a packet at the large user. As usual, we define the packet delay as

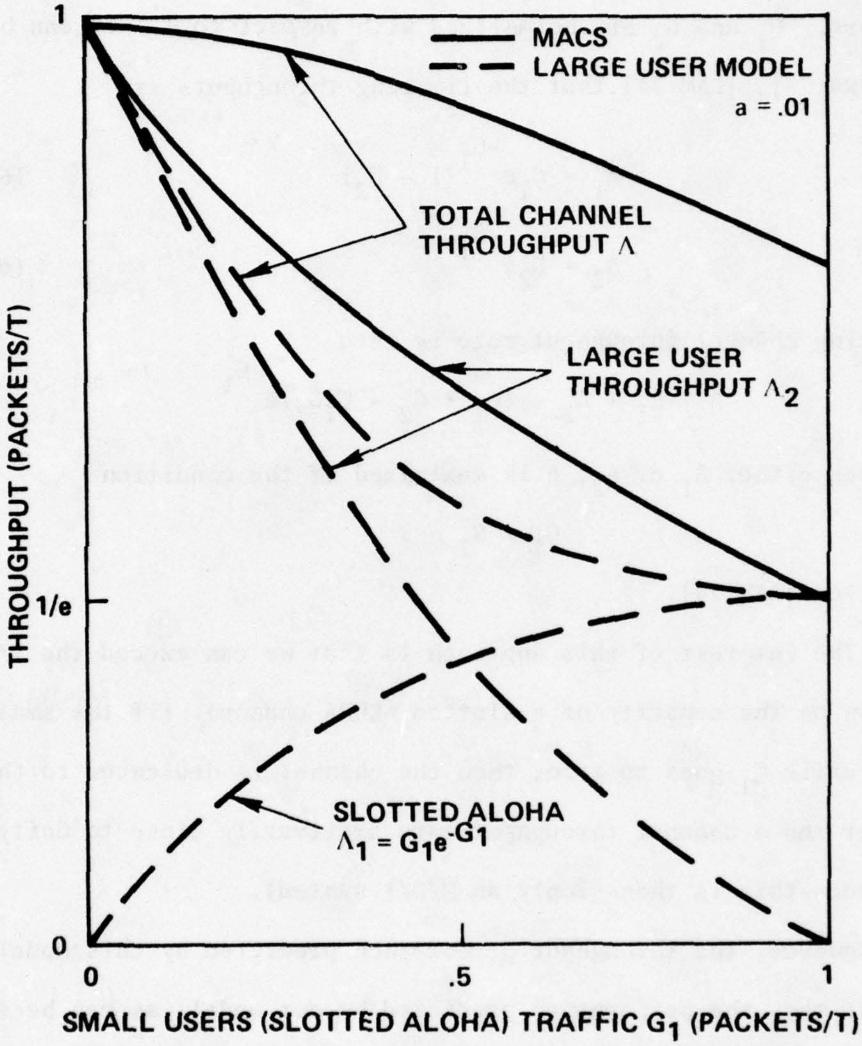


Figure 6.4. MACS and Large User Model: Channel and Large User Throughput *versus* Small Users' Traffic.

the time period elapsing from the generation instant to the end of transmission. Let us denote by Δ_2 the expected delay normalized with respect to a slot and by D_2 the expected delay normalized with respect to the packet transmission time T:

$$D_2 = (1 + 2a)\Delta_2 \tag{6.8}$$

Let us define the "service time" $\tilde{x}$ of a packet, as in Section 6.1.2, as the number of slots it takes to transmit the packet from the first time we sense the carrier for this packet until the end of the transmission of this packet. From Section 6.1.2, we know that $\tilde{x}$ is geometrically distributed with parameter e^{-G} (G is the ALOHA traffic rate).

A large user packet which finds the large user empty upon generation must wait until the beginning of the next slot before sensing the carrier, i.e., before the "service" starts. Thus we cannot represent the large user simply by an M/G/1 queueing system. However, we can model the large user as an M/G/1 queueing system with rest period* where the rest period *is deterministic with length one slot*; the Poisson arrival process has an intensity λ_2 and the service time is geometric (with parameter e^{-G}). From Appendix A (Eq. (A.26)), we know that the average delay D in an M/G/1 queue with rest period is given by:

$$D = \overline{x} + \frac{\lambda \overline{x^2}}{2(1 - \lambda\overline{x})} + \frac{\overline{T_0^2}}{2\overline{T_0}} \tag{6.9}$$

---

*See Section 3.4 or Appendix A for the definition of an M/G/1 queue with rest period.

where $\overline{x}$ and $\overline{x^2}$ are the first and the second moments of service time, T_0 and $\overline{T_0^2}$ are the first and second moments of the rest period and λ is the arrival rate.

Recall that S_2' is the arrival rate at the large user normalized with respect to a slot (normalized input rate). We have

$$S_2' = \lambda_2(1 + 2a)T \tag{6.10}$$

The rest period being deterministic (with a length of one slot) we have

$$\frac{\overline{T_0^2}}{2T_0} = \frac{1}{2} \quad \text{(slots)} \tag{6.11}$$

Finally we have

$$\overline{x} = e^G \quad \text{(slots)}$$

$$\overline{x^2} = (2 - e^{-G})e^{2G} \quad ([\text{slots}]^2) \tag{6.12}$$

Substituting Eqs. (6.10) through (6.12) into Eq. (6.9), we obtain the following expression for the normalized average packet delay at the large user:

$$\Delta_2 = \frac{2\ S_2'}{2(e^{-G} - S_2')} + \frac{1}{2} \quad \text{(slots)} \tag{6.13}$$

6.1.4 Delay-Throughput Characteristics

ALOHA and large user throughputs are compared on Figure 6.5 for $G \geq 1$, and on Figure 6.6 for $G \leq 1$. The shaded regions in both figures identify the feasible regions. In Figure 6.5 ($G \geq 1$), the contour

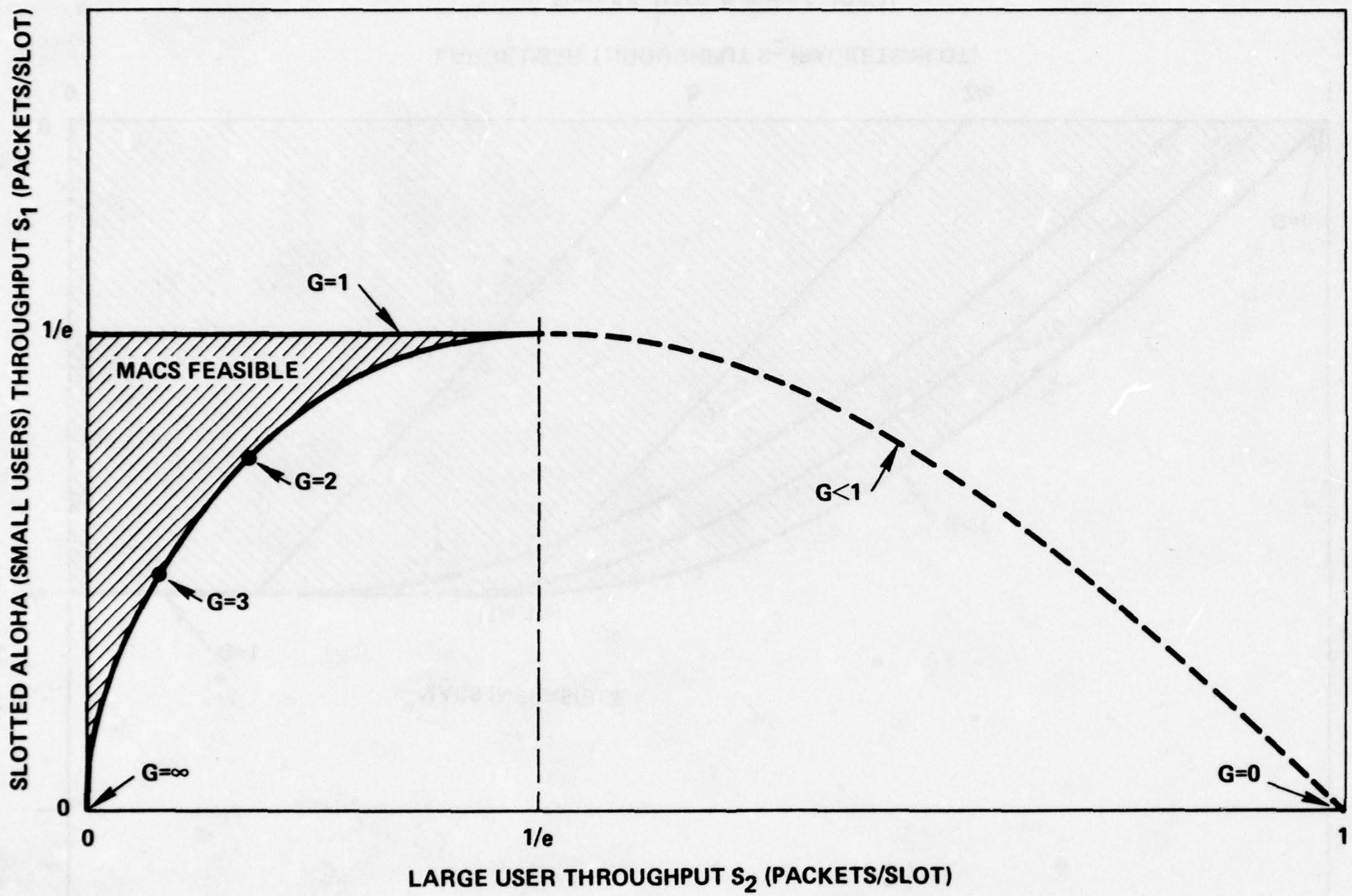


Figure 6.5. MACS: S_1 versus S_2 ($G \geq 1$).

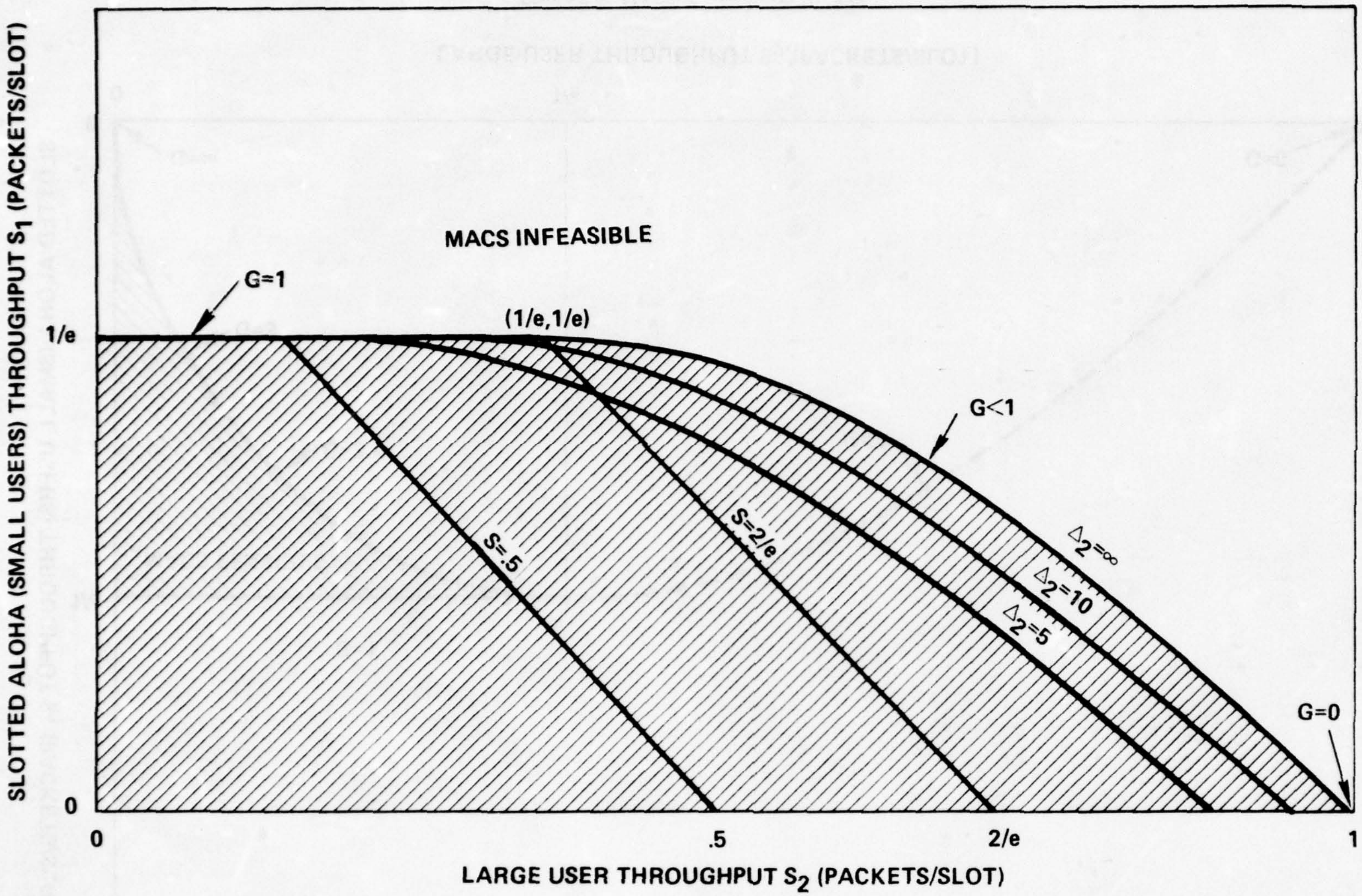


Figure 6.6. MACS: S_1 *versus* S_2 ($G < 1$)

S_2 versus S_1 for $G < 1$ is represented by dotted lines. When G increases, both S_1 and S_2 go to zero (the channel drifts into saturation). When $G = 1$, both S_1 and S_2 are equal to $1/e$. In Figure 6.6 ($G \leq 1$), the contour S_2 versus S_1 for $G < 1$ is a limiting contour which corresponds to infinite delays. Contours S_2 versus S_1 are drawn for some finite values of Δ_2 (packet delay at the large user). The region inside the boundary delimited by the axes S_1, S_2 and the contour S_2 versus S_1 for a given value of Δ_2 (e.g., $\Delta_2 = 5$) represents the feasible achievable throughputs at the large user and small users with the constraint of a maximum delay at the large user equal to Δ_2 slots. Clearly, with a maximum delay $\Delta_2 = 5$, the maximum achievable throughput at the large user is close to the limiting throughput (infinite delays: $\Delta_2 = \infty$), for all values of ALOHA throughput. Constant total throughput S lines are drawn in Figure 6.6 and illustrate how S_2 decreases when S_1 increases for a given value of S. The line $S = 2/e$ delimits the region where there are no infinite delays (all those S_1 and S_2 for which $S = S_1 + S_2 < 2/e$).

6.1.5 Split Channel versus Single Channel

At the beginning of this chapter, we introduced the idea of including traffic from two independent sources (small users and large user) on the same channel. The motivation behind this inclusion is two-fold:

i) Since the small users are accessing the channel in a slotted ALOHA fashion*, we want to increase the channel utilization beyond the 1/e limitation of slotted ALOHA channel capacity.

---

*See the introduction of this chapter and Section 1.2 for the discussion about the choice of ALOHA access mode for the small users.

ii) By increasing the bandwidth available to the small users we provide them better throughput-delay performance.

The results of Section 6.1.2 justify the inclusion of traffic from the two different sources on the same channel since this provides a very good total channel utilization (> 2/e) and a significant large user throughput (> 1/e).

Here we approach the problem from a synthesis viewpoint. That is, given Source 1 (small users) and Source 2 (large user), the question is whether one should split the channel so that one part, αW of the bandwidth ($\alpha < 1$), is assigned to the large user and the other part, $(1 - \alpha)W$, to the small users; or whether one should mix the two traffic sources according to the MACS access mode studied above.

We already know that for the small users, the best performance is obtained when they are provided the entire bandwidth. For the small users it is better not to split the channel, for by splitting the channel we increase the small users' delay by a factor of $\frac{1}{1 - \alpha}$ and it is not certain that we achieve the same throughput.

How about the large user? Is the packet delay lower with a split channel or with a mixed use of the entire bandwidth at the large user? How significant is the degradation in throughput at the large user when we mix the traffic?

Our optimization problem now becomes finding, for a channel of limited bandwidth W and for given values of the input rates of the large user (λ_2) and of the small users (λ_1), the value of α, namely α_0, such that a split channel with $\alpha > \alpha_0$ provides a lower delay at the large user than the mixed use of the entire bandwidth, and a split channel with $\alpha < \alpha_0$ provides at the large user a larger delay than the mixed use of the entire bandwidth.

Splitting the channel with $\alpha > \alpha_0$ will produce a small users' delay of at least $1/(1 - \alpha_0)$ times larger than that obtained under a mixed mode. Therefore, if α_0 is too close to one (e.g., $\alpha_0 \geq .5$) one may prefer to choose the mixed mode and thus penalize the large user, rather than the small users.

More precisely, let us call $D_S(\alpha)$ the expected packet delay at the large user when the channel is split into two parts of respective bandwidths, αW for the large user and $(1 - \alpha)W$ for the small users. $D_S(\alpha)$ is normalized with respect to $T = b_m/W$, T being the packet transmission time on a channel of bandwidth W. Modeling the large user by an M/D/1 queue, we have by the Pollaczek-Khinchin formula [KLEI 75C]

$$D_S(\alpha) = \frac{1}{\alpha}\left[1 + \frac{\lambda_2(T/\alpha)}{2(1 - \lambda_2(T/\alpha))}\right] \tag{6.14}$$

Substituting Eq. (6.13) into Eq. (6.8), we obtain the expected packet delay D_2, at the large user (normalized with respect to T), under the mixed mode (MACS)

$$D_2 = \left[\frac{2 - S_2'}{2(e^{-G} - S_2')} + \frac{1}{2}\right](1 + 2a) \tag{6.15}$$

We wish to solve for α_0, which gives

$$D_2 = D_S(\alpha) \tag{6.16}$$

such that:

1) if $0 < \alpha < \alpha_0$, $D_2 < D_S$

2) if $\alpha_0 < \alpha < 1$, $D_S < D_2$

3) $\alpha_0 \geq \lambda_2 T = \Lambda_2$

4) $$\frac{1 - \alpha_0}{e} \geq \frac{S_1}{1 + 2a} = \Lambda_1 \qquad (6.17)$$

The third constraint insures us that the input rate at the large user when the channel is split (with a value of $\alpha = \alpha_0$) is less than the service rate, i.e., the number of packets tranmitted per unit of time. In other words, the utilization factor ρ (see [KLEI 75C]) of the M/D/1 queue representing the large user must be less than 1. We have

$$\rho = \lambda_2 \frac{T}{\alpha} < 1 \qquad \text{for all } \alpha \geq \alpha_0$$

where $\frac{T}{\alpha}$ is the transmission time of one packet (α/T is the service rate). Thus we get the third constraint in (6.17). Similarly, the fourth constraint indicates that the feasible values of α for the small users are such that on the separate channel of bandwidth $(1 - \alpha)W$, the small users' throughput is less than or equal to $1/e$.

Equating the right-hand sides of Eq. (6.14) and Eq. (6.15), and substituting Eq. (6.10) we obtain a second degree equation in α, for which there is at most one solution, α_0 respecting the constraints in Eq. (6.17).

It turns out that for $G_1 > .2$, the solution α_0 of Eq. (6.16) does not satisfy the fourth constraint (Eq. (6.17)). In other words, when $G > .2$ $(\Lambda_1 > .16)$, to get a large user delay lower than that obtained with the mixed mode, one must dedicate a part greater than $\alpha_0 W$ to the large user, such that the remaining part $(1 - \alpha_0)W$ of the bandwidth is not large enough to achieve the small users' throughput (Λ_1); this, of course, is unacceptable.

In Figures 6.7 and 6.8, α is plotted versus Λ_2, the large user's input rate (normalized with respect to T), for two small users' traffic: $G_1 = .01$ $(\Lambda_1 = .01)$ and $G = .1$ $(\Lambda_1 = .09)$. Five regions appear in each of these figures.

In the first (shaded) region, the mixed mode is not feasible. A vertical line delimits this region; at the large user, one cannot achieve a throughput greater than .98 $(\Lambda_1 = .01$, Figure 6.7) or .89 $(\Lambda_1 = .09$, Figure 6.8) because of the presence of the small users.

In the second and third regions, the "split-channel" mode is not feasible; the shaded region $(\alpha < \Lambda_2)$ is forbidden, since with a bandwidth αW one cannot achieve a throughput Λ_2 greater than α at the large user. The shaded region $(\alpha > (1 - e\Lambda_1))$ is also forbidden, since with a bandwidth $(1 - \alpha)W$, one cannot achieve a throughput greater than $1/e$ for the small users.

In the two remaining regions, both the split-channel mode and the mixed mode are feasible and comparable. The contour for α_0, which is the solution of Eq. (6.16), is plotted versus Λ_2. The contour α_0 versus Λ_2 delimits the two regions. Below this contour, delays at the large user are lower with the mixed mode than those obtained with a split channel $(D_2 < D_S)$. Above this contour the split-channel mode provides lower delays $(D_S < D_2)$ than the mixed mode.

