A DEMAND ASSIGNED MULTIPLE ACCESS STRATEGY

FOR LAND MOBILE SATELLITE VOICE SERVICES

by

CHRIS J. POWELL

BSc. (Hons), The University of British Columbia, 1989

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF APPLIED SCIENCE

in

THE FACULTY OF GRADUATE STUDIES

Department of Electrical Engineering

We accept this thesis as conforming

to the required standard

THE UNIVERSITY OF BRITISH COLUMBIA

March 1992

In presenting this thesis in partial fulfilment of the requirements for an advanced degree at the University of British Columbia, I agree that the Library shall make it freely available for reference and study. I further agree that permission for extensive copying of this thesis for scholarly purposes may be granted by the head of my department or by his or her representatives. It is understood that copying or publication of this thesis for financial gain shall not be allowed without my written permission.

(Signature)
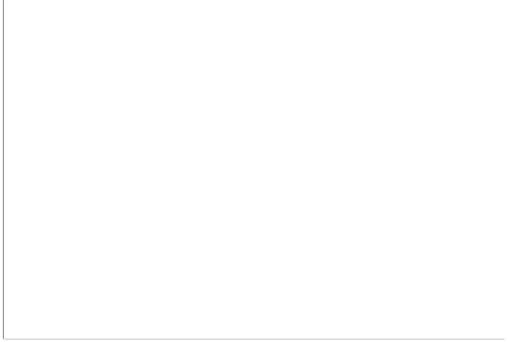
Department of ___ Electrical Engineering ___

The University of British Columbia
Vancouver, Canada

Date ___ March 11, 1992 ___

DE-6 (2/88)

Abstract

Development of land mobile satellite systems is progressing rapidly, and implementation of new voice and data services for North America is scheduled for early 1994. In order to make the best use of the bandwidth allocated for these services, efficient demand assigned multiple access (DAMA) protocols must be employed. Efficiency achieved in terms of higher channel utilization and availability translates into more revenue for network management and better service for network subscribers.

In particular, mobile voice services are examined, and a new blocked-calls-queued discipline, which processes calls in batches, designed specifically for dispatch radio via satellite is analyzed. Computer simulation is used to examine several batch service disciplines, and it is thereby shown that the new system meets the objectives of efficiency in providing a high level of performance by exploiting traffic characteristics unique to dispatch radio rather than adapting conventional techniques used in telephony.

In addition, a technique for integrating mobile radio service and mobile telephone service in a dynamic resource sharing strategy using a common DAMA channel pool is introduced. The new strategy attempts to reserve a small margin of free channels for blocked-calls-dropped telephone traffic, while permitting the remaining channels to be shared between radio and telephone on a first come first served basis. It is shown that, without degrading the performance of either traffic source, the proposed integrated system achieves higher throughput than any other system of traffic integration found in the literature which is applicable to these services.

Contents

List of Tables

List of Figures

List of Symbols

0 — a matrix consisting entirely of zeros

AMSC — American mobile satellite corporation

b — batch size

CB — closed batch

c_R — number of radio channels for a fixed channel allocation mobile radio network

c_T — number of telephone channels for a fixed channel allocation mobile telephone network

DAMA — demand assigned multiple access

DCS — DAMA control centre

DOC — Department of Communications

d — number of dispatchers

$E[\ ]$ — expected value

e_i — column matrix which contains all zeros except in position i where there is a 1