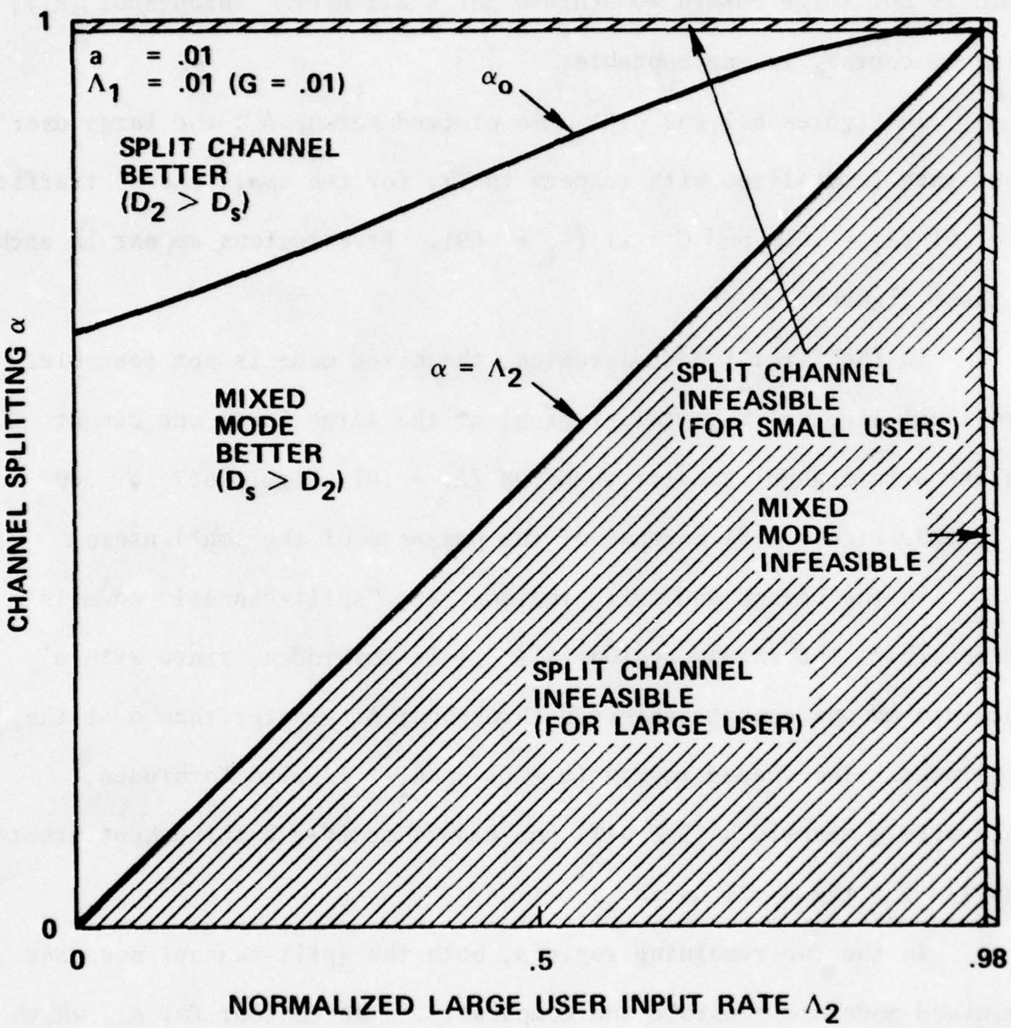


Figure 6.7. MACS and Split Channel: α vs Λ_2

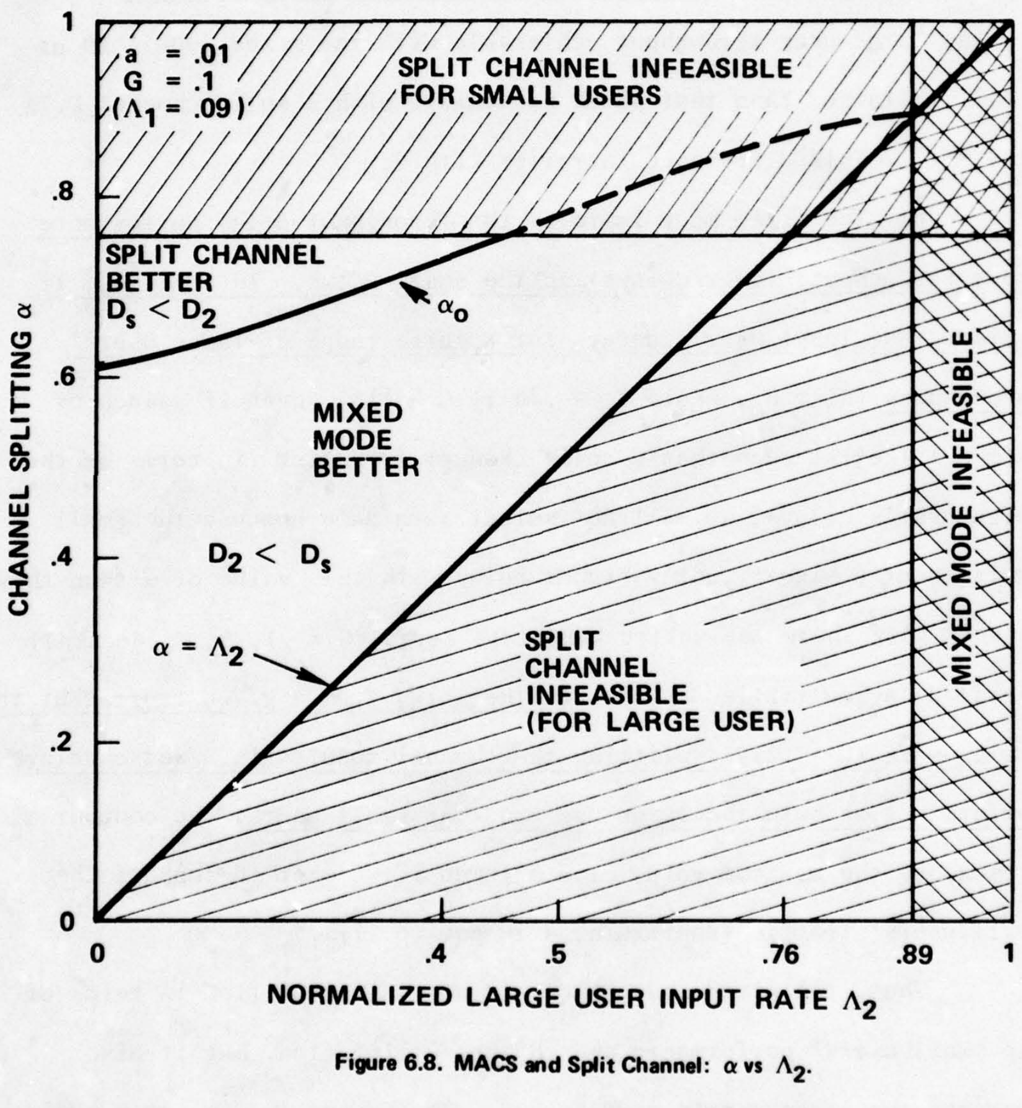


Figure 6.8. MACS and Split Channel: α vs Λ_2.

From Figures 6.7 and 6.8, it is clear that:

a) The larger the small users input rate is, the more important is the increase in the total channel throughput achievable with the mixed mode compared to that achieved when the channel is split. Given Λ_1, the large user throughput achievable with the mixed mode (.89 at G =.1) is larger than that which we achieve with a split channel (.76 at G = .1). This difference increases with G.

b) The mixed mode improves the throughput-delay performance (more throughput, lower delays) of the small users. In addition, it improves the large user's delay, for a large range of large users' throughputs ($\Lambda_2 > \beta_0$, where $\beta_0 < .46$ if $G > .1$). Even if values of α ($\alpha > \alpha_0$) exist such that a split channel is better (in terms of the large user's delay), we will not select such an α because the small users incur a significantly higher delay with this value of α than they do when they share the entire bandwidth (e.g., G = .1, $\Lambda_2 < .46$ small users' delay multiplied by 3). Furthermore, from a given traffic at the small users ($G > .2$), splitting the channel results in a worse delay performance at both the large user and the small users; the contour α_0 lies above the maximum value of α allowed if we want to support the small users' traffic (constraint 4 in Eq. (6.17)).

Thus, the mixed mode (MACS) is not only justified in terms of the small users' performance and channel utilization, but it also improves the large user's performance. By sharing the entire bandwidth, the large user achieves more throughput with delays comparable to those incurred when he is dedicated a part of the channel.

6.1.6 Conclusion

The MACS access mode is particularly suitable for a population composed of a large buffered user and a large number of small bursty users.

Indeed, it was shown in Section 6.1.2 that MACS provides a good channel utilization (> 2/e), and that the achievable throughput at the large user is important when the small users' traffic is not too large, and in any case greater than 1/e. In addition, the small users' packet delay is significantly decreased by allowing the latter to share the entire bandwidth. Finally, it was shown in Section 6.1.5 that MACS performs better than a mode where the channel is split into two sub-channels, one dedicated to the small users, the other one to the large user; MACS improves the performance of both the large user and the small users.

Until now, the population of small users has been considered large in number. In the next section we consider the case of a small number of users requiring buffering capability and we introduce a mixed mode where we include the traffic of these small buffered users, as well as the traffic of a large buffered user on a single channel.

6.2 Mixed Alternating Priorities Carrier Sense Multiple Access MAPCS

We consider a population composed of N buffered users, called small users (N not too large, e.g., $N \leq 20$) and one large buffered user. All (N + 1) users are within range and in line of sight of each other. All users transmit packets of length b_m bits and have Poisson packet arrival (generation) processes which are independent of each other. The (combined) intensity of the small users arrival process is λ_1; the arrival process intensity at the large user is λ_2. We wish to share a single channel of bandwidth W among those (N + 1) users. As in the previous section, we require a short response time (packet delay) for a small user, and thus we wish to give priority to the small users over the large user. Since we have a finite number of buffered users hearing each other, one may here apply the methods developed in Chapter 3 for multiplexing the (N + 1) users over the channel.

6.2.1 Extension of the Access Modes Studied in Chapter 3

In Chapter 3, we introduced four protocols, HOL, AP, RR and RO. Below, we define a fifth protocol which is, as a matter of fact, a mixture of HOL and any of the three protocols, AP, RR and RO. This protocol has exactly the same features as the four protocols developed in Chapter 3. In particular, the channel model is that introduced in Chapter 2, and the slot configuration and operational features are those described in Section 3.1. We define the new protocol by defining a new assignment rule. Denote by $u_1, \ldots, u_N, u_{N+1}$ the (N + 1) users. Then the assignment rule orders the (N + 1) users in a given slot. Let

$$u_{\pi_1}, u_{\pi_2}, \ldots, u_{\pi_N}, u_{\pi_{N+1}}$$

be the ordered sequence of users in the current slot ($\{\pi_1, \ldots, \pi_N, \pi_{N+1}\}$ is a permutation of $1, 2, \ldots, N, N+1$). The user u_{π_1} has the highest priority to transmit a packet in this slot. If u_{π_1} has no packet to transmit, u_{π_2} has the highest priority, etc. If $u_{\pi_1}, \ldots, u_{\pi_N}$ are idle, and if $u_{\pi_{N+1}}$ has a packet to transmit in this slot, he will transmit it. Otherwise the slot is unused and the assignment rule orders the users in sequence (possibly the same sequence) for the next slot, at the beginning of which the process starts again. By sensing the carrier, each user knows if higher priority users are idle or not, except the highest priority user who transmits without sensing the carrier (see Section 3.1). The slot configuration is reproduced below in Figure 6.9.

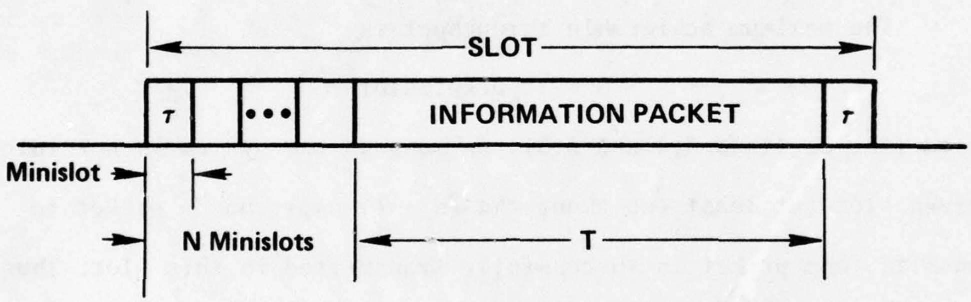


Figure 6.9. Slot Configuration.

6.2.2 Protocol

The large user always has the lowest priority and thus transmits a packet (if any) in a given slot only if all small users are idle in this slot. The N small (identical) users compete for the channel according to the AP protocol (we could also have chosen RR or RO). They are not affected by the presence of the large user and the protocol under which they operate is that described in Section 3.2.2*. The large user senses the carrier at the beginning of each slot if he has a packet to transmit:

---

* except that the overhead in each slot is now (N + 1) minislots (Figure 6.9) instead of N (Figure 3.1), and therefore the transmission of the (information) packet at each small user is delayed by a supplementary minislot (see Section 3.2.2).

(1) If after N minislots ($N\tau$ seconds) the carrier is not detected, the large user transmits his packet.

(2) Otherwise (carrier present), he waits for the next slot, and then operates as above.

6.2.3 Delay-Throughput Characteristics

6.2.3.1 Throughput, Channel Capacity

The maximum achievable throughput is

$$S = 1 \text{ packet/slot}$$

Indeed (see Sections 3.1 and 3.3), as long as the system is not idle in a given slot (at least one among the (N + 1) users has a packet to transmit), one packet is successfully transmitted in this slot. Thus, we have a conflict-free system. However, in each slot there is an overhead of (N + 1) minislots lost for packet transmission. Therefore the channel capacity is

$$C = \frac{1}{1 + (N + 1)a} \tag{6.18}$$

Denote by S_1 the (combined) small users' throughput and by S_2 the large user's throughput; S_1 and S_2 are normalized with respect to one slot. We then have

$$S_1 = \lambda_1(1 + (N + 1)a)T \tag{6.19}$$

$$S_2 = \lambda_2(1 + (N + 1)a)T \tag{6.20}$$

$$S_1 + S_2 = S \leq 1 \tag{6.21}$$

For a given small user's throughput S_1, the maximum achievable throughput at the large user (with infinite delays) is

$$S_2 = 1 - S_1 \ .$$

6.2.3.2 Comparison with Slotted ALOHA Finite Population

As we did for MACS, we compare the throughput performance under MAPCS with that obtained when the (N + 1) users (1 large user and N small users) compete on the same channel in a slotted ALOHA fashion. (Here all users have the same priority.) This slotted AL HA finite population model was first studied by Abramson [ABRA 73].

This approach, as in the approach to the slotted ALOHA large user (plus infinite population) model (see Section 6.1.2 and [ABRA 73]), allows us to exceed the 1/e limit on the slotted ALOHA capacity, when the large user's traffic is higher than the (combined) small users' traffic. Indeed the large user is buffered and his packets never conflict with each other; they conflict only with the small users' packets. In addition, this model does not require all users to hear each other. We use the same definitions and notation as those of Section 6.1.2 for the "large user model." The only difference between the two models is that the population of small users is finite in number. It can be shown [ABRA 73], [LAM 74] that

$$\begin{cases} \Lambda_1 = G_1\left(1 - \frac{G_1}{N}\right)^{N-1} (1 - G_2) & (6.22) \\ \\ \Lambda_2 = G_2\left(1 - \frac{G_1}{N}\right)^{N} & (6.23) \end{cases}$$

and the maximum channel throughput,

$$\Lambda = \Lambda_1 + \Lambda_2 \tag{6.24}$$

when

$$G_1 + G_2 = 1 \ . \tag{6.25}$$

Throughput and traffic rates are normalized with respect to the packet transmission time $T = b_m/W$.

When the small users' traffic is very small ($G_1 = 0$, $\Lambda_1 = 0$), the large user may then achieve a throughput of 1; $\Lambda_2 = 1$. In other words, the total channel utilization may reach 1 (when the small users are quiet).

In the MAPCS mode, for all values of the small users' traffic, the maximum total channel utilization cannot exceed $\frac{1}{1 + (N + 1)a}$ (Eq. (6.18)). In Chapter 3 we discussed at length the degradation of performance under AP, RR and RO, with increasing values of N. We are interested in finding, for each value of the throughput rate Λ_1 of the small users, the critical value of the number of small users N_{crit} beyond which the maximum channel throughput C under MAPCS is lower than that Λ obtained under slotted ALOHA: $C = 1/1 + (N + 1)a = \Lambda$.

Solving the system composed of the previous equation and Eqs. (6.22) through (6.25), we get N_{crit} as a function of Λ_1. N_{crit} is plotted versus Λ_1 in Figure 6.10. Four regions are shown in this figure. The first (shaded) region is delimited by the contour $\Lambda_1 = \frac{1}{1 + (N + 1)a}$. Above this contour, MAPCS is infeasible. The second (shaded) region is the infeasible region for ALOHA; for $\Lambda_1 > 1/e$, the maximum number N of users decreases very fast with Λ_1 ($N = 5$ for $\Lambda_1 = .4$, $N = 2$ for $\Lambda_1 = .5$). The contour N_{crit} versus Λ_1 for $\Lambda_1 < 1/e$ delimits the two remaining (non-shaded) regions where both ALOHA and MAPCS are feasible. Above this contour, ALOHA provides a higher total channel throughput than MAPCS does. MAPCS perfoms better below this contour.

In conclusion, one main advantage of MAPCS is to allow much

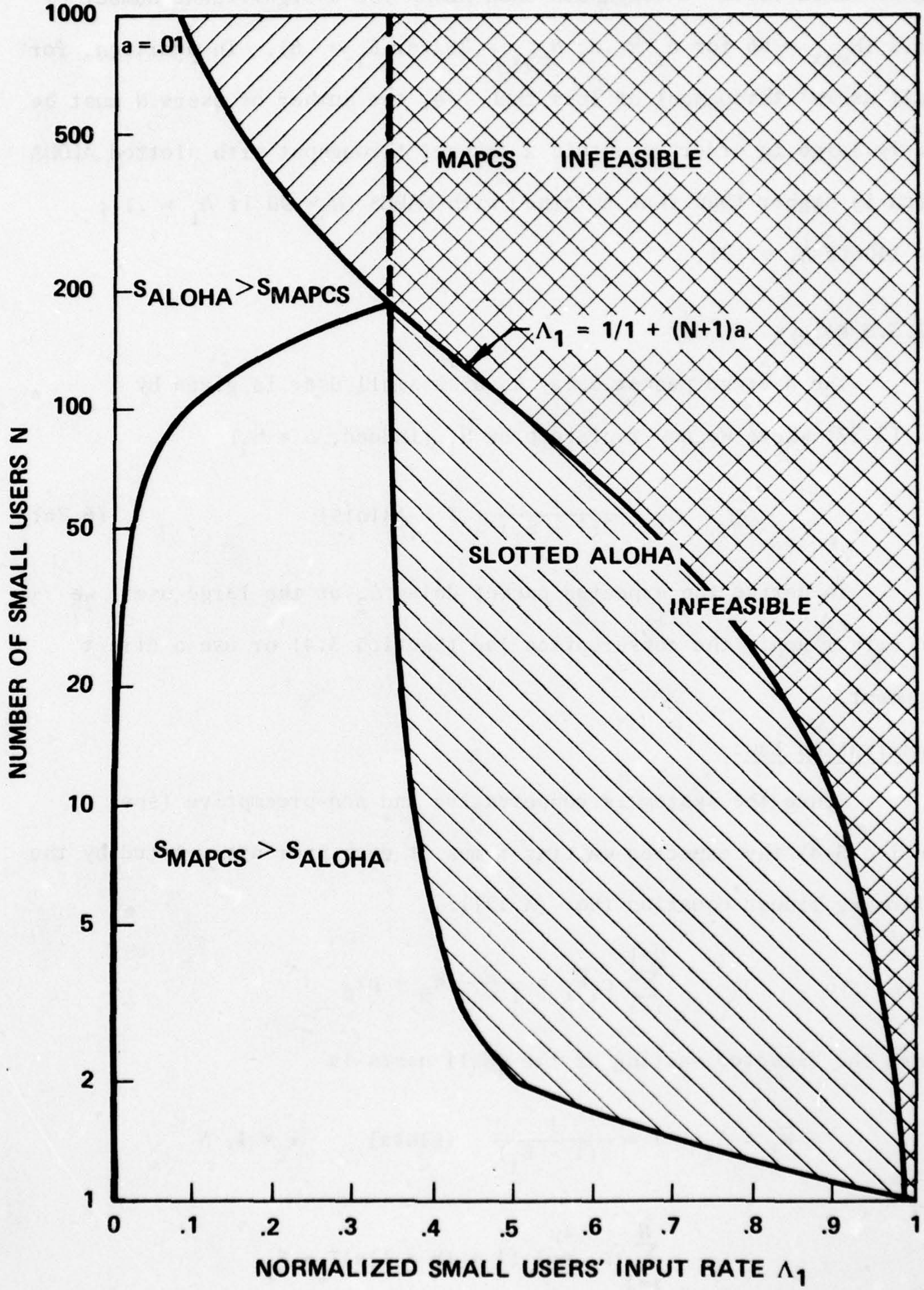


Figure 6.10. Total Channel Throughput with Slotted Aloha and MAPCS: N_{CRIT} vs Λ_1.

higher small users' throughputs than ALOHA for a significant number of users (N_{max} = 66 for Λ_1 = .6; N_{max} = 25 for Λ_1 = .8). In addition, for small users' throughput of less than 1/e, the number of users N must be fairly large in order to obtain a channel throughput with slotted ALOHA which is higher than that obtained with MAPCS ($N \simeq 90$ if $\Lambda_1 = .1$; $N \simeq 140$ if $\Lambda_1 = .2$).

6.2.3.3 Delays

The average packet delay Δ_1 at a small user is given by Eq. (3.27) where we now replace ρ by S_1 (indeed, $\rho = S_1$)

$$\Delta_1 = \frac{1}{2(1 - S_1)} + 1 \quad \text{(slots)} \qquad (6.26)$$

To derive the expected packet delay Δ_2 at the large user, we can either apply the conservation law (Section 3.4) or use a direct argument.

Conservation Law:

Since the system is conservative and non-preemptive (see Section 3.4) the expected waiting times at each user are related by the following linear equation (Eq. (3.12)):

$$\sum_{i=1}^{N+1} \rho_i \overline{W}_i = \frac{\rho}{1 - \rho} \overline{W}_0 + \rho\tau_0$$

where the expected waiting at the small users is

$$\overline{W}_i = \Delta_1 - 1 = \frac{1}{2(1 - S_1)} \quad \text{(slots)} \qquad i = 1, N$$

and

$$\sum_{i=1}^{N} \rho_i = \lambda_1(1 + (N + 1)a)T = S_1$$

$$\rho_{N+1} = \lambda_2(1 + (N + 1)a)T = S_2$$

$$\rho = \sum_{i=1}^{N+1} \rho_i = S_1 + S_2 = S$$

where $\overline{W}_0$ as given by Eq. (3.7) becomes

$$\overline{W}_0 = \frac{S}{2} \quad \text{(slots)}$$

$$\tau_0 = \frac{\overline{T_0^2}}{2\overline{T}_0} = \frac{1}{2} \quad \text{(slot)}$$

and $\overline{W}_{N+1} = \Delta_2 - 1$ is the expected waiting time of a packet at the large user, our unknown.

Then Eq. (3.12) becomes

$$\overline{W}_{N+1} = \frac{1}{2(1 - S)(1 - S_1)} \quad \text{(slots)} \tag{6.27}$$

and the expected packet delay is

$$\Delta_2 = 1 + \frac{1}{2(1 - S)(1 - S_1)} \tag{6.28}$$

<u>Direct Argument</u>:

For the large user, the N small users behave like a single user with higher priority. We then obtain the packet delay at the large user by considering the whole system as an HOL system with two users, one with (combined) input rate S_1 (per slot) and the other one with input rate S_2 (per slot).

Therefore, the delay at the large user (user with lower priority) is given by Eq. (3.28)

$$\Delta_2 = 1 + \frac{1}{2(1 - \sigma_2)(1 - \sigma_1)} \tag{6.29}$$

where

$$\sigma_1 = \lambda_1(1 + (N + 1)a)T = S_1 \tag{6.30}$$

$$\sigma_2 = \lambda_1(1 + (N + 1)a)T + \lambda_2(1 + (N + 1)a)T = S \tag{6.31}$$

Substituting Eq. (6.30) and Eq. (6.31) into Eq. (6.29), we obtain Eq. (6.28).

It remains to compare the performance predicted by our model (Eqs. (6.18), (6.26) and (6.28) with that obtained when we dedicate separate channels to the large user and the small users.

6.2.4 MAPCS versus Split Channel

Given the two independent sources, i.e., the large user and the small users, the question arises as to whether one should mix them on the same single channel of bandwidth W, according to a mode like MAPCS, or whether one should split the channel into two parts, one of bandwidth αW for the large user, and one of bandwidth $(1 - \alpha)$ for the small users.

Since the small users control the channel under MAPCS (the small users have priority over the large user), they perform best with MAPCS which provides them the entire bandwidth. By splitting the channel, we decrease the maximum achievable throughput for the small users by a factor of approximately $(1 - \alpha)$ and increase the packet delay by a factor close to $\frac{1}{1 - \alpha}$. Therefore, one reason for introducing the MAPCS mode is to decrease the packet delay and increase the achievable throughput for the small users.

In MAPCS, the large user's performance is much affected by the presence of the small users. In particular, if the small users input rate (Λ_1) is high, the large user's throughput (Λ_2) is small, $\left(\Lambda_2 = \frac{1}{1 + (N + 1)a} - \Lambda_1\right)$, and the average packet delay at the large user may become prohibitive so that one may be led to choose a separate channel for the large user.

As in Section 6.1.4, we want to determine the regions in the (α, Λ_2) plane for various values of the input rate Λ_1 of the small users, in which splitting the channel will provide a better performance at the large user without degrading the small users performance too much (α not too large).

When the large user is dedicated a separate channel of bandwidth αW, the expected packet delay at the large user $D_S(\alpha)$ (normalized with respect to $T = b_m/W$) is given by Eq. (6.14).

The expected packet delay at the large user D_2, normalized with respect to T, under MAPCS is

$$D_2 = (1 + (N + 1)a)\Delta_2$$

where Δ_2 is given by Eq. (6.28). We have

$$D_2 = (1 + (N + 1)a) \left[1 + \frac{1}{2(1 - S)(1 - S_1)}\right] \qquad (6.32)$$

Recalling that

$$S = (\lambda_1 + \lambda_2)(1 + (N + 1)a)T \qquad (6.33)$$

$$= (\Lambda_1 + \Lambda_2)(1 + (N + 1)a)$$

and

$$S_1 = \lambda_1(1 + (N + 1)a)T = \Lambda_1(1 + (N + 1)a) \qquad (6.34)$$

We can find α_0 which solves the following system

$$D_2 = D_S(\alpha)$$

$$\alpha_0 > \Lambda_2$$

$$\alpha_0 < 1 - \Lambda_1 \tag{6.35}$$

$$\text{if } 0 < \alpha < \alpha_0 \quad \text{then} \quad D_2 < D_S$$

$$\text{if } \alpha_0 < \alpha < 1 \quad \text{then} \quad D_2 > D_S$$

Equating the right-hand side of Eq. (6.14) and Eq. (6.32), substituting Eq. (6.33) and Eq. (6.34), we obtain a second degree equation in α. For given values of the (combined) input rate of the small users Λ_1, and the input rate of the large user Λ_2, the system (6.35) has one and only one solution α_0.

In Figures 6.11 and 6.12 we plot α_0 versus Λ_2 for N = 25 users, for various values of the input rate Λ_1 (.16 (S_1 =.2) and .56 (S_1 =.7)).

As in Figures 6.7 and 6.8, five regions are delimited; three shaded regions are forbidden, one for MAPCS ($\Lambda_2 > \frac{1}{1 + (N + 1)a} - \Lambda_1$), the two others for the split channel mode ($\alpha < \Lambda_2$, $1 - \alpha < \Lambda_1$). In the two remaining regions MAPCS and the split channel mode are comparable. These two last regions are delimited by the contour α_0 versus Λ_2. Above the contour, if we split the channel, we get lower delays at the large user than with the mixed mode; below the contour, MAPCS provides lower delay than that obtained by splitting the channel.

Comparing MAPCS to a "split channel" mode, we observe different trends than the ones we observed when comparing MACS (large user and a large number of small users, see Section 6.1.4), namely

a) by splitting the channel, one may achieve a higher

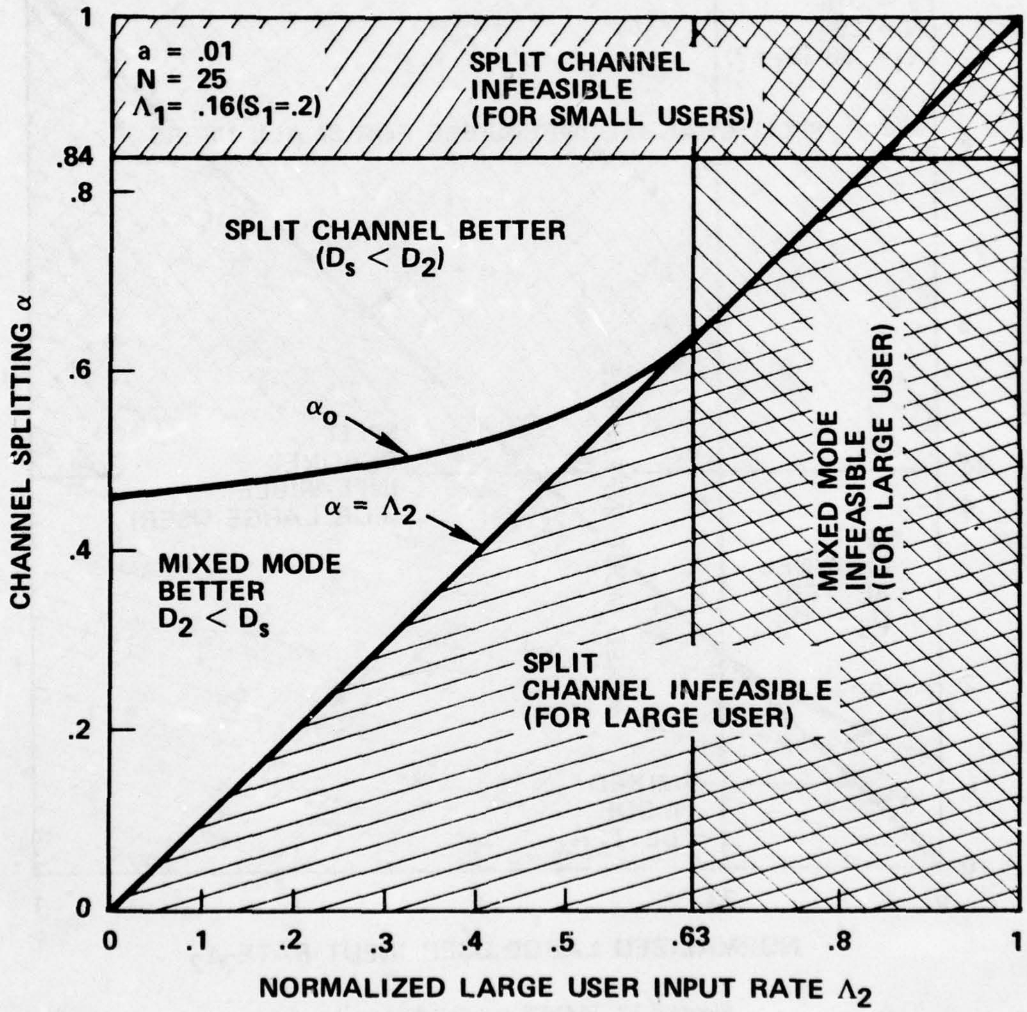


Figure 6.11. MAPCS and Split Channel : α vs Λ_2.

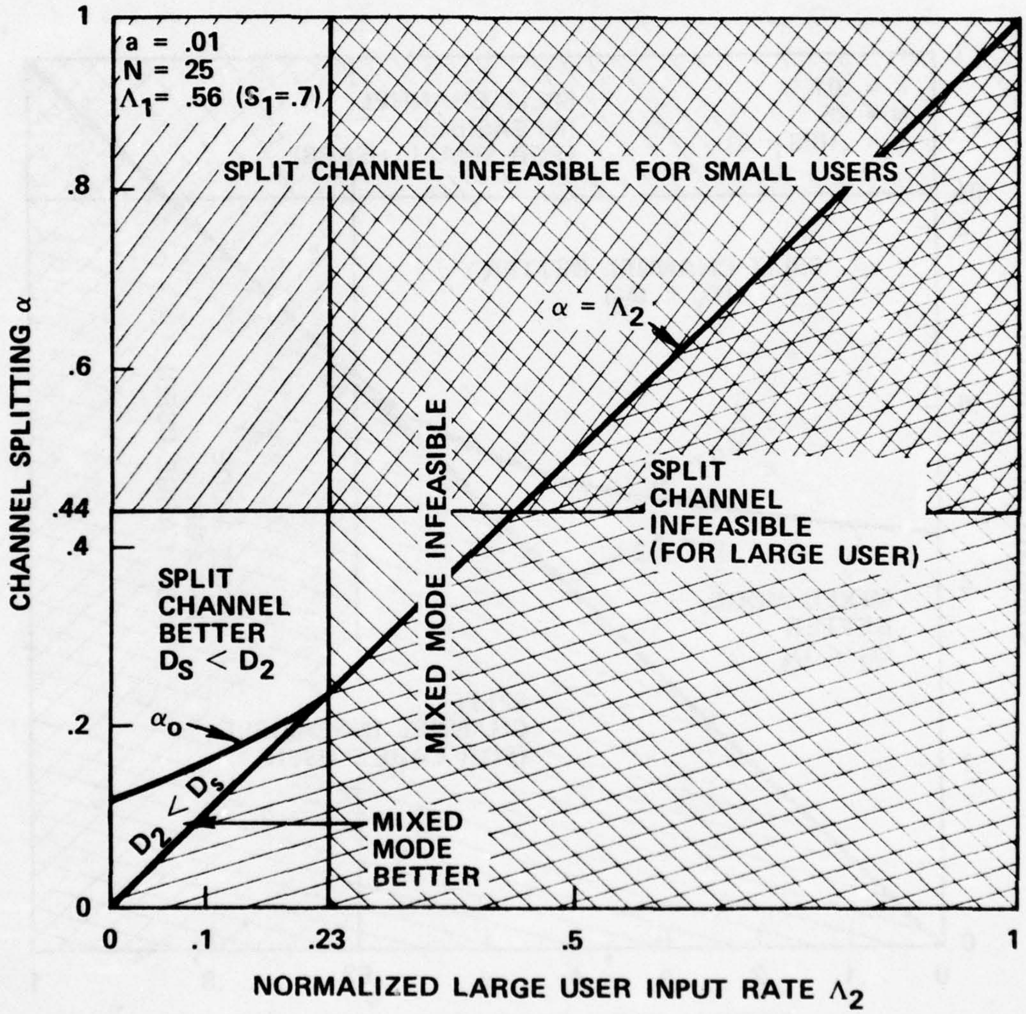


Figure 6.12. MAPCS and Split Channel : α vs Λ_2.

section).

the small users' performance. The maximum achievable throughput Λ_2 depends on Λ_1 and N: If N = 25 and $\Lambda_1 = .16$ ($S_1 = .2$) (Figure 6.11), then $\Lambda_2 \leq .84$ (instead of .63 with MAPCS). But to achieve a limiting throughput of .84, we have to split the channel so that $\alpha \geq .84$, increasing then the delays at the small users by a factor of approximately 5.

b) The choice of the access mode (MAPCS or split channel) will depend on the small users' traffic. If the latter is high, splitting the channel will not significantly degrade the small users' delay: If $\Lambda_1 = .56$ ($S_1 = .7$) (Figure 6.12), by choosing $\alpha = .4$ we approximately double the delay for the small users; we have a better delay for the large user for most values of Λ_2 and we may achieve a large user throughput of .44 (instead of .23 with MAPCS). If the small users' traffic Λ_1 is small (Figure 6.11), splitting the channel degrades the small users' performance quite significantly.

Therefore, compared with a split channel mode, MACS and MAPCS differ in the following respects:

1) For N = 25, MAPCS achieves a smaller channel utilization than the split mode*, while MACS achieves a larger channel utilization than the split mode.

2) The larger N is, the smaller is the maximum achievable throughput with MAPCS. The large user throughput is greater with MACS than with MAPCS if N is not too small ($N \geq 30$) and $\Lambda_1 < 1/e$ (see next

*The larger N is, the smaller is the channel utilization with MAPCS.

section).

3) When N is not too large (N < 171), the small users maximum throughput is greater with MAPCS than with MACS $\left(\frac{1}{1 + (N + 1)a} > \frac{1}{e}\right)$ at the expense of a smaller throughput at the large user.

4) The large user's delay is lower by splitting the channel than with MAPCS, but is higher than with MACS.

Let us compare precisely MACS and MAPCS in terms of total channel utilization and packet delay at the large user.

6.3 MACS versus MAPCS

We first compare the throughput characteristics under both schemes. In Figure 6.13 the total maximum achievable throughput Λ, normalized with respect to $T = b_m/W$, and the maximum achievable throughput of the large user Λ_2, are plotted versus the small user's normalized input rate Λ_1. For MACS we plot Λ versus Λ_1, and Λ_2 versus Λ_1 for all values of G ($G = 0$ to ∞), as given by Eqs. (6.4) and (6.1) where $S = \Lambda(1 + 2a)$ and $S_2 = \Lambda_2(1 + 2a)$. As discussed in Section 6.1, only values of $G \leq 1$ will be considered as an approximation for a stable channel (a channel which never drifts into saturation). With MAPCS, Λ is equal to $\frac{1}{1 + (N + 1)a}$ (Eq. (6.18)) for all values of the small users' input rate, and the maximum achievable throughput at the large user is $\Lambda_2 = \frac{1}{1 + (N + 1)a} - \Lambda_1$. Λ and Λ_2 are plotted for two values of the number of small users: N = 10 and N = 50. When N is large (N = 50), MACS achieves a higher throughput than MAPCS. When N is small (N = 10), this is true only for $\Lambda_1 < .32$. However with MACS one can only achieve a small users' throughput of less than or equal to 1/e; for $\Lambda_1 > 1/e$

AD-A039 018 CALIFORNIA UNIV LOS ANGELES DEPT OF COMPUTER SCIENCE F/G 9/2

ADVANCED TELEPROCESSING SYSTEMS.(U)

DEC 76 L KLEINROCK DAHC15-73-C-0368

UNCLASSIFIED NL

3 OF 4

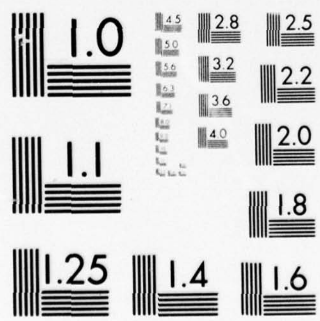
AD A039018

END

CONT.

DATE FILMED 5-77

390



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

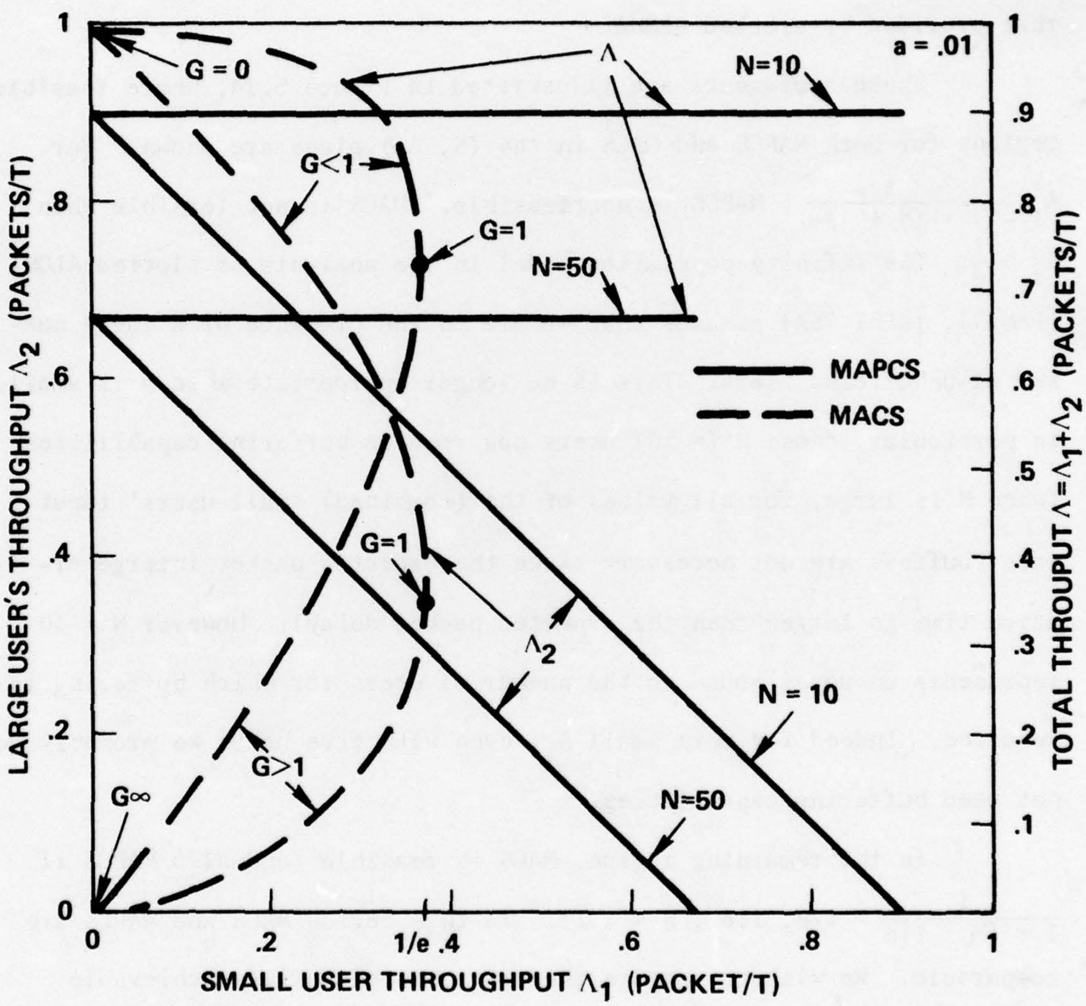


Figure 6.13. *Large Users Throughput Λ_2 and Total Throughput Λ versus Small User's Throughput Λ_1.*

only MAPCS is feasible. But if N is very large N > 171, $\frac{1}{1 + (N + 1)a} < \frac{1}{e}$ then the maximum achievable throughput of the small users, as well as the capacity of the channel under MAPCS, are less than that provided by slotted ALOHA.

These statements are illustrated in Figure 6.14, where feasible regions for both MAPCS and MACS in the (N, Λ_1) plane are shown. For $\Lambda_1 > \frac{1}{1 + (N + 1)a}$, MAPCS is not feasible. MACS is not feasible when $\Lambda_1 > \frac{1}{e}$. The infinite population model in the analysis of slotted ALOHA [LAM74], [KLEI 75A] assumes that we are in the presence of a large number of unbuffered users. This is no longer appropriate when N is small. In particular, those N (< 10) users now require buffering capabilities (when N is large, for all values of the (combined) small users' input rate, buffers are not necessary since the expected packet intergeneration time is larger than the expected packet delay). However N = 10 represents un upper bound to the number of users for which buffering is required. Indeed for very small Λ_1, even with five users we probably do not need buffering capabilities.

In the remaining region, MACS is feasible (and also MAPCS if $\frac{1}{1 + (N + 1)a} > 1/e$, i.e., N < 171). In this region MACS and MAPCS are comparable. We wish to compare them in terms of maximum achievable throughput at the large user for Λ_1 given, and therefore in terms of total channel throughput for Λ_1 given.

By solving the following system in N, we obtain the minimum value N_{crit} of the number of users, beyond which MACS provides higher throughput than MAPCS:

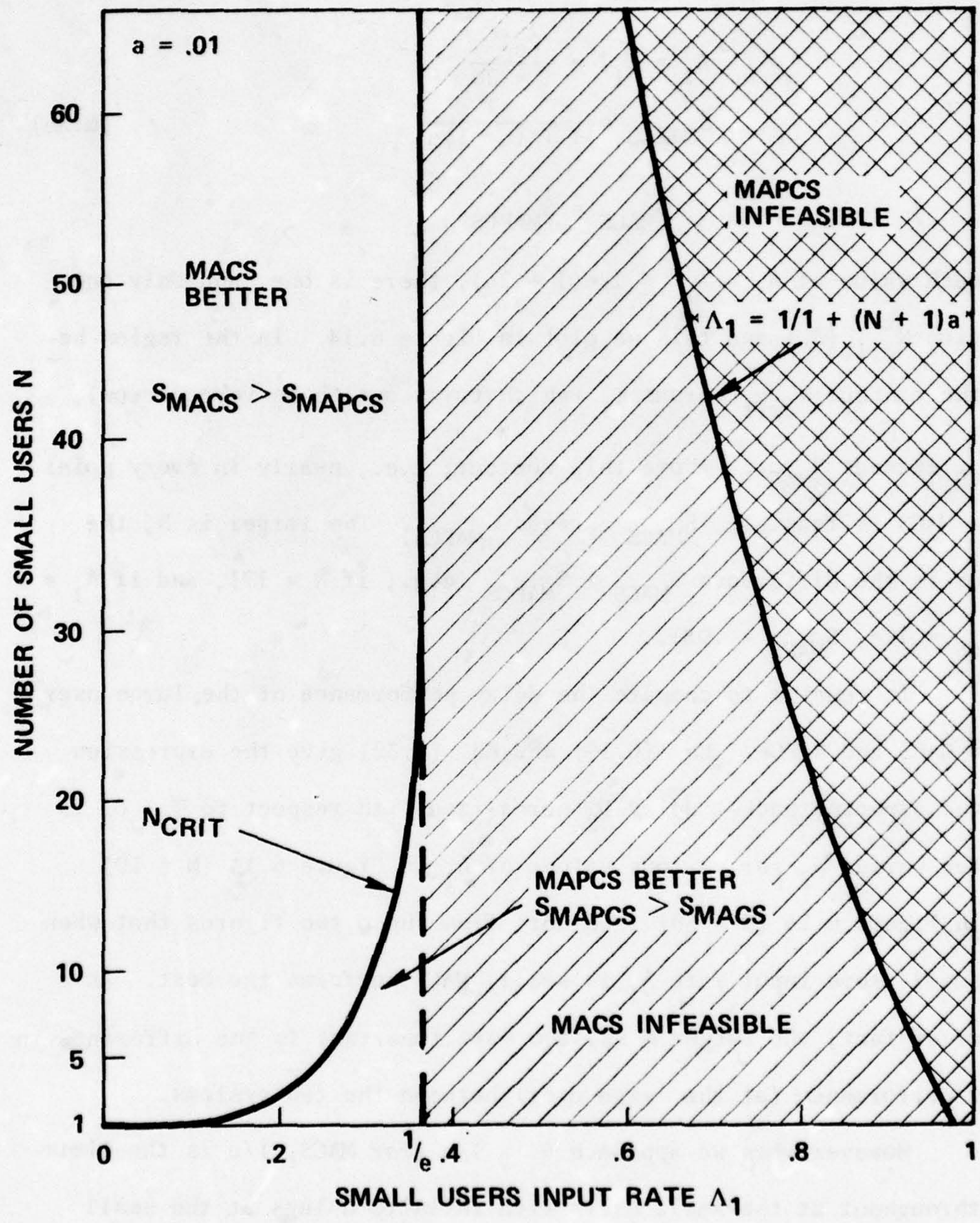


Figure 6.14. MACS and MAPCS: N vs Λ_1.

$$S_{MACS} = (1 + G)e^{-G}/1 + 2a$$

$$\Lambda_1 = \lambda_1 T = \frac{Ge^{-G}}{1 + 2a}$$

$$S_{MAPCS} = \frac{1}{1 + (N + 1)a} \tag{6.36}$$

$$S_{MACS} = S_{MAPCS}$$

For each value of $\Lambda_1 = \lambda_1 T \leq 1/e(1 + 2a)$, there is one, and only one solution $N_{crit}(\Lambda_1)$ and this we plot in Figure 6.14. In the region below the contour N_{crit} versus Λ_1 (which turns out to be very narrow), S_{MAPCS} exceeds S_{MACS}. Above this contour, i.e., nearly in every point where MACS is feasible, S_{MACS} exceeds S_{MAPCS}. The larger is N, the larger is the difference $S_{MACS} - S_{MAPCS}$ (e.g., if N = 171, and if $\Lambda_1 = 0$ $S_{MAPCS} = 1/e$, $S_{MACS} = .98$).