FAX — facsimile

FDM — frequency division multiplexing

FCFS — first come first served

GHz — gigahertz

i — denotes an index

INMARSAT — International Marine Satellite Organization

j — denotes an index

Ku-band — 12/14 GHz

L-band — 1.5/1.6 GHz

LMR — land mobile radio

LMSS — land mobile satellite services

m — number of mobile telephone terminals

MDS — mobile data service

MHz — megahertz

min — minute

MRS — mobile radio service

MRT — mobile radio terminal

ms — millisecond

MSAT — mobile satellite

MSS — mobile satellite systems

MTS — mobile telephone service

MTT — mobile telephone terminal

n — number of mobile radio terminals

N — mean number of customers in a Markov chain

NASA — National Aeronautics and Space Administration

NCC — network control centre

n — number of mobile radio terminals

OB — open batch

OJF — oldest job first

$\mathbf{p}$ — state probability matrix

p_B — telephone service blocking probability

p_i — the limiting probability of state i in an imbedded Markov chain

p_{ij} — the limiting probability of state ij in an imbedded Markov chain

p.d.f. — probability density function

PSTN — public switched telephone network

$\mathbf{Q}$ — infinitesimal generator matrix

R — mean number of free telephone channels realized in an integrated network after mobile radio service

traffic has been increased as much as possible, while mobile telephone traffic has been held constant

RM — specified margin of free telephone channels to be reserved

RPS — remote position sensing

s — second

s — number of spot-beams

S_i — a Markov chain state

S_{ij} — a Markov chain state

SCADA — supervisory control and data acquisition

T — mean number of free telephone channels realized in an integrated network after mobile telephone service traffic has been increased as much as possible, while mobile radio traffic has been held constant

T — mean time spent in system by a customer in a Markov chain

T_C — mean time between the time a dispatch radio call is initiated and the time service commences excluding the mean time to wait for a free channel to come available at the DAMA control center

T_{CA} — mean time for a channel assignment to reach a dispatcher

T_{CB} — mean time for a channel request to reach the DAMA control center

Tm — control message serializing delay

TMI — Telesat Mobile Incorporated

Tp — earth station to earth station propagation delay

T_Q — mean time spent in queue by a customer in a Markov chain

t_{QC} — mean time to wait for a free channel to come available at the DAMA control center

T_{QC} — mean wait for a channel allocation *per call*

T_R — mean time for the call request to reach the dispatch centre

T_S — mean time to complete service for leading calls in the same batch

T_{Setup} — mean time between the time a dispatch radio call is initiated and the time service commences

T_V — mean time to verify a mobile radio terminal is set up on an assigned channel

T_{V_A} — mean verification delay for an open batch call joining a batch for which service has not begun

T_{V_B} — mean verification delay for an open batch call joining a batch for which service has begun

T_{WB} — mean waiting time associated with batch formation

T_{WC} — mean time waiting to receive a channel assignment

T_{WD} — mean time waiting for a dispatcher to service the batch

WARC — World Administrative Radio Conference

$\bar{x}$ — mean of the random variable X

X — random variable representing the number of open batch calls that arrive after batch service starts but have still not commenced service

$\Delta\%$ — change in percent

Δm — change in the number of mobile telephone terminals

Δn — change in the number of mobile radio terminals

Λ — mean overall call arrival rate for a Markov chain

λ — mean arrival rate for a Markov chain client which is not queued or in service

λ_R — mean individual mobile radio terminal call rate

λ_T — mean mobile telephone call rate

μ_R — mean rate of mobile radio call service for a single channel

μ_T — mean rate of mobile telephone call service for a single channel

π_i — the limiting probability of state i in a continuous time Markov chain

π_{ij} — the limiting probability of state ij in a continuous time Markov chain

ρ — channel or server utilization factor

Acknowledgment

I would like to express my thanks to my advisor, Dr. Victor Leung, whose support and guidance throughout my studies is sincerely appreciated.

I would also like to express sincere thanks to the technical support staff for their help in answering many questions and keeping the computer network up and running, the women in the Electrical Engineering office for their frequent administrative assistance, and to many other new friends who have made my stay a pleasant one.

The financial support I received is also gratefully acknowledged. Many thanks to Dr. Leung for his Research Assistantships, the Department of Electrical Engineering for its scholarships and Teaching Assistantships, and to the British Columbia Science Council and Microtel Pacific Research for their contributions toward my Graduate Research in Engineering and Technology award.

Finally, the support and encouragement I received from my family and friends has been invaluable — especially from my parents who were with me at the start of this endeavour and without whom it would not have been possible.