It remains to compare the delay performance of the large user under MACS and MAPCS. Eq. (6.15) and Eq. (6.32) give the expression for the expected packet delay D_2 normalized with respect to T. D_2 is plotted versus Λ_2 for various values of Λ_1 in Figure 6.15 (N = 10) and in Figure 6.16 (N = 50). We note from these two figures that when the small users input rate Λ_1 is small, MACS performs the best. As a matter of fact, the larger N is, the more important is the difference in delay performance (at the large user) between the two systems.

However when we approach $\Lambda_1 = 1/e$ (for MACS, 1/e is the limiting throughput at the small users with infinite delays at the small users), the delay at the large user increases very fast in MACS. Thus for values of Λ_1 around 1/e, MAPCS provides lower delays at the large user.

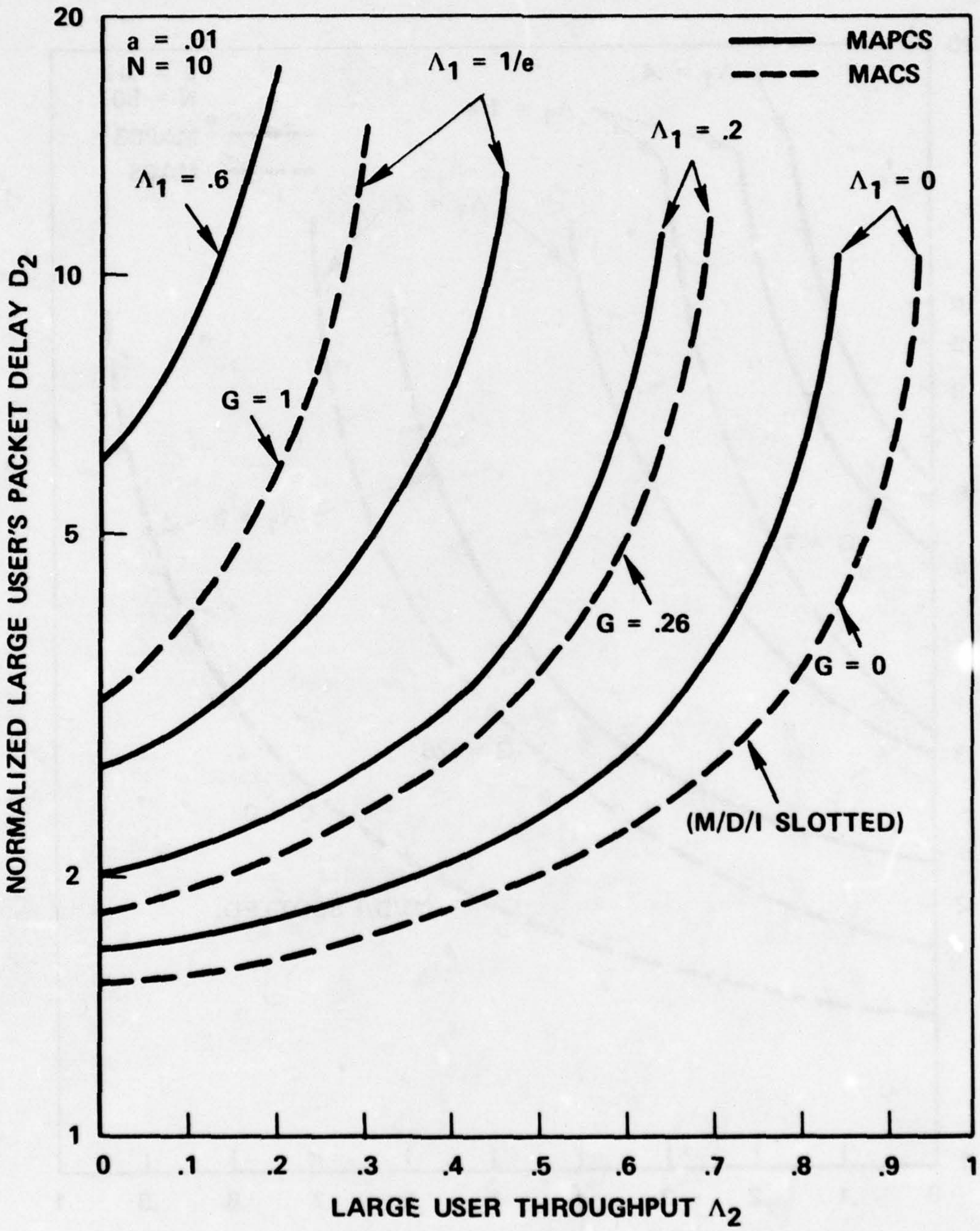


Figure 6.15. Effect of Small User's Traffic on Large User's Throughput Delay Performance D_2 vs Λ_2.

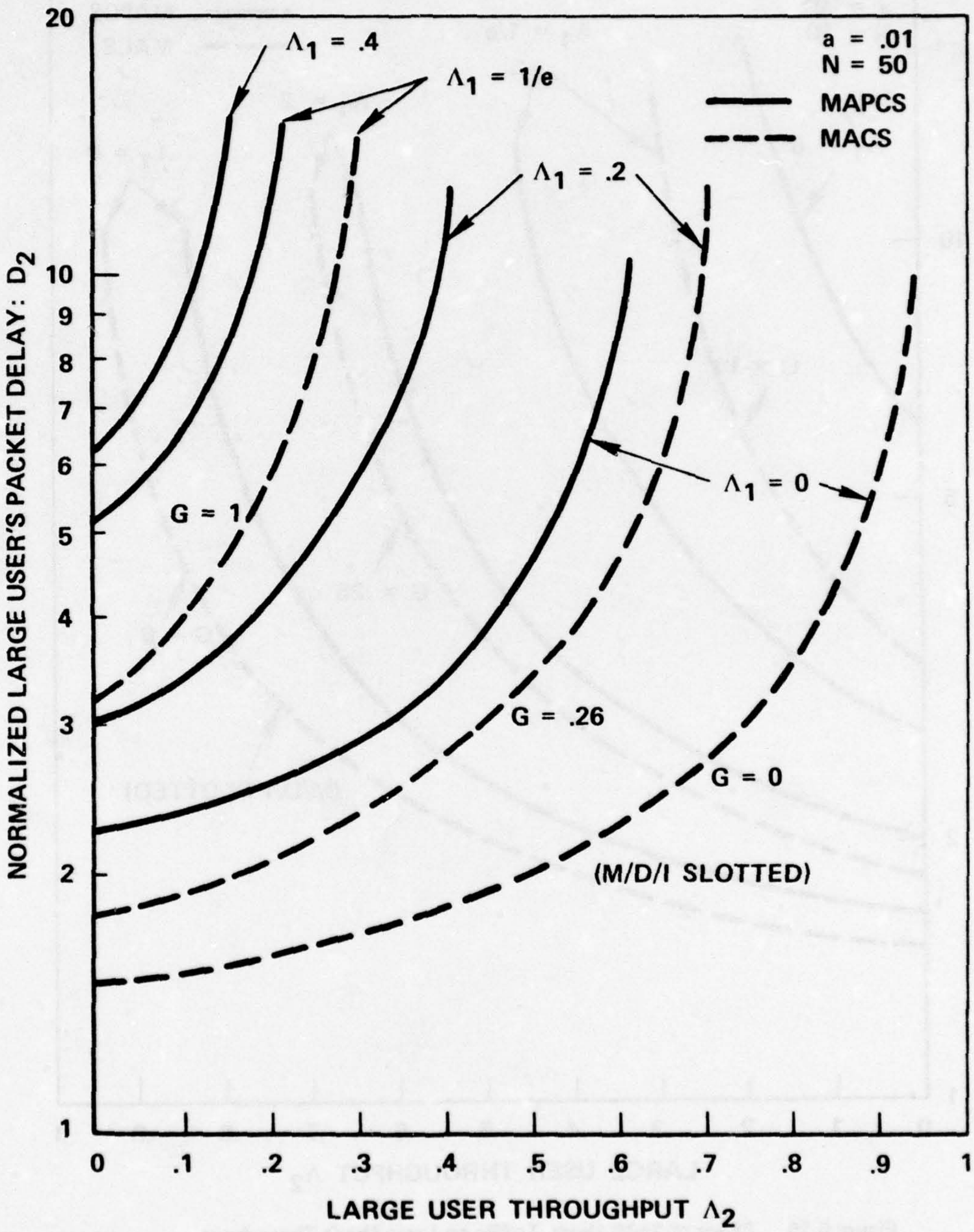


Figure 6.16. Effect of Small Users Traffic on Large User's Throughput Delay Performance : D_2 vs Λ_2.

In conclusion, the Mixed Alternating Priorities Carrier Sense mode (MAPCS) is particularly appropriate when the background of small users is not large in number ($N \leq 10$). MAPCS allows high input rates at the small users, but requires that all small users be in line of sight and within range of each other.

The Mixed ALOHA Carrier Sense mode (MACS) does not allow input rates in the background exceeding 1/e, but does not require that all users hear each other. MACS is particularly suitable for a background composed of a large number of bursty users and performs better than MAPCS.

CHAPTER 7

CONCLUSIONS AND SUGGESTIONS FOR FUTURE RESEARCH

7.1 Conclusions

Packet-switched ground radio systems provide attractive solutions to the interconnection between (potentially mobile) terminals scattered over wide geographical areas and packet-switched computer communication networks. New packet-switched radio techniques have been introduced in the past few years as an alternative to the conventional wire communication techniques which have been adapted to a radio environment. These new techniques, called random access techniques (e.g., ALOHA, CSMA) provide the multiple access of a broadcast channel to a large number of users. Measurement studies [JACK 69] conducted on time-sharing systems indicate that both computer and terminal data streams are bursty. Consequently, if a high-speed channel is used for the communication between two points (terminal and computer), the channel utilization is low since the channel is idle most of the time. On the other hand, if a low-speed point-to-point channel is used, the delay is large. As a result, random access techniques have been introduced in which the channel is dynamically shared among all users and the required channel capacity is much less than in the unshared case of dedicated channels. This concept has also been applied to conventional schemes (e.g., Polling). However, random access techniques present the advantage of not requiring the control from a master user, while conventional techniques (such as Polling) do. This control from a central station (e.g., computer) is not desirable for many reasons including reliability, difficulty in mobile environments, cost in overhead, or complexity in point-to-point communications.

When we are in the presence of a small number of users requiring buffering space for more than one packet, random access techniques are not necessarily optimal. New multiple access techniques are needed.

The first objective of this research was to provide the communication system designer with new access modes to the radio channel. These access modes are suitable for a small number of users and do not require control from a central station. Alternating Priorities (AP), Round Robin (RR) and Random Order (RO) were introduced as new multiplexing techniques. Their performance was shown to be heavily affected by the number of users N, and the major assumption was made that all users are in line of sight and within range of each other. When $N < 20$, the capacity of the channel is larger under those new modes than that obtained with CSMA, and they provide a delay performance comparable to that of Polling. However the performance of AP, RR and RO degrades badly when N increases. A modification to AP, RR and RO was introduced which reduces this performance degradation when N is not too large ($N \leq 50$).

A good solution to the problem is provided by a new multiple access scheme, Mini-Slotted Alternating Priorities (MSAP). MSAP was shown to achieve a channel capacity of one and to perform better than Polling. Compared to random access techniques (CSMA), MSAP provides better delays under heavy traffic conditions. At light traffic, when the number of users is very large (N > 100), CSMA provides much better delays. For $N \leq 100$, the packet delay under MSAP is slightly larger than under CSMA. MSAP assumes that all users are within range and in line of sight, but presents the following advantages:

i) It is suitable for all values of the number of users N.

ii) It allows buffering capability at each user.

iii) It does not require the control from a master user.

iv) Its performance is optimal when the number of users is large, except at light traffic, in which case random access techniques perform better.

AP, RR, RO and MSAP have been introduced as new methods of multiplexing a homogeneous population of users.

The second objective of this dissertation was to introduce new methods of multiplexing a buffered large user and a population of small users on the same radio channel. The communication channel is to be shared by two independent sources of traffic (a large user and a population of small users); the simplest solution is to assign a dedicated channel to each source. However, by dynamically sharing the channel in some fashion among the two sources, the required channel capacity may be much less than in the unshared case of dedicated channels, especially if one (or both) source(s) is (are) bursty.

Three examples of application are considered:

(1) One file user and a collection of interactive users.

(2) A repeater* (or central station) and terminals

*In wide geographical areas, repeaters are required to extend the range of terminals and computers.

(3) A computer and repeaters.

The Mixed ALOHA Carrier Sense (MACS) access mode was introduced as a method of multiplexing a large population of bursty (small) users and a large user within range and in line of sight of all small users (examples (1) and (2) above). The small users contend for the channel in a slotted ALOHA fashion. The slotted ALOHA access mode [KLEI 75A] is an efficient random access method of multiplexing a large number of bursty users and presents the advantage of being very simple to implement. But the channel utilization with slotted ALOHA is very limited (1/e). By providing the total available bandwidth to the small users, we increase the achievable throughput at the small users and decrease their packet delay. With MACS, the large user "steals" a large part of the wasted capacity (which is 1 - 1/e in slotted ALOHA). We have shown that not only is the small users' performance improved, but also the performance of the large user is better with MACS than it is when dedicated channels are assigned to the large user and the small users.

If the population of small users is composed of a few buffered users (example (3) above), we may apply the same idea. This was done in the Mixed Alternating Priorities Carrier Sense (MAPCS) access mode. In MACPS, the small users share the total available bandwidth under the AP protocol. The large user steals the slots unused by the small users. MACPS provides a very good performance to the small users. However the large user's performance is much affected by the presence of the small users. MAPCS allows a high input rate at the small users, but this high input rate at the small users implies a small throughput and high delays at the large user.

7.2 Extensions to this Research

7.2.1 Extension to MSAP and Polling

MSAP has been shown to provide the best performance of all existing ground radio multiple access techniques for a homogeneous population of (possibly) buffered users within range and in line of sight of each other, under heavy traffic conditions.

At light traffic, if the number of users is large ($N \leq 50$), random access techniques provide a better delay then MSAP or Polling. We make the following suggestion to reduce the delay at light traffic under MSAP and Polling:

The N users are split into k disjoint groups ($k \geq 1$), each of them composed of $h = \left\lfloor \frac{N}{k} \right\rfloor$ users*. After M ($< N$) unsuccessful attempts to find a busy user under the original protocol (MSAP or Polling), a group of users contend for the channel in a slotted ALOHA fashion.

Either (a) One, and only one, user among the h users of the group has a packet to transmit. This packet is successfuly transmitted. At the end of the transmission, operate as in the original protocol, or

(b) More than one user in the group transmit a packet; their packet transmissions collide. At the end of transmission, the h users are scanned one by one as in the original protocol, or

(c) No busy user is found. Then (one minislot later in MSAP, r (> 1) minislots later in Polling) another group of h users contend for the channel in a slotted ALOHA fashion and we

* except one group which is composed of $N - (k - 1) \left\lfloor \frac{N}{k} \right\rfloor$

operate as above.

7.2.2 Extension to MACS and MALCS

MACS and MALCS are two techniques of multiplexing a large user and a population of small users on the same radio channel. We suggest to extend the same idea to various types of small user populations. As an example, we might consider a population of a a small number of bursty users who do not require a fast response time. We might then multiplex these users in TDMA and allow the large user to steal the slots assigned to an idle user.

Another extension would be to study the inclusion on the same channel of the traffic of an infinite population of small users as well as the traffic of a finite number (> 1) of large users.

7.2.3 Packet Radio Networks

Throughout this dissertation we considered single-hop systems: a number of users communicating with each other or with a central station in line of sight and within range of all users. In wide geographical areas, repeaters are required to extend the range of terminals and computers, thus giving rise to multihop networks, where a packet must be stored and forwarded by at least one node (repeater) when being transmitted from source (terminal) to destination (terminal or computer).

The design of packet radio networks involves a wide range of unsolved problems, e.g., routing, behavior and performance evaluation of existing multiple access schemes for various configurations, characterization of the traffic burstiness, etc. We make the following suggestion concerning the choice of access modes:

For various configurations, systems where the channel is shared among all users according to a common access scheme (e.g., ALOHA, CSMA) ought to be compared to dedicated channel systems. In the latter, dedicated channels are assigned (possibly dynamically) to clusters of users (repeaters) in the network. Within each cluster, various schemes may be chosen. We suggest a study of the decomposition of the network in clusters, such that:

- In each cluster, a particular node forwards all traffic from the cluster's users toward another cluster to which it also belongs.
- Each cluster is assigned a dedicated channel. The allocation of channel bandwidth must take advantage of the following considerations:
  - Two "out of range" clusters may use the same bandwidth without any packet interference.
  - The traffic of a cluster should be as "unbursty" as possible. When two neighbor clusters are bursty, it may be preferable to make them share the same bandwidth.
- A cluster could be modelled as a population composed of
  - One (or more) large user(s): the repeater(s) forwarding the cluster's traffic toward neighbor clusters.
  - A population of small users: the repeaters (terminals) of the cluster.

Then, we might consider MACS, MAPCS, or other similar schemes, the study of which has been suggested in Section 7.2.2, as methods for multiplexing the users of a cluster.

The optimal decomposition of the network into clusters depends on various parameters, such as the network topological configuration, the traffic burstiness, etc.

7.2.4 Application to Distributed Packet Switching for Local (wire) Computer Networks.

The schemes introduced and analyzed in this dissertation are applicable to distributed local (wire) Computer Networks. As an example, we consider the operating Ethernet (see [METC 76] and Chapter 1) experimented along a kilometer of coaxial cable with a speed of 3 Megabits/s. All the system assumptions defined in Chapter 2 are applicable to the Ethernet environment. In particular, there is no multipath effect (Assumption 4). Indeed there is one and only one path for any source-destination pair (tree structured topology). All stations are within range of each other (Assumption 7). The larger the packet size is, the better is Assumption 6. Indeed for packet sizes of 1024 bits and 4096 bits, the ratio of the maximum propagation delay between any source-destination pair over the packet transmission time is respectively $a = .02$ and $a = .005$. The delay-throughput performance of AP and MSAP ought to be compared to that obtained with the scheme currently implemented in the Ethernet [METC 76]. In any case, AP and MSAP are suitable to such a network in which control is distributed among stations. In addition, the schemes introduced in Chapter 6 (MACS and MAPCS) are suitable for the connection between separate local networks. Indeed, consider two Ethernet like networks connected through a gateway (large user) which stores and forwards all packets the source and destination of which do not belong to the same network (and therefore are not within range of each other). This gateway's traffic competes on each of the network with the stations local traffic (small users).

The introduction and implementation of packet radio techniques is likely to have a large impact on the data communication field. Studies such as this one are important in that they contribute to those techniques.

APPENDIX A

THE QUEUE M/G/1 WITH REST PERIOD AND FIRST-COME-FIRST-SERVED (FCFS) ORDER OF SERVICE

The queue M/G/1 with rest period and FCFS order of service has been studied by Miller [MILL 64]. Using a simpler approach we solve for the z-transform of the distribution function of the number in system in such a queueing system. The method we use is based on the "imbedded Markov chain" approach introduced by Kendall [KEND 51] and used extensively in queueing theory [KLEI 75C]. We then give some results concerning the delay and the busy period duration in such a queueing system. These results were first obtained by Miller [MILL 64].

A.1 The Model

Customer arrival instants are generated by a Poisson point process with intensity λ. Each customer requires from the single server a random amount of service time, $\tilde{x}$. The service times are identically distributed with Laplace-Stieljes transform (LST), $B^*(s)$, and are independent of each other and of the arrival process. When the server completes a customer's service, he will select the next one in queue, if any, according to a FCFS discipline and begin to serve him immediately. If there are no more customers in queue waiting for service so that the server goes idle for lack of work, he will be withdrawn from the system for some time (which we call a rest period), $\tilde{T}_0$ with

LST $R_0^*(s)$. At the end of the rest period, the server will return and begin to serve the customers that have accumulated during the rest period. If there is no backlog, he will take another rest period which starts immediately. The rest periods are independent of each other and of the arrival and service processes.

The main derivation is that of the z-transform Q(z) of the distribution function of the number in system $\tilde{q}$ just after customer departure instants. It is shown in [MILL 64] that the solution at these imbedded Markov points provides the solution for <u>all</u> points in time, under steady state conditions.