Dedication

In Memory of my Parents,

Francis H. and Hilda G.

1. Introduction

1.1 Background

Satellite technology has progressed enormously since its inception. In the early days of satellite communication, the expense was high for both service suppliers and system users. Today the cost of the space segment is still high, but due to advanced technology offering affordable terrestrial transceivers and a rapidly expanding demand for mobile communications, services can be offered to a large, diverse population at a reasonable cost [1].

1.1.1 Historical Perspective

In 1945, twelve years before the first Sputnik was launched, Arthur C. Clarke, a radar officer with the Royal Air Force, first proposed communication systems using geosynchronous satellites [2,3]. Since that initial vision, satellite systems have burgeoned from dream to reality.

The American National Aeronautics and Space Administration (NASA) first tested voice transmission from a satellite by broadcasting a tape-recorded message in 1958. Two years later the first active radio repeater was in orbit; had three watts of power and lasted only seventeen days. Finally, in 1963, the idea created in the imagination of Arthur C. Clarke was realized as NASA launched the first geosynchronous communications satellite into orbit. From that time, satellite technology continued to progress rapidly, and in 1966 the first multiple access satellite with multidestination capability was introduced by COMSAT in the United States [4].

1

Despite the fact that satellite communications were very expensive at the time, the 1971 World Administrative Radio Conference (WARC-71) allocated 1 MHz of L-band (~1.6/1.5 GHz) spectrum to maritime and aeronautical communications to open the way for the development of mobile satellite systems (MSS). Following the WARC-71 L-band allocation, the Canadian Department of Communications (DOC) developed the concept of the first multiple purpose MSS and, in 1972, Telesat Canada launched the ANIK — the world's first domestic communications satellite. The DOC undertook the capital intensive ANIK project in its commitment to Canadian national communications; nevertheless, the DOC hoped that by achieving full utilization, economies of scale would make the project end up being economical [5]. As it turned out, the ANIK project showed a virtually unprecedented return on capital investment in the telecommunications industry. In the US at that time, legislation was in place which prohibited domestic satellites and restricted US satellite service providers to COMSAT alone. However, following the ANIK's resounding success, a fury of US legislation was passed that same year which broke the US domestic market wide open. By 1974 the first US domestic satellite was operational, and within months two additional domestic US satellites from two different carriers joined the first [4,6]. With the feasibility of satellite communications firmly established, a new age of satellite communications began.

In 1975, several Canadian government agencies were participating in the establishment of the first maritime MSS by the International Maritime Satellite Organization (INMARSAT). A continued interest in MSS resulted in the DOC's formation of Telesat Mobile Incorporated (TMI) for the development and implementation of mobile satellite (MSAT) technology in Canada [5]. Comprehensive market studies and business plans were compiled by NASA in the US in 1983 and by the DOC in Canada in 1986, and pursuant to these, decisions

were made in both countries to proceed in a cooperative effort to provide commercial land mobile satellite systems (LMSS). The decisions to proceed were made despite the fact that there was limited spectrum available for the proposed services. It was hoped that WARC-87 would allocate sufficient L-band spectrum to easily accommodate LMSS, but despite extensive lobbying by Canada and the US, only 14 MHz (on a primary or co-primary basis) was allocated in WARC-87 (Figure 1) [7–10], which fell short of that deemed a minimal requirement by Canada and the US for LMSS in North America [11].