A.2 z-Transform of the Distribution Function of the Number of Customers in System just after the Departure Instants

Theorem:

The z-transform of the distribution function of the number of customers just after the departure instants (and therefore that of the number of customers in system at all points in time) is given by

$$Q(z) = \frac{B^*(\lambda - \lambda z)}{\lambda E(T_0)} \frac{(1 - \rho)(1 - R_0^*(\lambda - \lambda z))}{B^*(\lambda - \lambda z) - z} \tag{A.1}$$

where, as in a regular M/G/1 queue (without rest period):

$$\rho = \lambda E(x) \tag{A.2}$$

In addition,

$$P[\text{system is idle}] \triangleq p_0 = \frac{(1 - \rho)(1 - R_0^*(\lambda))}{\lambda E(T_0)} \tag{A.3}$$

Eq. (A.1) may be rewritten as

$$Q(z) = Q_{M/G/1}(z) \times V_0(z) \qquad (A.4)$$

where $Q_{M/G/1}(z)$ represents the z-transform of the distribution function of the number of customers in a regular M/G/1 queue, and is given by (see [KLEI 75C]):

$$Q_{M/G/1}(z) = B^*(\lambda - \lambda z) \frac{(1 - \rho)(1 - z)}{B^*(\lambda - \lambda z) - z}$$

and $V_0(z)$ represents the z-transform of the distribution function of the number of arrivals during the residual life of a rest period and is given by

$$V_0(z) = C_0^*(\lambda - \lambda z) = \frac{(1 - R_0^*(\lambda - \lambda z))}{E(T_0)(\lambda - \lambda z)} \qquad (A.5)$$

where $C_0^*(s)$ is the LST of the distribution function of the residual life of a rest period.

Indeed, given any time interval $\tilde{t}$ with LST of the distribution function denoted by $T^*(s)$, it is easy to show (see for example [KLEI 75C] p. 197) that the z-transform $V(z)$ of the distribution function of the number of (Poisson) arrivals during this time interval t is given by

$$V(z) = T^*(\lambda - \lambda z) \qquad (A.6)$$

Observing that the LST, $C_0^*(s)$ of the distribution function of the residual life of a rest period is given by

$$C_0^*(s) = \frac{1 - R_0^*(s)}{sE(T_0)} \qquad (A.7)$$

We then get Eq. (A.5).

Thus, the number of customers in system just after the departure instants (and therefore the number of customers in system at all points in time) is the convolution of the two following independent random variables:

number in system in a regular M/G/1 queue

number of arrivals during a time interval distributed as the residual life of the rest period.

This result is not obvious and could not be proven directly.

In addition, by taking the first derivative of the right-hand side (RHS) of Eq. (A.4) at z = 1, we obtain the expected number of customers in system E(q) just after the departure instants (and therefore the expected number of customers at all points in time)

$$E(q) = Q^{(1)}(1) = \lambda E(x) + \frac{\lambda^2 E(x^2)}{2(1 - \rho)} + \frac{\lambda E(T_0^2)}{2E(T_0)} \qquad (A.8)$$

The sum of the first two terms of the RHS of Eq. (A.8) represents the expected number in system $E(q_{M/G/1})$ in a regular M/G/1 queue (P-K formula, see [KLEI 75C], p. 187).

<u>Proof</u>:

Let us define the following sequences of random variables related to C_n, the n^{th} customer entering the system at time t_n:

x_n: service time for C_n

q_n: number of customers left behind by the departure of C_n

v_n: number of customers arriving during the service time of C_n

w_n: number of customers arriving during a rest period if C_n finds the system empty (C_n is the first customer arriving during this rest period); 0, otherwise

We choose q_n as our state variable: We proceed to solve for the system behavior just after the departure instants. These instants in time form an imbedded Markov chain. The solution at these points will also provide the solution for all points in time [MILL 64].

Our first step is to find an equation relating the random variable q_{n+1} to the random variable q_n by considering two cases. The first is shown in Figure A.1 and corresponds to the case where C_n leaves behind a non-empty system (i.e., $q_n > 0$; C_{n+1} is already in the system when C_n departs). Clearly we have

$$q_{n+1} = q_n - 1 + v_{n+1} \qquad q_n > 0 \tag{A.9}$$

Indeed, the number of customers left behind by the departure of C_{n+1} is equal to the number of customers left behind by the departure minus one (departure of C_{n+1}) plus the number of customers that arrive during the service interval x_{n+1}.

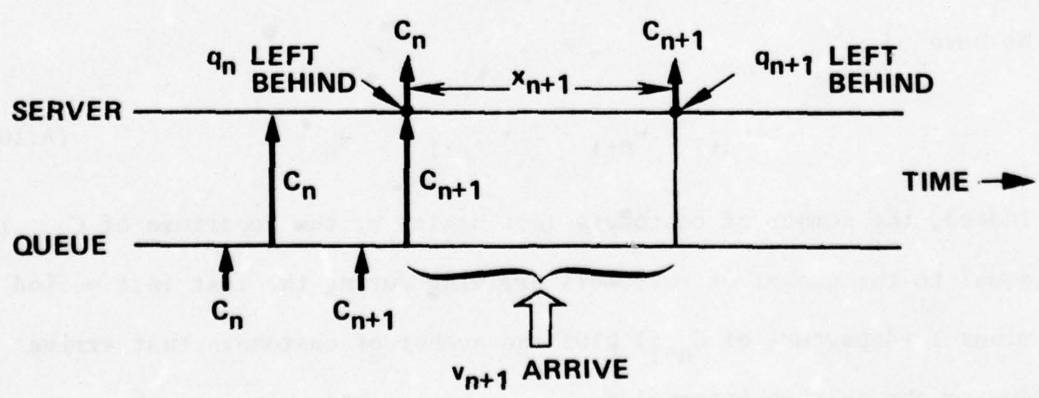


Figure A.1. Case where $q_n > 0$.

Now consider the case where $q_n = 0$ (i.e., C_n leaves behind an empty system). An example of this case is illustrated in Figure A.2. At the departure of C_n, the server takes a first rest period. When he returns, there is no backlog (C_{n+1} has not yet arrived by the time the server returns). The server takes a second rest period, during wh ch C_{n+1} arrives. At the end of this rest period, the server begins to serve C_{n+1}.

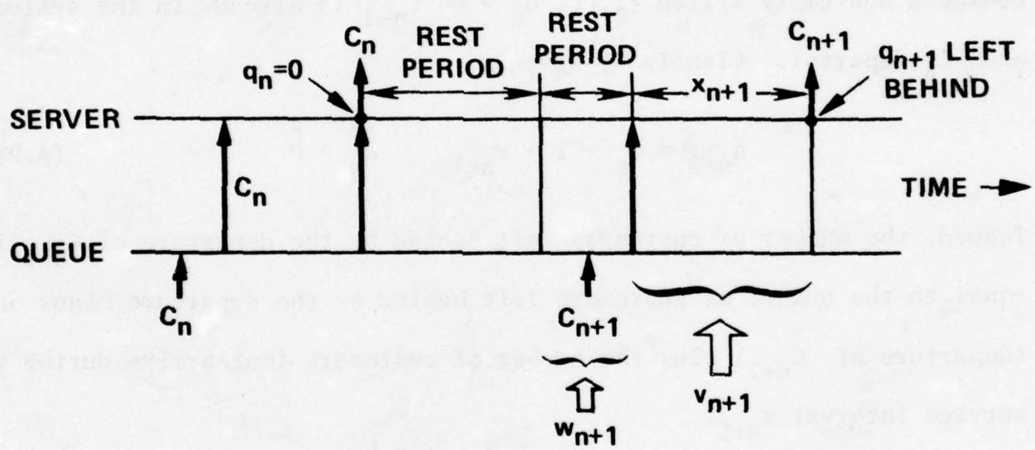


Figure A.2. Case where $q_n = 0$.

We have

$$q_{n+1} = w_{n+1} - 1 + v_{n+1} \qquad q_n = 0 \tag{A.10}$$

Indeed, the number of customers left behind by the departure of C_{n+1} is equal to the number of customers arriving during the last rest period minus 1 (departure of C_{n+1}) plus the number of customers that arrive during the service interval x_{n+1}.

Collecting together Eq. (A.9) and Eq. (A.10), we may now write the single equation relating q_{n+1} to q_n as

$$q_{n+1} = q_n - 1 + v_{n+1} + w_{n+1} \tag{A.11}$$

Let us define

$$\tilde{q} = \lim_{n\to\infty} q_n$$

$$E(\tilde{q}^j) = \lim_{n\to\infty} E(\tilde{q}_n^j)$$

These limits exist if the imbedded Markov chain is ergodic. Assuming ergodicity and forming the expectation of both sides of Eq. (A.11), we have in the limit as $n \to \infty$

$$E(\tilde{q}) = E(\tilde{q}) - 1 + E(\tilde{v}) + E(\tilde{y})$$

which yields

$$E(\tilde{y}) = 1 - E(\tilde{v}) \tag{A.12}$$

$E(\tilde{v})$, which is the average number of arrivals during a service time is given by

$$E(\tilde{v}) = \lambda E(\tilde{x}) \overset{\Delta}{=} \rho \tag{A.13}$$

$E(\tilde{y})$ is the expected value of the following random variable:

$$\tilde{y} \begin{cases} \text{number of customers arriving during a rest period,} \\ \text{given there is at least one arrival during this} \\ \text{rest period, if the last departure left behind an} \\ \text{empty system} \\ \\ 0, \text{ otherwise} \end{cases}$$

Clearly, we have

$$P[\text{there is at least one arrival during a rest period}] = 1 - R_0^*(\lambda)$$

and

$$E[\text{number of arrivals during a rest period}] = \lambda E(T_0)$$

Then

$$E(\tilde{y}) = P[\tilde{q} = 0] \frac{\lambda E(T_0)}{1 - R_0^*(\lambda)} + 0 \times (1 - P[\tilde{q} = 0])$$

Denoting $P[\tilde{q} = 0]$ by p_o, we have

$$E(\tilde{y}) = p_0 \frac{\lambda E(T_0)}{1 - R_0^*(\lambda)} \tag{A.14}$$

Substituting Eq. (A.13) and Eq. (A.14) in Eq. (A.12), we obtain p_o which is also the probability that the system is idle (empty):

$$p_o = \frac{(1 - \rho)}{\lambda E(T_0)} (1 - R_0^*(\lambda)) \tag{A.15}$$

Eq. (A.15) is precisely Eq. (A.3).

For stability we, of course, require $p_0 > 0$ or equivalently as in a regular M/G/1 queue

$$\rho = \lambda E(x) < 1 \tag{A.16}$$

We now proceed to calculate the z-transform $Q(z)$ of the distribution function of the random variable $\tilde{q}$:

$$Q(z) = E[z^{\tilde{q}}] = \lim_{n\to\infty} Q_n(z) = \lim_{n\to\infty} E[z^{\tilde{q}_n}]$$

From Eq. (A.11) we have

$$E[z^{q_{n+1}}] = E[z^{q_n - 1 + v_{n+1} + w_{n+1}}]$$

Since v_{n+1} is independent of q_n and also is independent of the subscript $(n+1)$ we may write

$$E[z^{q_{n+1}}] = \frac{E[z^v]}{z} E[z^{q_n + w_{n+1}}] \tag{A.17}$$

By the same argument as above (Eq. (A.6)), it is easy to show that

$$E[z^{v}] = B^{*}(\lambda - \lambda z) \tag{A.18}$$

Let us examine the second factor of the right-hand side of Eq. (A.17):

$$E[z^{q_n + w_{n+1}}] = \begin{cases} E[z^{q_n}] & \text{if } q_n > 0 \\ E[z^{y_{n+1}}] & \text{if } q_n = 0 \end{cases}$$

where y_{n+1} is the number of arrivals during a rest period given there is at least on arrival. This random variable is independent of the subscript (n+1). Therefore

$$E[z^{y_{n+1}}] = \frac{R(z) - R(0)}{1 - R(0)}$$

where R(z) is the z-transform of the distribution function of the number of arrivals in a rest period and R(0) is the probability there is no arrival during a rest period. Using the same argument as before (Eq. (A.6)), we have

$$R(z) = R_0^{*}(\lambda - \lambda z),$$

then

$$E[z^{y_{n+1}}] = \frac{R_0^{*}(\lambda - \lambda z) - R_0^{*}(\lambda)}{1 - R_0^{*}(\lambda)}$$

Thus we have

$$E[z^{q_n + w_{n+1}}] = \sum_{i=1}^{\infty} p_{n,i} z^{i} + p_{n,0} E[z^{y_{n+1}}]$$

where $p_{n,i}$, i = 0, 1, 2, ... denotes $P[\tilde{q}_n = i]$. We finally obtain

$$E[z^{q_n+w_{n+1}}] = Q_n(z) - p_{n,0} + p_{n,0} \frac{R_0^*(\lambda - \lambda z) - R_0^*(\lambda)}{1 - R_0^*(\lambda)}$$

$$E[z^{q_n+w_{n+1}}] = Q_n(z) + p_{n,0} \frac{R_0^*(\lambda - \lambda z) - 1}{1 - R_0^*(\lambda)} \quad \text{(A.19)}$$

Substituting Eq. (A.18) and Eq. (A.19) into Eq. (A.17), we have

$$Q_{n+1}(z) = \frac{B^*(\lambda - \lambda z)}{z} \left\{ Q_n(z) + p_{n,0} \frac{R_0^*(\lambda - \lambda z) - 1}{1 - R_0^*(\lambda)} \right\}$$

We now take the limit as n goes to infinity. Denoting by p_0 the limit as n goes to infinity of $p_{n,0}$, we have in the limit

$$Q(z) = \frac{B^*(\lambda - \lambda z)}{z} \left\{ Q(z) + p_0 \frac{R_0^*(\lambda - \lambda z) - 1}{1 - R_0^*(\lambda)} \right\}$$

$$Q(z) = p_0 \frac{B^*(\lambda - \lambda z)}{B^*(\lambda - \lambda z) - z} \; \frac{1 - R_0^*(\lambda - \lambda z)}{1 - R_0^*(\lambda)} \quad \text{(A.20)}$$

Since Q(1) = 1, p_0 must satisfy the following limit:

$$\lim_{z \to 1} p_0 \frac{B^*(\lambda - \lambda z)[1 - R_0^*(\lambda - \lambda z)]}{[B^*(\lambda - \lambda z) - z][1 - R_0^*(\lambda)]} = 1$$

Applying L'Hospital rule, we then have

$$p_0 = \frac{1 - \rho}{\lambda E(T_0)} (1 - R_0^*(\lambda)) \quad \text{(A.21)}$$

which is the expression obtained earlier for p_0 (Eq. (A.15)). Substituting Eq. (A.21) in Eq. (A.20), we finally have the expression for Q(z) as given by Eq. (A.1).

This equation provides the solution for all points, i.e., the z-transform of the distribution function of the number of customers in system at all instants.

We may then easily derive the LST, $S^*(s)$ of the distribution function of the delay (time in system) incurred by a customer from his arrival instant until his service completion.

A.3 LST of the Distribution Function of the Time in System

Using the same argument as before (Eq. (A.6)), we observe that $S^*(\lambda - \lambda z)$ is the z-transform of the distribution function of the number of customers arriving during the time spent in system by any customer (waiting time plus service time). But this number of customers is precisely the number of customers which arrive after a given "tagged" customer and left behind after his departure. Therefore we may write [KLEI 75C]

$$Q(z) = S^*(\lambda - \lambda z) \tag{A.22}$$

By making the change of variable

$$s = \lambda - \lambda z$$

which gives

$$z = 1 - \frac{s}{\lambda}$$

and by substituting Eq. (A.22) in Eq. (A.1) we then have

$$S^*(s) = \frac{B^*(s)}{E(T_0)} \frac{(1 - \rho)(1 - R_0^*(s))}{s - \lambda + \lambda B^*(s)} \tag{A.23}$$

which is the LST of the distribution function of time in system. Eq. (A.23) may be rewritten as

$$S^*(s) = \frac{(1 - \rho)sB^*(s)}{s - \lambda + \lambda B^*(s)} \frac{1 - R_0^*(s)}{sE(T_0)} \tag{A.24}$$

The first factor of Eq. (A.24) is the LST of the distribution function of time in system in a regular M/G/1 queue (see for example [KLEI 75C]). The second factor of Eq. (A.24) is the LST of the distribution function of the residual life of a time interval distributed as the rest period (see Eq. (A.7)).

$$S^*(s) = S^*_{M/G/1}(s) \times C^*_0(s) \tag{A.25}$$

Thus the time spent in system is equal to the convolution of the two following independent random variables:

i) time a customer would spend in a regular M/G/1 queueing system (without rest period) with the same arrival and service processes, and

ii) additional delay distributed as the residual life of the rest period.

By taking the first and second derivatives at s = 0 of the right-hand side of Eq. (A.25) we obtain the first and second moments of time in system, which we denote by $E(T)$ and $E(T^2)$:

$$E(T) = E(T_{M/G/1}) + E[\text{residual life of a rest period}]$$

$$E(T) = E(x) + \frac{\lambda E(x^2)}{2(1-\rho)} + \frac{E(T_0^2)}{2E(T_0)} \tag{A.26}$$

The sum of the first two terms of the right-hand side of Eq. (A.26), $(E(T_{M/G/1}))$ represents the Pollaczek-Khinchin mean value formula [KLEI 75C]. Then the expected waiting time $E(w)$ is given by

$$E(w) = \frac{\lambda E(x^2)}{2(1-\rho)} + \frac{E(T_0^2)}{2E(T_0)} \tag{A.27}$$

The second moment $E(T^2)$ is given by

$$E(T^2) = E(T^2_{M/G/1}) + 2E(T_{M/G/1}) \frac{E(T_0^2)}{2E(T_0)} + \frac{E(T_0^3)}{3E(T_0)} \tag{A.28}$$

where $E(T^2_{M/G/1})$ is the second moment of time in system in an M/G/1 queue. Then we have

$$E(T^2_{M/G/1}) = E(w^2_{M/G/1}) + 2E(w_{M/G/1})E(x) + E(x^2) \tag{A.29}$$

where $E(w_{M/G/1})$ and $E(w^2_{M/G/1})$ are the first and second moments of waiting time in an M/G/1 queue and are given [KLEI 75C] by

$$E(w_{M/G/1}) = \frac{\lambda E(x^2)}{2(1 - \rho)} \tag{A.30}$$

$$E(w^2_{M/G/1}) = 2[E(w_{M/G/1})]^2 + \frac{\lambda E(x^3)}{3(1 - \rho)} \tag{A.31}$$

But the second moment of waiting time $E(w^2)$ is such that:

$$E(w^2) = E(T^2) - 2E(w)E(x) - E(x^2) \tag{A.32}$$

Substituting Eq. (A.28), (A.29), (A.30) and (A.31) into Eq. (A.32) we obtain after some manipulations

$$E(w^2) = \frac{\lambda E(x^2)}{(1 - \rho)} E(w) + \frac{\lambda E(x^3)}{3(1 - \rho)} + \frac{E(T_0^3)}{3E(T_0)} \tag{A.33}$$

Finally, we may observe that the variance σ^2 of time in system is equal to the variance $\sigma^2_{M/G/1}$ of time in an M/G/1 queueing system with the same arrival and service processes (without rest period) plus the variance of the residual life of a rest period

$$\sigma^2 = \sigma^2_{M/G/1} + \sigma^2_{\text{residual life}}$$

$$\sigma^2 = \sigma^2_{M/G/1} + \frac{E(T_0^3)}{3E(T_0)} - \left[\frac{E(T_0^2)}{2E(T_0)}\right]^2 \quad (A.34)$$

By looking at the expression of S*(s) as given by Eq. (A.23) we were able to decompose the time in system as the convolution of two independent random variables (we were not able to prove this result directly):

i) time in system if there were no rest period

ii) additional delay distributed as the residual life of the rest period

When the distribution function of the rest period is such that any rest period duration is smaller than or equal to any service time*, then this additional delay is the time t_1 elapsing between the arrival instant of the first customer (customer who starts the busy period) and the end of the rest period. Clearly (Poisson arrivals), the random variable t_1 is distributed as the residual life of the rest period. This is illustrated in Figure A.3 where the unfinished work U(t) (backlog in seconds of service) is plotted for both systems (with and without rest period) versus the time for a particular busy period. However, below we give two counterexamples, where the additional delay is not the time t_1 elapsing between the arrival of the first customer, denoted by C_1, and the end of the rest period, although this delay is distributed as the residual life of the rest period; this we know from Eq. (A.23).

*As an example, we might consider an M/D/1 slotted queueing system in which the rest period and the service time are identically distributed and deterministic with length equal to one slot.

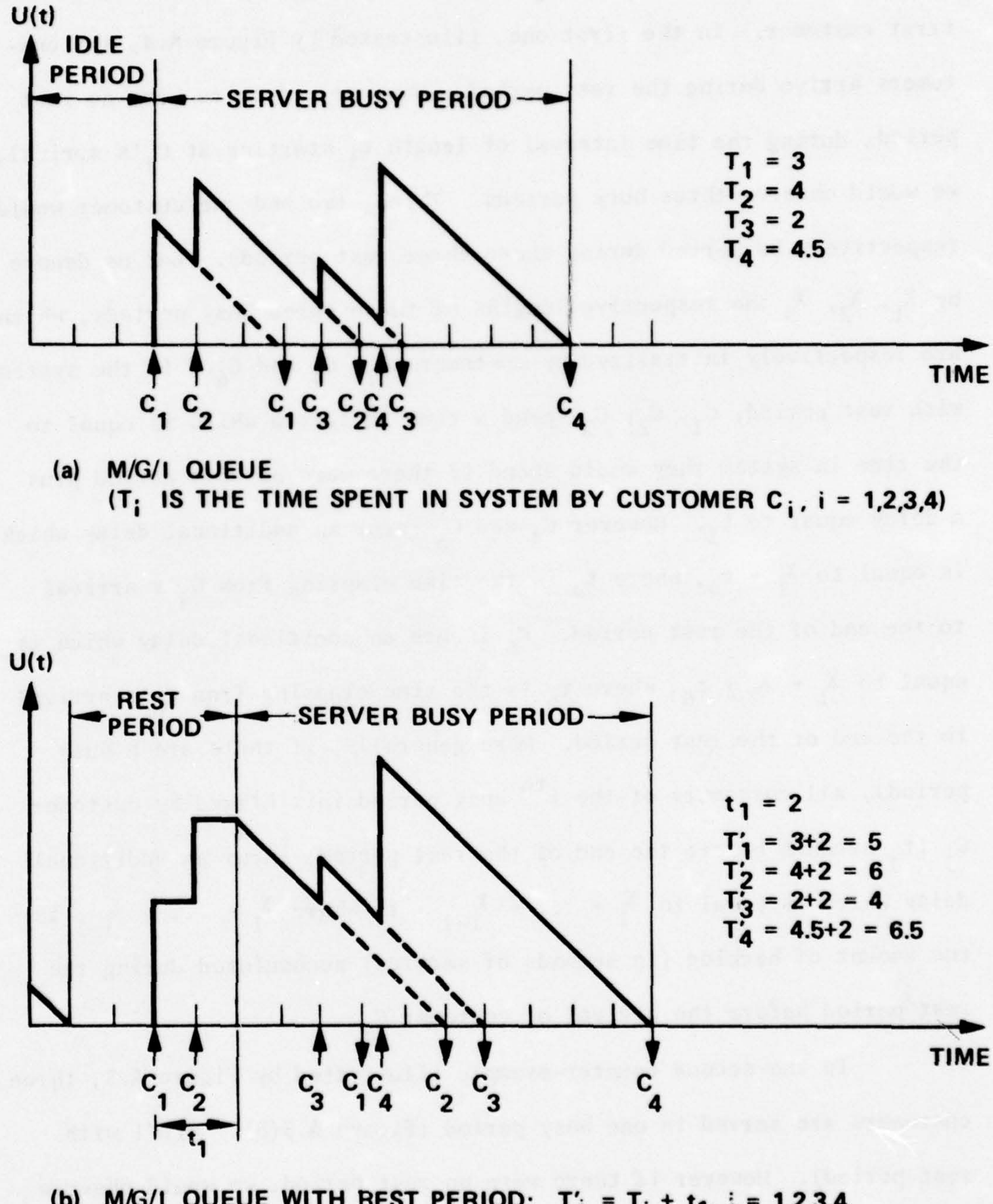
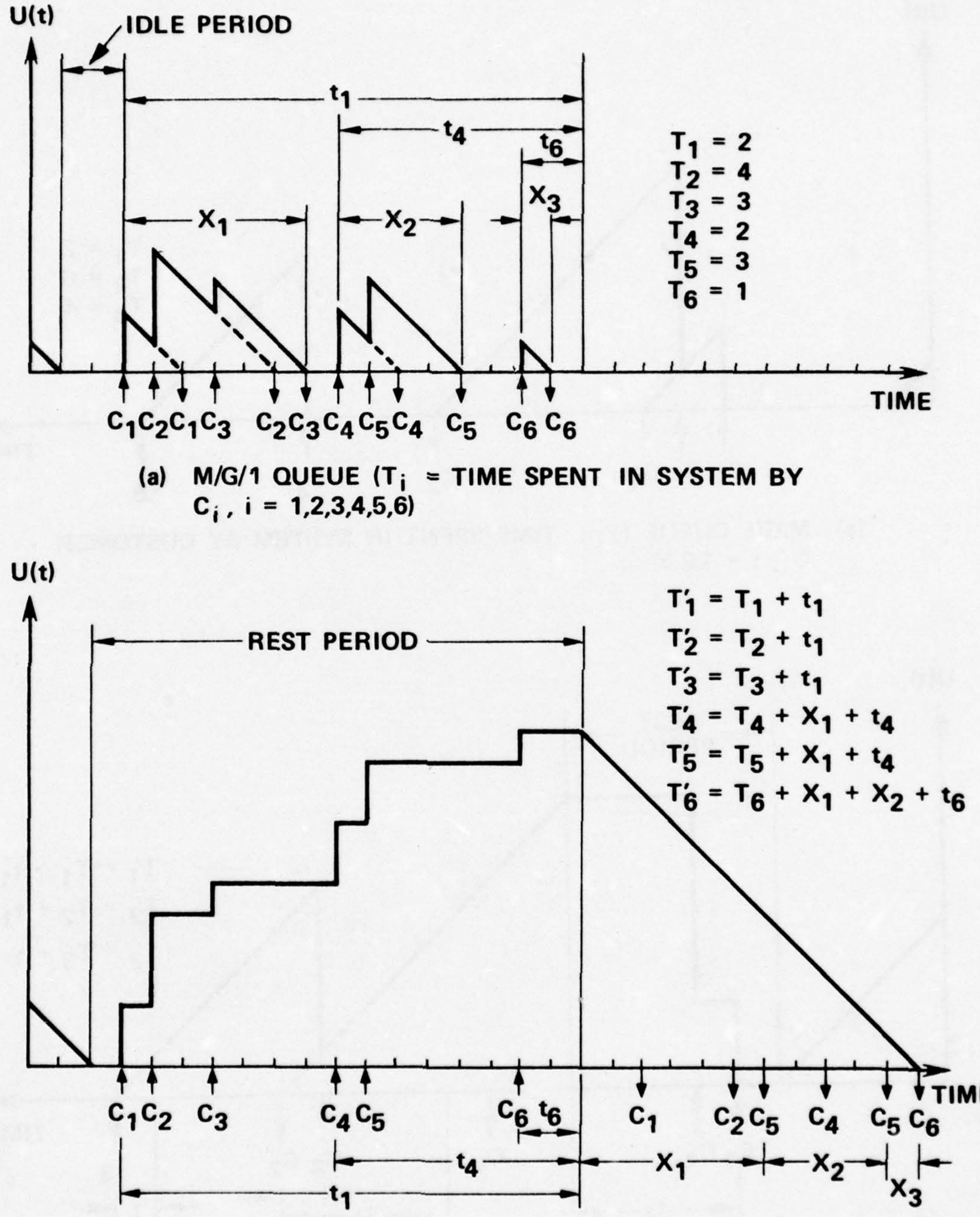


Figure A.3. Unfinished Work and Delay for a Particular Busy Period of 4 Customers

↑ Arrival ↓ Departure

In these two counterexamples t_1 is longer than the service time of the first customer. In the first one, illustrated by Figure A.4, six customers arrive during the rest period. However, if there were no rest period, during the time interval of length t_1 starting at C_1's arrival, we would observe three busy periods. Three, two and one customer would respectively be served during these three rest periods. Let us denote by X_1, X_2, X_3 the respective lengths of these three busy periods, which are respectively initialized by customers C_1, C_4 and C_6. In the system with rest period, C_1, C_2, C_3 spend a time in system which is equal to the time in system they would spend if there were no rest period plus a delay equal to t_1. However C_4 and C_5 incur an additional delay which is equal to $X_1 + t_4$, where t_4 is the time elapsing from C_4's arrival to the end of the rest period. C_6 incurs an additional delay which is equal to $X_1 + X_2 + t_6$, where t_6 is the time elapsing from C_6's arrival to the end of the rest period. More generally, if there are n busy periods, all customers of the i^{th} busy period initialized by customer C_j (t_j seconds before the end of the rest period) incur an additional delay which is equal to $X_1 + \dots + X_{i-1} + t_j$ where $X_1 + \dots + X_{i-1}$ is the amount of backlog (in seconds of service) accumulated during the rest period before the arrival of customer C_j.

In the second counter-example illustrated by Figure A.5, three customers are served in one busy period (Figure A.5(b): M/G/1 with rest period). However if there were no rest period, we would observe two busy periods, i.e., an idle period between the service of C_2 and the service of C_3. C_1 and C_2 incur an additional delay t_1 , where t_1 is defined as above while C_3 incurs an additional delay γ . It looks

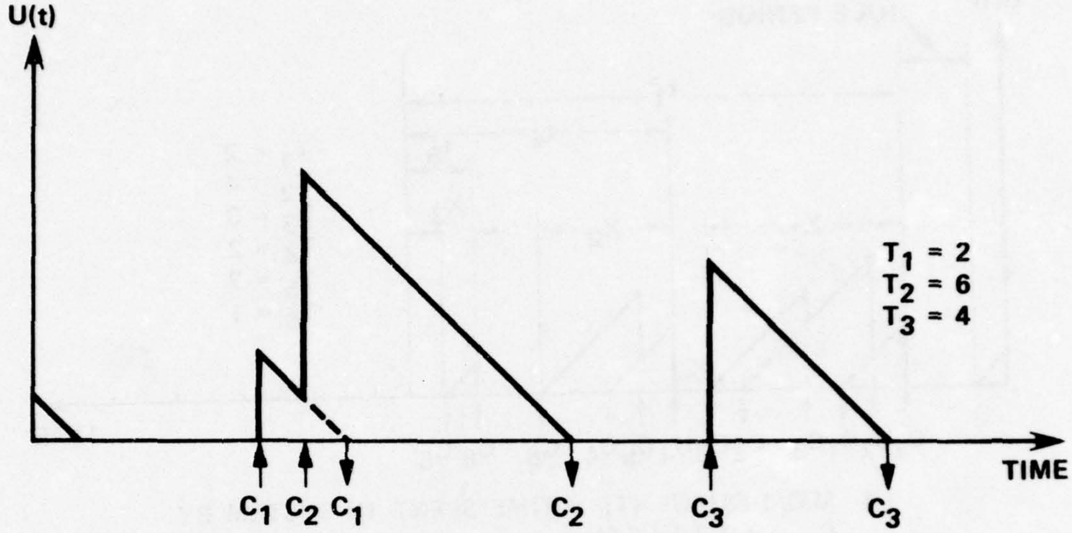


(a) M/G/1 QUEUE (T_i = TIME SPENT IN SYSTEM BY C_i, i = 1,2,3,4,5,6)

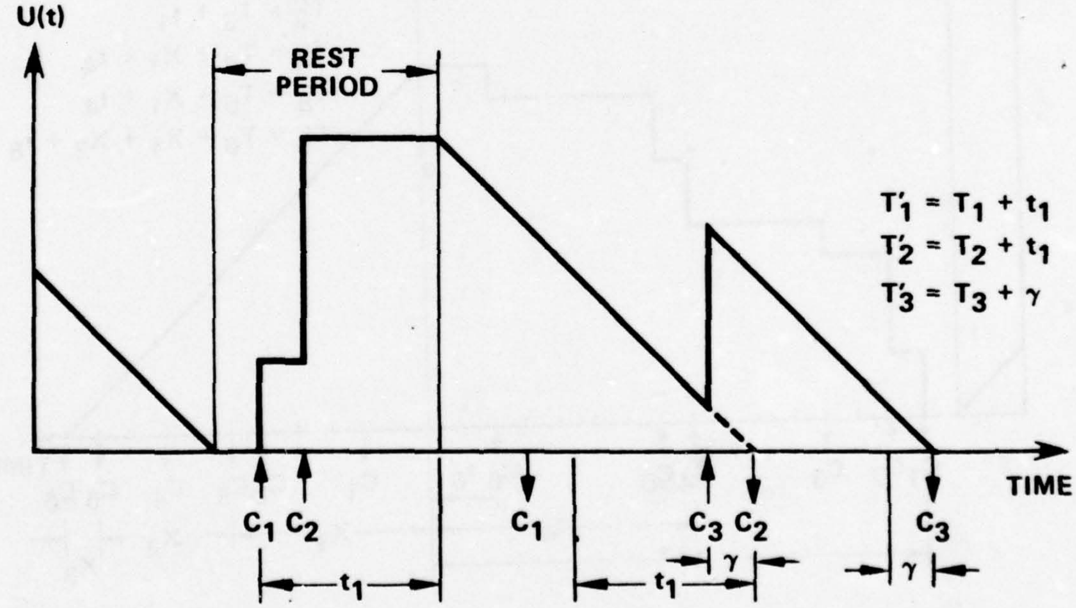
(b) M/G/1 QUEUE WITH REST PERIOD (T'_i = TIME SPENT IN SYSTEM BY C_i, i = 1,2,3,4,5,6)

Figure A.4. Unfinished Work and Time in System: First Counterexample for a Particular Busy Period of 6 Customers.

↑ ARRIVAL, ↓ DEPARTURE



(a) M/G/I QUEUE (T_i : TIME SPENT IN SYSTEM BY CUSTOMER C_i, i = 1,2,3)



(b) M/G/I QUEUE WITH REST PERIOD (T'_i = TIME SPENT IN SYSTEM BY CUSTOMER C_i, i = 1,2,3)

Figure A.5. Unfinished Work and Time in System: Second Counterexample for a Particular Busy Period of Three Customers; ↑ Arrival, ↓ Departure

like, when we introduce a rest period, the first busy period is shifted on the time axis by t_1 seconds. If during the last t_1 seconds, a new customer arrives and initiates a busy period (in the system without rest period: Figure A.5(a)), then this customer and all his "descendants" are going to incur the same additional delay γ (in the system with rest period: Figure A.5(b)).

A.4 Cycle, Busy Period, Probability the server is busy

Following Miller [MILL 64], we define a cycle as the length of time which elapses between the beginning of a rest period and the beginning of the following rest period (Figure A.6). We denote this random variable by T_c and the LST of its distribution function by $G_c^*(s)$. The server's busy period has a length denoted by T_b and lasts from the end of a rest period until the start of the next rest period (Figure A.6). the LST of its distribution function is denoted by $G_b^*(s)$. T_b, which has a finite probability of being equal to zero is the time during a cycle when the server is busy. Using an argument referred to as delay cycle analysis,* Miller derived $G_c^*(s)$ and $G_b^*(s)$ [MILL64]:

$$G_c^*(s) = R_0^*(s + \lambda - \lambda G^*(s)) \qquad (A.35)$$

$$G_b^*(s) = R_0^*(\lambda - \lambda G^*(s)) \qquad (A.36)$$

where $G^*(s)$ is the LST of the distribution function of the length of a busy period in an M/G/1 queueing system, with first and second moments denoted by g_1 and g_2.

*See [KLEI 76], Section 3.3, for a description of the delay cycle analysis.

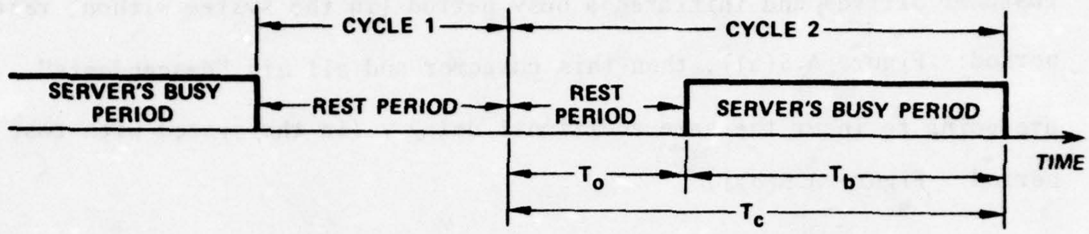


Figure A.6. Example of Cycles.

$G^*(s)$ is given [KLEI 75C] by

$$G^*(s) = B^*(s + \lambda - \lambda G^*(s)) \tag{A.37}$$

By taking the first derivative of Eq. (A.37), one easily gets g_1

$$g_1 = \frac{E(x)}{1 - \rho} \tag{A.38}$$

By taking the first derivative of Eq. (A.35) and Eq. (A.36), we have

$$E(T_c) = (1 + \lambda g_1)E(T_0)$$

$$E(T_b) = \lambda g_1 E(T_0)$$

Therefore, substituting Eq. (A.38) into the two previous equations we obtain

$$E(T_c) = \frac{E(T_0)}{1 - \rho} \tag{A.39}$$

$$E(T_b) = \frac{\rho\, E(T_0)}{1 - \rho} \tag{A.40}$$

where ρ has been defined (Eq. (A.2)) by

$$\rho = \lambda E(x)$$

By taking the second derivative of Eq. (A.37) we get

$$g_2 = \frac{E(x^2)}{(1 - \rho)^3} \qquad (A.41)$$

Taking the second derivative of Eq. (A.36) and substituting the value of g_2, as given in Eq. (A.41), we finally obtain

$$E(T_b^2) = \frac{\lambda E(x^2)}{(1 - \rho)^3} E(T_0) + \frac{\rho^2}{(1 - \rho)^2} E(T_0^2) \qquad (A.42)$$

Similarly, we obtain

$$E(T_c^2) = \frac{\lambda E(x^2)}{(1 - \rho)^3} E(T_0) + \frac{1}{(1 - \rho)^2} E(T_0^2) \qquad (A.43)$$

Busy Fraction

The proportion of time during which the server is idle (i.e., is taking rest periods) is given by $E(T_0)/E(T_c)$. This can be seen from the following argument:

At any instant during the realization of N cycles, the probability that we find the server in a rest period is equal to the total time represented by the rest periods divided by the total time of the N cycles. Dividing the numerator and denominator by N and letting N grow large, the numerator tends to $E(T_0)$, while the denominator tends to $E(T_c)$. It can be shown more formally from renewal theory [COX 62] [ROSS 70] that if $E(T_c) < \infty$ and the distribution of T_c is non-lattice, then

$$\lim_{t\to\infty} P(\text{server is idle at time } t) = \frac{E(T_0)}{E(T_c)} \qquad (A.44)$$

From Eq. (A.39), the condition $E(T_c) < \infty$ becomes

$$E(T_0) < \infty \quad \text{and} \quad \rho < 1 .$$

Substituting the expression of $E(T_c)$ as given by Eq. (A.39) into Eq. (A.44) we finally obtain

$$P\{\text{server is idle}\} = 1 - \rho \tag{A.45}$$

and of course

$$P\{\text{server is busy}\} = \rho = \lambda E(x) \tag{A.46}$$

For $\rho < 1$, we see then that the busy fraction in an M/G/1 queue with rest period is equal to the busy fraction in the M/G/1 queue (without rest period) having the same arrival and service processes. Thus, the busy fraction is independent of the distribution of T_0. If T_0 is long, then the busy period is long and the cycle is long. When the cycle is long, there are fewer rest periods per unit of time. Thus there is a self-regulating effect such that the M/G/1 queue with rest period has the same non-saturation condition ($\rho < 1$) as does the regular M/G/1 queue.

Denoting by p_0 the probability that the system is idle, i.e., there are no customers waiting for service or being served, we have in an M/G/1 queue

$$p_0 = P[\text{system idle}] = P[\text{server idle}] = 1 - \rho$$

while in an M/G/1 queue with rest period, we have

$$1 - \rho - p_0 = P[\text{server idle and there is at least one customer waiting for service}] > 0.$$

Until now, a cycle represented the interval elapsing from the beginning of a rest period to the beginning of the following rest period.

Below, such a cycle is referred to as an elementary cycle with length T_c.

Let us define a large cycle by the time interval elapsing from the end of a busy period to the end of the following busy period (Figure A.7). A large cycle, the length of which is denoted by T_L, is composed of $N(\geq 1)$ consecutive rest periods ("vacation" of length T_V) followed by one server's busy period. Let us denote by $E(T_L)$ the expected length of a large cycle. By the same argument as above [COX 62] we have

$$p_0 = \frac{E(T_I)}{E(T_L)} \tag{A.47}$$

where

$$E(T_I) = \frac{1}{\lambda} \tag{A.48}$$

is the expected length of an idle period of length T_I, i.e., a period where there is no customer either waiting for service or being served.

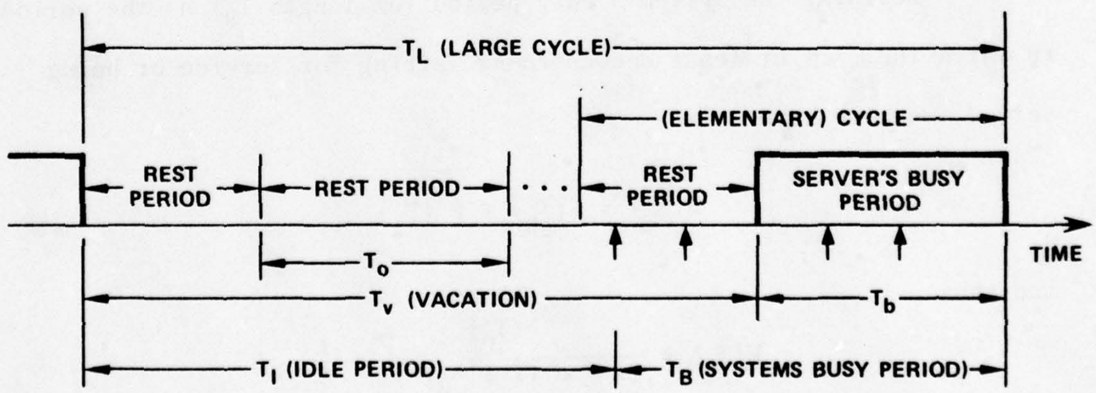


Figure A.7. Large Cycle, Vacation, Busy and Idle Periods.

We have

$$P\{N = k\} = \{1 - R_0^*(\lambda)\}[R_0^*(\lambda)]^{k-1} \qquad k = 1, 2, \ldots$$

and therefore, the expected length of the vacation is

$$E(T_V) = \frac{E(T_0)}{1 - R_0^*(\lambda)}$$

But this is also equal to the fraction of time in a large cycle $(1 - \rho)$ where the server is idle times the expected length $E(T_L)$ of a large cycle. We then have

$$E(T_L) = \frac{E(T_0)}{(1 - \rho)(1 - R_0^*(\lambda))} \tag{A.49}$$

Substituting Eq. (A.48) and Eq. (A. 49) in Eq. (A.47), we finally have

$$p_0 = \frac{(1 - \rho)(1 - R_0^*(\lambda))}{\lambda E(T_0)}$$

which is precisely Eq. (A.3).

Defining the system's busy period (of length T_B) as the period in which there is at least one customer waiting for service or being served, we also have

$$E(T_L) = E(T_B) + E(T_I)$$

and thus

$$E(T_B) = \frac{E(T_0)}{(1 - \rho)(1 - R_0^*(\lambda))} - \frac{1}{\lambda} .$$

APPENDIX B

SECOND MOMENT OF THE WAITING TIME IN AN M/G/1 QUEUE WITH REST PERIOD AND RANDOM ORDER OF SERVICE (ROS)

The queue M/G/1 with ROS has previously been studied and reported upon [KING 62, TAKA 63]. Following a similar approach to that which can be found in [KING 62], we derive below the second moment of the waiting time in an M/G/1 queue with rest period and ROS. We know that Eq. (A.1) gives the z-transform of the distribution function of the number in system (since the order of service is independent of service time; see Section 3.4). Moreover, the average waiting time must be the same as for FCFS (see Eq. (A.27) and Section 3.4).

The model we consider below is identical to the model studied in Appendix A in all respects except that the service discipline is now chosen as ROS; the server chooses a customer to be served at random from among all customers present in the queue.

We consider a tagged customer C entering the system. There are two possibilities, with respective probabilities ρ and $(1 - \rho)$. Either the server is busy, in which case C must wait a time u_1 before the server becomes free and we may write

$$\tilde{w}_1 = \tilde{u}_1 + \tilde{v}_1 \tag{B.1}$$

or the server is idle, in which case C must wait a time u_2 before the server ends his rest period and we may write

$$\tilde{w}_2 = \tilde{u}_2 + \tilde{v}_2 \tag{B.2}$$

The first and second moments of the distribution function of the waiting time w, are then expressed as

$$E(w) = \rho E(w_1) + (1 - \rho)E(w_2) \tag{B.3}$$

$$E(w^2) = \rho E(w_1^2) + (1 - \rho)E(w_2^2) \tag{B.4}$$

First we evaluate the first and second moments $E(w_1)$ and $E(w_1^2)$ of the waiting time of C, given that C finds the server busy.