Downlink

1530   1533   1544   1555   1559

Uplink

1626.5   1631.5   1634.5   1645.5   1656.5   1660.5

■ LMSS Primary or Co-primary   ▨ LMSS Secondary   ☐ Not LMSS or Not allocated
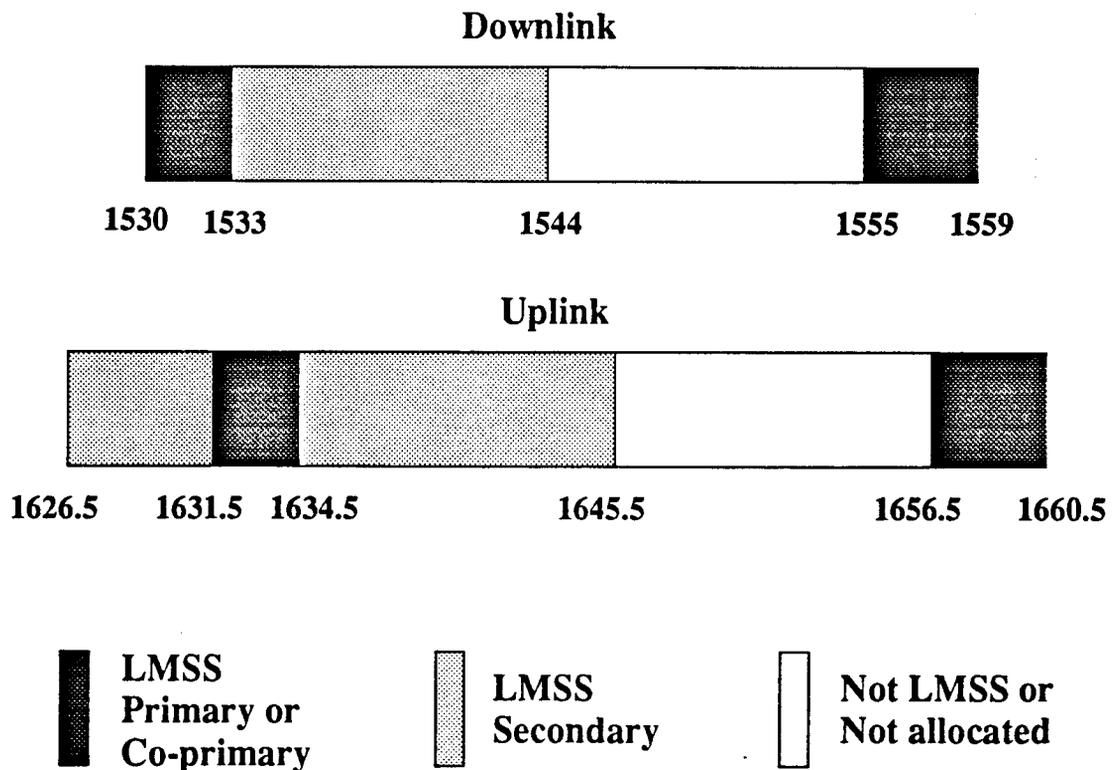
Figure 1 WARC-87 L-band Allocation (MHz)

1.1.2 Current LMSS Development

Today LMSS are under development on virtually every continent, and services are scheduled for introduction as early as 1994 [10–12]. In North America, a synergy of Canada's TMI and the American Mobile Satellite Corporation (AMSC) has resulted in conciliatory

agreements to provide mutually compatible and complementary services to Canada and the US by their respective carriers [10,13]. System compatibility allows each system to provide backup for the other, and provides for uninterrupted service to mobile vehicles roaming across international boundaries [10–14].

North American LMSS will provide a full range of voice and data services to mobile vehicles in rural and sparsely populated areas, and complement terrestrial systems predominant in more densely populated areas [15,16]. In addition, LMSS will service light aircraft, and coastal and inland marine applications. Target markets encompass all areas of the private sector and all levels of government. Among the services to be offered initially are mobile radio service (MRS) for private subnets with closed user groups, mobile telephone service (MTS) as an extension to the public switched telephone network (PSTN), mobile data service (MDS) for remote position sensing (RPS), supervisory control and data acquisition (SCADA), national paging, and facsimile (FAX), although services may be modified or augmented as market demand warrants [10,18].