We denote by S, the service time of the customer being served when C arrives [KING 62]. u_1 is uniformly distributed in (0, S) and from a well-known result in connection with forward recurrence times in renewal theory (see for example [COX 62]), the distribution of S is given by

$$P\{S \leq t\} = \int_0^t SdB(S) \Big/ \int_0^\infty SdB(S)$$

In particular, we have

$$E(S) = E(x^2)/E(x) \tag{B.5}$$

$$E(S^2) = E(x^3)/E(x) \tag{B.6}$$

Let us denote by n the number of customers in the system (excluding C) just after the end of the service time S and by m the number of customers left behind by the previous departure (the last customer served before the period S) [KING 62]. From Appendix A (Eq. (A.1)) we know that

$$E(z^m) = \frac{B^*(\lambda - \lambda z)}{\lambda E(T_0)} \frac{(1-\rho)(1 - R_0^*(\lambda - \lambda z))}{B^*(\lambda - \lambda z) - z} \tag{B.7}$$

Then it is easy to show that

$$n = m - \Delta m + k \tag{B.8}$$

where $\Delta m = 1$ if $m > 0$ and 0 otherwise, and where k is the number of arrivals (excluding C) during the service time S. The generating function of k is given by

$$E(z^k/S) = e^{-\lambda S(1-z)} \tag{B.9}$$

Thus

$$E[z^n/S] = E(z^{m-\Delta m+k}/S)$$

and since k is independent of m, we have

$$E(z^n/S) = E(z^{m-\Delta m})E(z^k/S)$$

$$= E(z^{m-\Delta m})e^{-\lambda S(1-z)}$$

Using the same argument as in Appendix A to evaluate $E(z^{m-\Delta m})$, we finally have

$$E(z^n/S) = \frac{(1-\rho)(1 - R_0^*(\lambda - \lambda z))}{\lambda E(T_0)[B^*(\lambda - \lambda z) - z]} e^{-\lambda S(1-z)} \tag{B.10}$$

In particular

$$E(n/S) = \lambda S + \frac{\lambda^2 E(x^2)}{2(1-\rho)} + \frac{\lambda E(T_0^2)}{2E(T_0)} \tag{B.11}$$

and

$$E(n(n-1)/S) = \frac{\lambda^2 E(x^2)}{1-\rho}\left[\frac{\lambda^2 E(x^2)}{2(1-\rho)} + \lambda S\right] + (\lambda S)^2 + \frac{\lambda^3 E(x^3)}{3(1-\rho)}$$

$$+ \frac{\lambda E(T_0^2)}{E(T_0)}\left[\frac{\lambda^2 E(x^2)}{2(1-\rho)} + \lambda S\right] + \frac{\lambda^2 E(T_0^3)}{3E(T_0)} \qquad (B.12)$$

The problem of determining the waiting time w_1 is thus reduced to that of finding the distribution of v_1, given n [KING 62]. In particular, we may write

$$E(w_1) = E(u_1) + E(v_1) \qquad (B.13)$$

$$E(w_1) = E_S(E(w_1/S)) \qquad (B.14)$$

$$E(w_1/S) = S/2 + E(v_1/S) \qquad (B.15)$$

$$E(v_1/S) = E_n(E(v_1/S,n)) \qquad (B.16)$$

Similarly, we have

$$E(w_1^2) = E(u_1^2) + 2E(u_1)E(v_1) + E(v_1^2) \qquad (B.17)$$

$$E(w_1^2) = E_S(E(w_1^2/S)) \qquad (B.18)$$

$$E(w_1^2/S) = E(u_1^2/S) + 2E(u_1/S)E(v_1/S) + E(v_1^2/S) \qquad (B.19)$$

$$E(w_1^2/S) = S^2/3 + SE(v_1/S) + E(v_1^2/S) \qquad (B.20)$$

$$E(v_1^2/S) = E_n(E(v_1^2/S,n)) \qquad (B.21)$$

The problem is thus reduced to that of finding $E(v_1/S,n)$ and $E(v_1^2/S,n)$). Kingman solved for the distribution function of v_1, conditioned on S and n [KING 62]. Takacs evaluated the first and second moments

[TAKA 63]:

$$E(v_1/S,n) = \frac{E(x)n}{1 - \rho} \tag{B.22}$$

$$E(v_1^2/S,n) = An(n - 1) + Bn \tag{B.23}$$

where

$$A = \frac{2[E(x)]^2}{(2 - \rho)(3 - 2\rho)} \tag{B.24}$$

and

$$B = \frac{(6 - \rho)E(x^2)}{(2 - \rho)^2(3 - 2\rho)} \tag{B.25}$$

From Eqs. (B.22) and (B.16) we have

$$E(v_1/S) = \frac{E(x)E(n/S)}{2 - \rho} \tag{B.26}$$

From Eqs. (B.21) and (B.23) we have

$$E(v_1^2/S) = AE(n(n - 1)/S) + BE(n/S) \tag{B.27}$$

Substituting Eq. (B.11) in Eq. (B.26), we obtain an expresssion for $E(v_1/S)$. Substituting this expression in Eq. (B.15) we have

$$E(w_1/S) = S/2 + \frac{\rho}{2 - \rho}\left[S + \frac{\lambda E(x^2)}{2(1 - \rho)} + \frac{E(T_0^2)}{2E(T_0)}\right] \tag{B.28}$$

Substituting Eqs. (B.12) and (B.11) in Eq. (B.27), we get an expression for $E(v_1^2/S)$ that we substitute together with the expression for $E(v_1/S)$ in Eq. (B.20). We then have

$$E(w_1^2/S) = \frac{S^2}{3} + \left[\frac{S\rho}{2 - \rho} + \lambda B\right]\left[S + \frac{\lambda E(x^2)}{2(1 - \rho)} \div \frac{E(T_0^2)}{2E(T_0)}\right]$$

$$+ A\lambda^2\left[S^2 + \left(\frac{\lambda E(x^2)}{2(1 - \rho)} + S\right)\left(\frac{\lambda E(x^2)}{1 - \rho} + \frac{E(T_0^2)}{E(T_0)}\right) + \frac{\lambda E(x^3)}{3(1 - \rho)} + \frac{E(T_0^3)}{3E(T_0)}\right]$$

(B.29)

From Eqs. (B.5), (B.14) and (B.28), we finally have

$$E(w_1) = \frac{1}{2 - \rho}\left[(2 + \rho)\frac{E(x^2)}{2E(x)} + \rho\left\{\frac{\lambda E(x^2)}{2(1 - \rho)} + \frac{E(T_0^2)}{2E(T_0)}\right\}\right] \tag{B.30}$$

From Eqs. (B.5), (B.6), (B.18), (B. 24), (B. 25) and (B. 29), after some algebraic manipulations, we have the result we were seeking:

$$\begin{aligned} E\left(w_1^2\right) = {} & \frac{2\lambda E(x^3)}{3\rho(1 - \rho)(2 - \rho)} + \frac{[\lambda E(x^2)]^2}{\rho(1 - \rho)^2(2 - \rho)} \\ & + \frac{2\rho^2}{(2 - \rho)(3 - 2\rho)}\,\frac{E(T_0^3)}{3E(T_0)} + \lambda E(x^2) \\ & \times \frac{E(T_0^2)}{2E(T_0)}\,\frac{2}{(2 - \rho)^2(3 - 2\rho)}\left[2 + \frac{(2 - \rho)^2}{1 - \rho}\right] \end{aligned} \tag{B.31}$$

It remains to evaluate the first and second moments $E(w_2)$ and $E(w_2^2)$ of the waiting time of C, given that C arrives during a rest period. We denote also by S the length of this rest period. u_2 is uniformly distributed in (0, S). From the same renewal argument as above [COX 62], we have

$$P\{S \leq t\} = \int_0^t S dR_0(S) \Big/ \int_0^\infty S dR_0(S) \tag{B.32}$$

where $R_0(S)$ denotes the distribution function of the rest period. In particular we have

$$E(S) = \frac{E(T_0^2)}{E(T_0)} \tag{B.33}$$

and

$$E(S^2) = \frac{E(T_0^3)}{E(T_0)} \tag{B.34}$$

Denoting also by n the number of customers in the system (excluding C), at the end of the rest period, we have:

$$E[z^n/S] = e^{-\lambda S(1-z)} \tag{B.35}$$

and

$$E(n/S) = \lambda S \tag{B.36}$$

$$E(n(n-1)/S) = (\lambda S)^2 \tag{B.37}$$

The problem of determining the waiting w_2 is reduced to that of finding the distribution of v_2 given n, which is precisely the distribution of v_1, given n. We may then apply the results obtained above.

In particular, substituting Eq. (B.36) in Eq. (B.26), we have

$$E(v_2/S) \approx \frac{\rho S}{2-\rho} \tag{B.38}$$

but

$$E(w_2/S) = \frac{S}{2} + E(v_2/S) \tag{B.39}$$

then

$$E(w_2/S) = \frac{2+\rho}{2(2-\rho)} S$$

and

$$E(w_2) = \frac{2+\rho}{2-\rho} \frac{E(T_0^2)}{2E(T_0)} \tag{B.40}$$

Substituting Eq. (B.30) and (B.40) in Eq. (B.3), we verify, as we might expect, that the expected waiting time is the same as in an M/G/1 queue with rest period and FCFS order of service (see Eq. (A.27)):

$$E(w) = \frac{\lambda E(x^2)}{2(1-\rho)} + \frac{E(T_0^2)}{2E(T_0)} \tag{B.41}$$

Substituting Eq. (B.36) and (B.37) in Eq. (B.27), we have

$$E(v_2^2/S) = A(\lambda S)^2 + B\lambda S \tag{B.42}$$

but

$$E(w_2^2/S) = \frac{S^2}{3} + SE(v_2/S) + E(v_2^2/S) \tag{B.43}$$

Substituting Eqs. (B.24), (B.25), (B.38) and (B.42) into Eq. (B.43), we have

$$E(w_2^2/S) = S^2\left[\frac{1}{3} + \frac{\rho}{2-\rho} + \frac{2\rho^2}{(2-\rho)(3-2\rho)}\right] + \frac{\lambda E(x^2)(6-\rho)}{(2-\rho)^2(3-2\rho)}S$$

Unconditioning on S, we obtain the result we were seeking:

$$E(w_2^2) = \frac{E(T_0^3)}{E(T_0)}\left[\frac{1}{3} + \frac{\rho}{2-\rho} + \frac{2\rho^2}{(2-\rho)(3-2\rho)}\right] + \lambda E(x^2)\,\frac{E(T_0^2)}{E(T_0)}\,\frac{6-\rho}{(2-\rho)^2(3-2\rho)} \tag{B.44}$$

Knowing the second moment of the waiting time in both cases (server idle or busy), we can finally obtain the second moment of waiting time. Substituting Eq. (B.31) and Eq. (B.44) into Eq. (B.4), we have, after some algebraic manipulations:

$$E(w^2) = \frac{2}{2-\rho}\left[\frac{\lambda E(x^3)}{3(1-\rho)} + \frac{\lambda E(x^2)}{1-\rho}\left(\frac{\lambda E(x^2)}{2(1-\rho)} + \frac{E(T_0^2)}{2E(T_0)}\right) + \frac{E(T_0^3)}{3E(T_0)}\right] \tag{B.45}$$

Comparing Eq.(B.45) to Eq. (A.33), which gives the second moment of the waiting time in an M/G/1 queue with rest period and FCFS order of service, we have the following relationship.

$$E(w^2) \triangleq \overline{W^2_{ROS}} = \frac{\overline{W^2_{FCFS}}}{1-\rho/2} \tag{B.46}$$

This is precisely the relationship found in [TAKA 63] between the second moments of the waiting time in a regular (no rest period) ROS M/G/1 queue and in a regular FCFS M/G/1 queue.

In Appendix C, we show that the second moments of the waiting time in a LCFS M/G/1 queue with rest period, and in a FCFS M/G/1 queue with rest period are related by the following equation:

$$\overline{W^2_{LCFS}} = \frac{\overline{W^2_{FCFS}}}{1 - \rho} \tag{B.47}$$

From Eq. (B.46) and Eq. (B.47) we obviously have

$$\overline{W^2_{FCFS}} < \overline{W^2_{ROS}} < \overline{W^2_{LCFS}}$$

APPENDIX C

THE QUEUE M/G/1 WITH REST PERIOD AND LAST-COME-FIRST-SERVED (LCFS) ORDER OF SERVICE

The queue M/G/1 with LCFS order of service has been extensively studied [VAUL 54, RIOR 61, TAKA 63]. Kleinrock gives [KLEI 76, Sec. 3.5] the Laplace-Stieljes Transform (LST) of the distribution function of the waiting time, using a delay cycle analysis. Following almost exactly the same argument, we solve for the LST of the distribution function of the waiting time $W^*_{LCFS}(s)$ in an M/G/1 queue with rest period and LCFS order of service. We know that Eq. (A.1) gives the distribution function of the number in system (since the order of service is independent of service time; see Section 3.4). Moreover, the average waiting time must be the same as for FCFS (see Eq. (A.27) and Section 3.4).

Laplace-Stieljes Transform of the Distribution Function of the Waiting Time $W^*_{LCFS}(s)$

The model we consider below is identical to the model studied in Appendix A in all respects except service discipline, which is now chosen as LCFS. All the system's parameters are denoted as in Appendix A.

A new arrival finds the server either busy (with probability ρ, see Appendix A), or in a rest period (with probability $1 - \rho$). In any case the new arrival is not affected by the customers, if any, present in the waiting line.

If the server is busy, only the customer found in service makes the new arrival wait, and his delay is due to arrivals that enter the system after he does and before his service's initiation.

If the server is idle, the new arrival has to wait until the end of the rest period, at which point in time his service is initiated only if no new arrivals occurred meanwhile. Otherwise (new arrivals occurred before the end of the rest period) he incurs a supplementary delay due to arrivals that enter the system after he does and before his entry into service.

Then we may apply the delay cycle analysis [KLEI 76] where the initial delay is either

the residual life of the customer found in service, or

the residual life of the rest period;

and the delay busy period is the time interval necessary to serve all those arrivals that enter after he does and before his initiation of service.

If we now express $W^*_{LCFS}(s)$ in terms of the LST's of the distribution functions of the waiting time conditioned on the server state (busy or idle), we have

$$W^*_{LCFS}(s) = E[e^{-sw}/\text{server busy upon arrival}]\rho + E[e^{-sw}/\text{server idle upon arrival}](1 - \rho) \qquad (C.1)$$

First we consider the case where the server is busy. The LST of the distribution function of the residual life of the customer found in service by the new arrival $G^*_0(s)$ is given by

$$G^*_0(s) = \frac{1 - B^*(s)}{sE(x)} \qquad (C.2)$$

Thus we have

$$E[e^{-s\tilde{w}}/\text{server busy upon arrival}] = G_0^*(s + \lambda - \lambda G^*(s)) \quad (C.3)$$

where $G^*(s)$ is the LST of the distribution function of the busy period in an M/G/1 queue. Substituting Eq. (C.2) into Eq. (C.3). we have

$$E[e^{-s\tilde{w}}/\text{server busy upon arrival}] = \frac{1 - B^*(s + \lambda - \lambda G^*(s))}{[s + \lambda - \lambda G^*(s)]E(x)} \quad (C.4)$$

Similarly, in the case where the server is idle we may write down the conditional transform for waiting time as

$$E[e^{-s\tilde{w}}/\text{server idle upon arrival}] = C^*(s + \lambda - \lambda G^*(s)) \quad (C.5)$$

where $C^*(s)$ is the LST of the distribution function of the residual life of the rest period and is given by

$$C^*(s) = \frac{1 - R_0^*(s)}{E(T_0)} \quad (C.6)$$

Substituting Eq. (C.6) in Eq. (C.5) we have

$$E[e^{-s\tilde{w}}/\text{server is idle upon arrival}] = \frac{1 - R_0^*(s + \lambda - \lambda G^*(s))}{[s + \lambda - \lambda G^*(s)]E(T_0)} \quad (C.7)$$

Substituting Eq. (C.4) and Eq. (C.7) in Eq. (C.1), we have the expression of $W^*(s)$ we were seeking:

$$W^*(s) = \frac{\rho[1 - B^*(s + \lambda - \lambda G^*(s))]}{[s + \lambda - \lambda G^*(s)]E(x)} + \frac{(1 - \rho)[1 - R_0^*(s + \lambda - \lambda G^*(s)]}{[s + \lambda - \lambda G^*(s)]E(T_0)} \quad (C.8)$$

where $G^*(s)$ is given [KLEI 76] by

$$G^*(s) = B^*(s + \lambda - \lambda G^*(s))$$

By taking the first and second derivatives of Eq. (C.8) at s = 0, we find the first and second moments of the waiting time:

$$E(W_{LCFS}) = \frac{\lambda E(x^2)}{2(1 - \rho)} + \frac{E(T_0^2)}{2E(T_0)} \tag{C.9}$$

$$E(W_{LCFS}^2) = \frac{\lambda E(X^2)}{(1 - \rho)^2} E(W_{LCFS}) + \frac{\lambda E(x^3)}{3(1 - \rho)^2} + \frac{E(T_0^3)}{3E(T_0)} \times \frac{1}{1 - \rho} \tag{C.10}$$

As we stated earlier, $E(W_{LCFS})$ is of course equal to $E(W_{FCFS})$, the expected waiting time in an M/G/1 queue with rest period and FCFS order of service (see Eq. (A.27)).

But the second moment (W_{LCFS}^2) is larger than $E(W_{FCFS}^2)$ the second moment of the waiting time in an M/G/1 queue with rest period and FCFS order of service (see Eq. (A.33)). From Eq. (A.33) we may simplify Eq. (C.10) to give

$$E(W_{LCFS}^2) = \frac{E(W_{FCFS}^2)}{1 - \rho} \tag{C.11}$$

This is precisely the relationship found in [TAKA 63] between the second moments of the waiting time in a regular (no rest period) LCFS M/G/1 queue and in a regular FCFS M/G/1 queue.

In conclusion, the second moments of the waiting time $\overline{W^2_{FCFS}}$, $\overline{W^2_{ROS}}$ and $\overline{W^2_{LCFS}}$ are related by the following equation (see Appendix B and Chapter 3, Eq. (3.61)).

$$\overline{W^2_{FCFS}} = \left(1 - \frac{\rho}{2}\right)\overline{W^2_{ROS}} = (1 - \rho)\overline{W^2_{LCFS}}$$

This result holds for the M/G/1 queue with rest period as well as for the regular M/G/1 queue (without rest period).

APPENDIX D

THE QUEUE M/G/1 WITH INITIAL SET-UP TIME AND FCFS ORDER OF SERVICE

We solve for the z-transform of the distribution function of the number in system Q(z) in the queue M/G/1 with initial set-up time and FCFS order of service. The method we use is exactly the one we used for the FCFS queue M/G/1 with rest period (Appendix A) and is based on the "imbedded Markov chain" approach [KEND 51, KLEI 75C]. We then extend the conservation law first obtained by Kleinrock [KLEI 65] to such a queueing system by exactly the same argument used for the M/G/1 queue with rest period (see Section 3.4).

The model we consider below is identical to the regular M/G/1 queue with FCFS order of service in all respects except that there is a "set-up time" $\tilde{y}$ associated with the beginning of a busy period. The idle period ends with the arrival of the first customer; y seconds after the first customer's arrival, the server starts to serve the first customer. When the server completes the first customer's service, he starts immediately to serve the next one in queue, if any, according to a FCFS discipline, etc. The set-up time y is drawn from an arbitrary distribution function with Laplace-Stieljes transform Y*(s) and first and second moments denoted by E[y] and E[y^2]. The set-up times are independent of each other and of the arrival and service processes.

D.1 z-Transform of the Distribution Function of the Number of Customers in System at all Points in Time, and Expected Time in System

Following the notations of Appendix A, we choose the number of customers q_n left behind the departure of the n^{th} customer, C_n, entering the system as our state variable. We proceed to solve for the system behavior just after the departure instants (which form an imbedded Markov chain). Following the same approach as the one used in [MILL 64] for the M/G/1 queue with rest period, one can show that the solution at these points will also provide the solution for all points in time. We may then relate q_{n+1} to q_n by the following equation:

$$q_{n+1} = q_n - \Delta q_n + V_{n+1} + (1 - \Delta q_n)u_{n+1} \tag{D.1}$$

where

$$\Delta q_n = \begin{cases} 1 & \text{if} \quad q_n > 0 \\ 0 & \text{if} \quad q_n = 0 \end{cases}$$

and u_{n+1} represents the number of customers arriving during the set-up time, given that C_{n+1} starts the busy period. We proceed now to calculate $Q(z)$. Assuming ergodicity, the following limits exist:

$$\tilde{q} = \lim_{n \to \infty} q_n$$

$$Q(z) = E[z^{\tilde{q}}] = \lim_{n \to \infty} E[z^{\tilde{q}_n}]$$

We have from Eq. (D.1)

$$E[z^{q_{n+1}}] = E[z^{q_n - \Delta q_n + V_{n+1} + (1-\Delta q_n)u_{n+1}}]$$

Following the same approach as in Appendix A, one easily obtains the following expression for Q(z):

$$Q(z) = p_0 \frac{B^*(\lambda - \lambda z)[1 - zY^*(\lambda - \lambda z)]}{B^*(\lambda - \lambda z) - z} \tag{D.2}$$

where $p_0 \triangleq P\{\Delta\tilde{q} = 0\} = \lim_{n \to \infty} P\{\Delta q_n = 0\}$ represents the probability of an empty system and is given by

$$p_0 = \frac{1 - \rho}{1 + \lambda E(y)} \tag{D.3}$$

where $\rho \triangleq 1 - \lambda E(x)$.

One can also derive Eq. (D.3) by the following argument: It is easy to show that*

$$P[\text{server is idle}] = 1 - \rho \tag{D.4}$$

then we have

$$p_0 = P[\text{system idle/server idle}] \times P[\text{server idle}] \tag{D.5}$$

From renewal theory [COX 61], one can easily show that

$$P[\text{system idle/server idle}] = \frac{1/\lambda}{1/\lambda + E(y)} \tag{D.6}$$

since the interval of time during which the server is idle is the convolution of the (system's) idle period (average length $1/\lambda$) and the set-up time (average length E(y)).

Substituting Eq. (D.4) and Eq. (D.6) into Eq. (D.5), we get Eq. (D.3).

---

*In a regular M/G/1 queue [KLEI 75C], we have

$$P[\text{system empty}] = P[\text{server idle}] = 1 - \lambda\bar{x}.$$

And of course, we have

$$P[\text{set-up time}] = (1 - \rho)\, \frac{E(y)}{1/\lambda + E(y)} \qquad (D.7)$$

By taking the first derivative of the RHS of Eq. (D.2), at $z = 1$, we have

$$E(q) = \rho + \frac{\lambda^2 E(x^2)}{2(1 - \rho)} + \frac{\lambda^2 E(y^2) + 2\lambda E(y)}{2(1 + \lambda E(y))}$$

and from Little's result we obtain the expected time in system

$$E(T) = E(x) + \frac{\lambda E(x^2)}{2(1 - \rho)} + \frac{\lambda E(y^2) + 2E(y)}{2(1 + \lambda E(y))} \qquad (D.8)$$

The sum of the two first terms of the RHS of Eq. (D.8) represent the Pollaczek-Khinchin mean value formula [KLEI 75C]. One may easily show, as in Appendix A, that the LST of time in system is given by

$$S^*(s) = \frac{(1 - \rho)B^*(s)}{1 + \lambda E(y)} \; \frac{[\lambda - (\lambda - s)Y^*(s)]}{s - \lambda + \lambda B^*(s)}$$

The results of this section are consistent with the results derived by another approach in [MILL 64] for a slightly different model (FCFS M/G/1 queue where the first customer served in a busy period has a service time drawn from a distribution function $B_1(\cdot)$ different from all other customers' service time distribution function $B(\cdot)$).

D.2 Conservation Law

We can easily extend the conservation law first stated and proved by Kleinrock [KLEI 65] to an M/G/1 queue with initial set-up time and any conservative, non-preemptive queueing discipline which is independent of any measure of the service time (see Section 3.4).

First, one may easily show that Eq. (3.14) holds for such a queueing system. Using the same notations as in Chapter 3, if arriving customers belong to one of a set of P different classes, the expected waiting times $\overline{W}_p$* are related by the following linear equation:

$$\overline{U} = \overline{W}_0 + \sum_{p=1}^{P} \rho_p \overline{W}_p \qquad (D.9)$$

where $\overline{U}$ denotes the expected backlog, $\overline{W}_0$ and ρ_p are given respectively by Eq. (3.7) and Eq. (3.5). Eq. (D.9) holds for any conservative, non-preemptive and service-independent queueing discipline.

Now since $\overline{U}$ is independent of the order of service, we may as well calculate $\overline{U}$ for a strict FCFS discipline. For an arrival occuring at time t, one recognizes that:

1) if the arrival occurs during a period when the server is busy (with probability ρ), then obviously

$$W_{FCFS}(t) = U(t)$$

2) if the arrival occurs during a (system's) idle period (with probability $(1 - \rho) \times \frac{1}{1 + \lambda E(y)}$, see Eq. (D.3)), then

$$W_{FCFS}(t) = U(t) + y$$

Indeed at t, the backlog is $U(t) = 0$, and the customer has to wait $\tilde{y}$ seconds (set-up time).

3) if the arrival occurs during the set-up time (with probability $(1 - \rho) \times \frac{\lambda E(y)}{1 + \lambda E(y)}$, see Eq. (D.7)), then

---

*Customers from group p, p = 1, ..., P, incur an expected waiting time $\overline{W}_p$.

$$W_{FCFS}(t) = U(t) + z$$

where z is the residual life of the set-up time. Indeed, the customer arriving at t has to wait z seconds before service starts plus U(t) seconds. (U(t) is the time it takes to serve all customers who arrived before our customer.)

Taking expectations and observing that $E(z) = \frac{E(y^2)}{2E(y)}$, we have

$$\overline{W}_{FCFS} = \overline{U} + E(y)\,\frac{(1 - \rho)}{1 + \lambda E(y)} + \frac{E(Y^2)}{2E(y)}\;\frac{(1 - \rho)\lambda E(y)}{1 + \lambda E(y)}$$

$$\overline{W}_{FCFS} = \overline{U} + (1 - \rho)\,\frac{2E(y) + \lambda E(y^2)}{2(1 + \lambda E(y))} \qquad \text{(D.10)}$$

If we use the value of $\overline{U}$ drawn from Eq. (D.10) in Eq. (D.9), we have

$$\overline{W}_{FCFS} - (1 - \rho)\,\frac{2E(y) + \lambda E(y^2)}{2(1 + \lambda E(y))} = \overline{W}_0 + \sum_{p=1}^{P} \rho_p \overline{W}_p \qquad \text{(D.11)}$$

Eq. (D.11) is true regardless of the order of service. In particular, if the discipline is FCFS, $\overline{W}_p = \overline{W}_{FCFS}$ for all p, and since $\sum_{p=1}^{P} \rho_p = \rho$ we finally have

$$\overline{W}_{FCFS} = \frac{\overline{W}_0}{1 - \rho} + \frac{2E(y) + \lambda E(y^2)}{2(1 + \lambda E(y))} \qquad \text{(D.12)}$$

Eq. (D.12) is consistent with Eq. (D.8) since $\overline{W}_0 = \lambda E(x^2)/2$ when the discipline is FCFS.

Substituting the value of $\overline{W}_{FCFS}$ as given by Eq. (D.12) into Eq. (D.11), we have the following conservation law:

$$\sum_{p=1}^{P} \rho_p \overline{W}_p = \rho\,\frac{\overline{W}_0}{1 - \rho} + \rho\,\frac{2E(y) + \lambda E(y^2)}{2(1 + \lambda E(y))} \qquad \rho < 1 \qquad \text{(D.13)}$$

BIBLIOGRAPHY

ABRA 70 Abramson, N. "The Aloha System - Another Alternative for Computer Communications," Fall Joint Computer Conference, AFIPS Conference Proceedings, Vol. 37, pp. 281-285, 1970.

ABRA 73 Abramson, N. "Packet Switching with Satellites," National Computer Conference, New York, June 4-8, 1973, AFIPS Conference Proceedings, Vol. 42, pp. 695-702, 1973.

BARA 64 Baran, P. "On Distributed Communications: XI. Summmary Overview," Rand Corporation, Santa Monica, California, Memorandum RM-3767-PR, August 1964.

BIND 72 Binder, R. "Another Aloha Satellite Protocol," ARPA Network Information Center, Stanford Research Institute, Menlo Park, California, ASS Note 32 (NIC 13147), December 1972.

BIND 75A Binder, R. "A Simple Mixed-Traffic Technique for Ground-based Aloha Channels," University of Hawaii, Honolulu, Hawaii, Technical report B75-6, January 1975.

BIND 75B Binder, R. "A Dynamic Packet-Switching System for Satellite Broadcast Channels," Proceedings of the IEEE International Conference on Communications, Vol. 3, June 1975.

BIND 75C Binder, R,. N. Abramson, F. Kuo, A. Okinaka, D. Wax. "Aloha Packet Broadcasting - A Retrospect," National Computer Conference, AFIPS Conference Proceedings, Vol. 44, pp. 203-215, 1975.

BURC 75 Burchfiel, J., R. Tomlinson and M. Beeler. "Functions and Structure of a Packet Radio Station," National Computer Conference, AFIPS Conference Proceedings, Vol. 44, pp. 245-251, 1975.

CARR 70 Carr, S., S. Crocker and V. Cerf. "Host-Host Communication Protocol in the ARPA Network," Spring Joint Computer Conference, AFIPS Conference Proceedings, Vol. 36, pp. 589-597, 1970.

CHU 69 Chu. W.W. "A Study of Asynchronous Time Division Multiplexing for Time-Sharing Computer Systems," Spring Joint Computer Conference, AFIPS Conference Proceedings, Vol. 35, pp. 669-678, 1969.

COBH 54 Cobham, A. "Priority Assignment in Waiting Line Problems," Operations Research, Vol. 2, pp. 70-76, 1954.

COX 62 Cox, D.R. "Renewal Theory," Monographs on Applied Probability and Statistics, Chapman and Hall, London, 1962.

CROW 73 Crowther, W., R. Rettberg, D. Walden, S. Ornstein, and F. Heart. "A System for Broadcast Communication: Reservation Aloha," Proceedings of the Sixth Hawaii International Conference on System Sciences, University of Hawaii, Honolulu, January 1973.

EURO 73 Eurodata. "A Market Study on Data Communications in Europe, 1972-1985," study sponsored by the European Conference of Postal and Telecommunications Administrations, 1973.

FARB 73 Farber, D.J., et al. "The Distributed Computing System," Proceedings of the 7th Annual IEEE Computer Society International Conference, pp. 31-34, February 1973.

FELL 68 Feller, W. An Introduction to Probability Theory and its Applications, 3rd ed., Vol. I, Wiley, New York, 1968.

FRAL 75A Fralick, S.C., and J.C. Garrett. "Technological Considerations for Packet Radio Networks," National Computer Conference, AFIPS Conference Proceedings, Vol. 44, pp. 233-243, 1975.

FRAL 75B Fralick, S., D.H. Brandin, F.F. Kuo and C. Harrison. "Digital Terminals for Packet Broadcasting," National Computer Conference, AFIPS Conference Proceedings, Vol. 44, pp. 253-261, 1975.

FRAN 70 Frank, H., I. Frisch and W. Chou. "Topological Considerations in the Design of the ARPA Computer Network," Spring Joint Computer Conference, AFIPS Conference Proceedings, Vol. 36, pp. 581-587, 1970.

FRAN 72 Frank, H. And W. Chou. "Topological Optimization of Computer Networks,: Proceedings of the IEEE, Vol. 60, No. 11, pp. 1385-1396, November 1972.

FRAN 75 Frank, H., I Gitman and R. Van Slyke. "Packet Radio System-Network Considerations," National Computer Conference, AFIPS Conference Proceedings, Vol. 44, pp. 217-231, 1975.

GERL 73 Gerla, M., W. Chou and H. Frank. "Computational Considerations and Routing Problems for Large Computer Communication Networks," Proceedings of the National Telecommunication Conference, Atlanta, November 26-28, 1973.

GITM 74 Gitman, J., R. Van Slyke and H. Frank. "On Splitting Random Accessed Broadcast Communication Channels," Seventh Hawaii International Conference on System Sciences, University of Hawaii, Honolulu, January 8-10, 1974, Proceedings of the Special Subconference on Computer Nets, 1974.

GRAY 74 Gray, E.M. "ANIK," Communications Society, Vol. 12, No. 1, pp. 3-5, January 1974.

HARD 34 Hardy, G., J. Littlewood and G. Polya. Inequalities, Cambridge University Press, Cambridge, Massachusetts, 1934.

HAYE 71 Hayes, J., and D. Sherman. "Traffic Analysis of a Ring Switched Data Transmission System," Bell System Technical Journal, Vol. 50, No. 9, pp. 2947-2978, November 1971.

HEAR 70 Heart, F., R. Kahn, S. Ornstein, W. Crowther and D. Walden. "The Interface Message Processor for the ARPA Computer Network," Spring Joint Computer Conference, AFIPS Conference Proceedings, Vol. 36, pp. 551-567, 1970.

JACK 69 Jackson, P.E. and C.D. Stubbs. "A Study of Multiaccess Computer Communications," Spring Joint Computer Conference, AFIPS Conference Proceedings, Vol. 34, pp. 491-504, 1969.

JAIS 68 Jaiswal, N. Priority Queues, Academic Press, New York, 1968.

KAHN 75 Kahn, R.E. "The Organization of Computer Resources into a Packet Radio Network," National Computer Conference, AFIPS Conference Proceedings, Vol. 44, pp. 177-186, 1975.

KEND 51 Kendall, D. "Some Problems in the Theory of Queues," J. Royal Statistical Society, Series B, Vol. 13, No. 2, pp. 151-185, 1951.

KING 62A Kingman, J. "The Effect of Queue Discipline on Waiting Time Variance," Proceedings of the Cambridge Philosophical Society, Vol. 58, pp. 163-164, 1962.

KING 62B Kingman, J. "On Queues in Which Customers are Served in Randon Order," Proceedings of the Cambridge Philosophical Society, Vol. 58, pp. 79-91, 1962.

KLEI 64 Kleinrock, L. Communication Nets: Stochastic Message Flow and Delay, McGraw Hill, New York, 1964, out of print. (Reprinted by Dover Publications, New York, 1972.)

KLEI 65 Kleinrock, L. "A Conservation Law for a Wide Class of Queueing Disciplines," Naval Research Logistics Quarterly, Vol. 12, pp. 181-192, 1965.

KLEI 70 Kleinrock, L. "Analytic and Simulation Methods in Computer Network Design," Spring Joint Computer Conference, AFIPS Conference Proceedings, Vol. 36, pp. 569-579, 1970.

KLEI 73 Kleinrock, L. and S.S. Lam. "Packet-Switching in a Slotted Satellite Channel," National Computer Conference, New York, June 4-8, 1973, AFIPS Conference Proceedings, Vol. 42, pp. 703-710, 1973.

KLEI 74 Kleinrock, L. "Resource Allocation in Computer Systems and Computer-Communication Networks," IFIP Conference Proceedings, pp. 11-18, 1974.

KLEI 75A Kleinrock, L. and S.S. Lam. "Packet Switching in a Multi-access Broadcast Channel: Performance Evaluation," IEEE Transactions on Communications, Vol. COM.23, pp. 410-423, April 1975.

KLEI 75B Kleinrock, L. and F.A. Tobagi. "Packet Switching in Radio Channels Part 1 : Carrier Sense Multiple-Access Modes and their Throughput - Delay Characteristics," IEEE Transactions on Communications, Vol. COM.23, No. 12, pp. 1400-1416, December 1975.

KLEI 75C Kleinrock, L. Queueing Systems, Vol. I, Theory, Wiley Interscience, New York, 1975.

KLEI 76 Kleinrock, L. Queueing Systems, Vol. II, Computer Applications, Wiley Interscience, New York, 1976.

KONH 74 Konheim, A.G. and B. Meister. "Waiting Lines and Times in a System with Polling," Journal of the Association for Computing Machinery, Vol. 21, No. 3, pp. 470-490, July 1974.

KUO 73 Kuo, F.F. and N. Abramson. "Some Advances in Radio Communications for Computers," Digest of papers, Seventh Annual IEEE Computer Society International Conference, San Francisco, California, pp. 57-60, February 1973.

LAM 74 Lam. S.S. "Packet Switching in a Multi-Access Broadcast Channel with Application to Satelite Communication in a Computer Network," School of Engineering and Applied Science, University of California, Los Angeles, UCLA-ENG-7429, April 1974.

LITT 61 Little, J. "A Proof of the Queueing Formula $L = \lambda W$," Operations Research, Vol. 9, No. 2, pp. 383-387, March-April 1961.

MART 70 Martin, J. Teleprocessing Network Organization, Prentice-Hall, Englewood Cliffs, New Jersey, 1970.

METC 76 Metcalfe, R.M. and D.R. Boggs. "Ethernet: Distributed Packet Switching for Local Computer Networks," Communications of the ACM, Vol. 19, No. 7, pp. 395-404, July 1976.

MILL 64 Miller, L.W. Alternating Priorities in Multi-class Queues, Ph.D. Dissertation, Cornell University, Ithaca, New York, 1964.

PIER 71 Pierce, J. "Network for Block Switching of Data," IEEE Convention Record, New York, March 1971.

PUEN 71 Puente, J. and A. Werth. "Demand Assignment Service for the INTELSTAT Global Network," IEEE Spectrum, Vol. 8, No. 1, pp. 56-69, January 1971.

RIOR 61 Riordan, J. "Delays for Last-Come First-Served Service and the Busy Period," Bell System Technical Journal, Vol. 40, pp. 785-793, May 1961.

ROBE 70 Roberts, L.G. and B.D. Wessler. "Computer Network Development to Achieve Resource Sharing," Spring Joint Computer Conference, AFIPS Conference Proceedings, Vol. 36, pp. 543-549, 1970.

ROBE 72 Roberts, L.G. "ALOHA Packet System With and Without Slots and Capture," ARPA Network Information Center, Stanford Research Institute, Menlo Park, California, ASS Note 8 (NIC 11290), June 1972.

ROBE 73 Roberts, L.G. "Dynamic Allocation of Satellite Capacity Through Packet Reservation," National Computer Conference, New York, June 4-8, 1973, AFIPS Conference Proceedings, Vol. 42, pp. 711-716, 1973.

ROBE 74 Roberts, L.G. "Data by the Packet," IEEE Spectrum, Vol. 11, No. 2, pp. 46-51, February 1974.

ROSS 70 Ross, S.M. Applied Probability Models with Optimization Applications, Holden-Day, San Francisco, 1970.

SPRA 72 Spragins, J. "Loop Transmission Systems - Mean Value Analysis," IEEE Transactions on Communications, Vol. COM-20, No. 3, pp. 592-602, June 1972.

TAKA 63 Takács, L. "Delay Distributions for One Line with Poisson Input, General Holding Times, and Various Orders of Service," Bell System Technical Journal, pp. 487-503, March 1963.

TELE 73 Telenet Communications Corporation. "Before the Federal Communications Commission, Application for Authority to Establish and Operate a Packet-Switched Data Communications Network in the Continental United States," Washington, D.C., October 1973.

TOBA 74 Tobagi, F.A. "Random Access Techniques for Data Transmission over Packet Switched Radio Networks," School of Engineering and Applied Science, University of California, Los Angeles, UCLA-ENG-7499, December 1974.

TOBA 75 Tobagi, F.A. and L. Kleinrock. "Packet Switching in Radio Channels: Part II - The Hidden Terminal Problem in Carrier Sense Multiple-Access and the Busy Tone Solution," IEEE Transactions on Communications, Vol. COM-23, No. 12, pp. 1417-1433, December 1975.

TOBA 76A Tobagi, F.A. and L. Kleinrock. "Packet Switching in Radio Channels: Part III - Polling and (Dynamic) Split-Channel Reservation Multiple Access," IEEE Transactions on Communications, Vol. COM-24, No. 8, pp. 832-845, August 1976.

TOBA 76B Tobagi, F.A. and L. Kleinrock. "Packet Switching in Radio Channels: Part IV - Stability Considerations and Dynamic Control in Carrier Sense Multiple Access," submitted to IEEE Transactions on Communications.

TOBA 76C Tobagi, F.A. and L. Kleinrock. "On the Analysis and Simulation of Buffered Packet Radio Systems," Ninth Hawaii International Conference on System Sciences, University of Hawaii, Honolulu, January 1976, Proceedings of the Special Subconference on Computer Nets, 1976.

VASI 76 Vasicek, O.A. "An Inequality for the Variance of Waiting Time Under a General Queueing Discipline," Graduate School Of Management, University of Rochester, New York, April 1976.

VAUL 54 Vaulot, E. "Délais d'attente des appels téléphoniques dans l'ordre inverse de leur arrivée," Comptes Rendus de l'Academie des Sciences, Paris 238, pp. 1188-1189, 1954.

WILL 73 Willard, D.G. "Mitrix: A Sophisticated Digital Cable Communications System," Proceedings of the National Telecommunications Conference, November 1973.

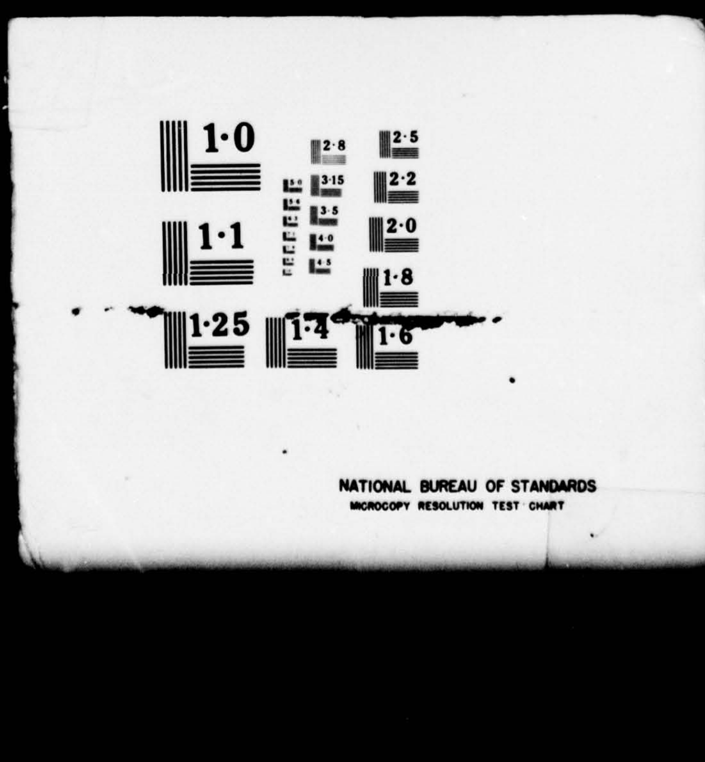
END

AD-A039 016 CALIFORNIA UNIV LOS ANGELES DEPT OF COMPUTER SCIENCE F/G 9/2
ADVANCED TELEPROCESSING SYSTEMS.(U)
DEC 76 L KLEINROCK DAHC15-73-C-0368
UNCLASSIFIED NL

4 OF 4
ADA
039018

SUPPLEMENTARY
INFORMATION

4 OF 4
ADA
039018



SUPPLEMENTARY

INFORMATION

DAHC15-73-C-0368

Note: This page replaces the current page 173 in the December 1976 SATR.

AD-A039018

throughput at the large user, at the expense of a large degradation of the small users' performance. The maximum achievable throughput Λ_2 depends on Λ_1 and N: If N = 25 and Λ_1 = .16 (S_1 = .2) (Figure 6.11), then $\Lambda_2 \leq .84$ (instead of .63 with MAPCS). But to achieve a limiting throughput of .84, we have to split the channel so that $\alpha \geq .84$, increasing then the delays at the small users by a factor of approximately 5.

b) The choice of the access mode (MAPCS or split channel) will depend on the small users' traffic. If the latter is high, splitting the channel will not significantly degrade the small users' delay: If Λ_1 = .56 (S_1 = .7) (Figure 6.12), by choosing α = .4 we approximately double the delay for the small users; we have a better delay for the large user for most values of Λ_2 and we may achieve a large user throughput of .44 (instead of .23 with MAPCS). If the small users' traffic Λ_1 is small (Figure 6.11), splitting the channel degrades the small users' performance quite significantly.

Therefore, compared with a split channel mode, MACS and MAPCS differ in the following respects:

1) For N = 25, MAPCS achieves a smaller channel utilization than the split mode*, while MACS achieves a larger channel utilization than the split mode.

2) The larger N is, the smaller is the maximum achievable throughput with MAPCS. The large user throughput is greater with MACS than with MAPCS if N is not too small ($N \geq 30$) and $\Lambda_1 < 1/e$ (see next

*The larger N is, the smaller is the channel utilization with MAPCS.