1.1.3 Typical LMSS Network Configuration

A typical LMSS network (Figure 2) consists of a collection of mobile terminals communicating with terrestrial stations or other mobile terminals via fixed earth stations over a geosynchronous satellite [8,9]. The satellite provides connections between L-band links to mobile terminals and Ku-band links to fixed earth stations. The L-band footprint of the satellite will have multiple spot-beams to increase transmit and receive power and to facilitate frequency reuse. It is expected that in North America, a modest frequency reuse factor of between 1.3 and 1.7 is achievable [15,19,20].
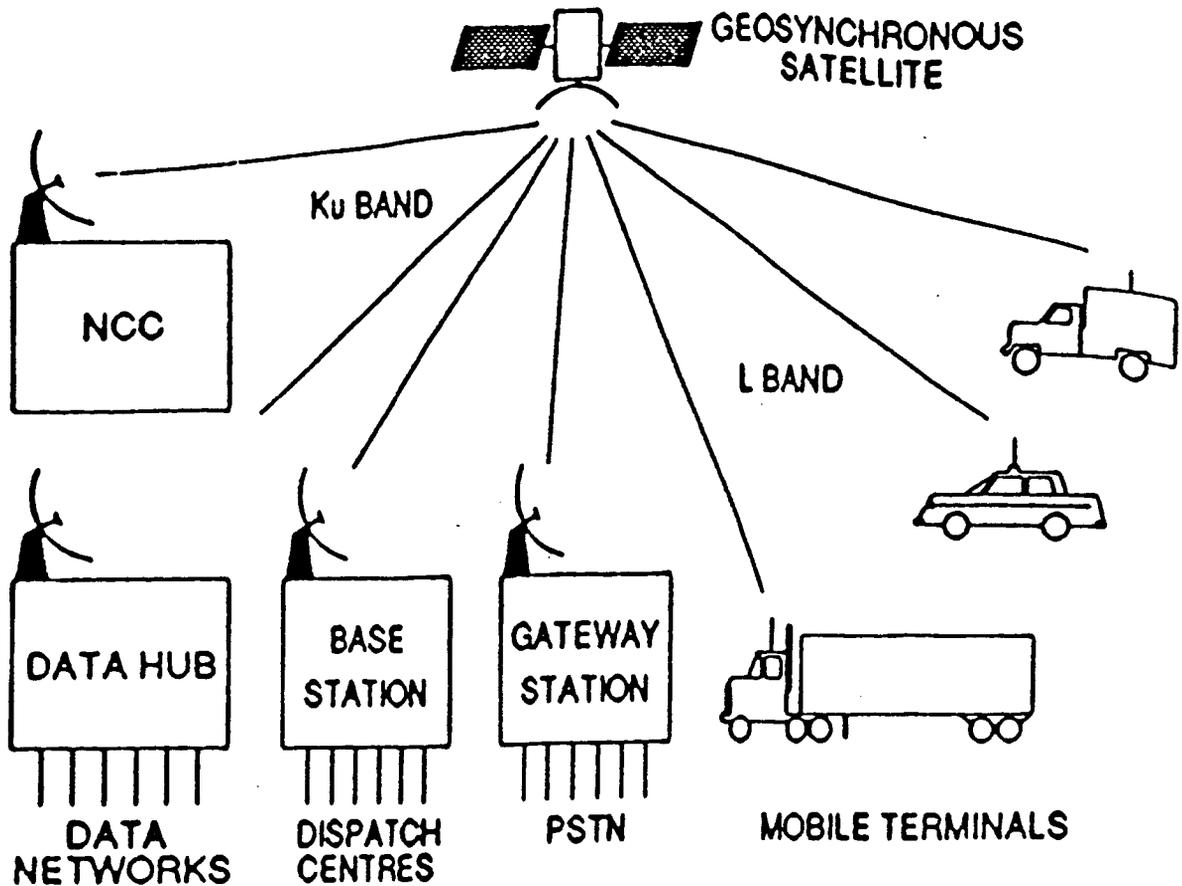
Figure 2 Network Configuration

A possible configuration for North American spot-beam coverage is illustrated in Figure 3, where each spot-beam is the area covered by a single satellite transponder. Fixed earth stations include gateways, base stations, and data hubs which provide interfaces to the PSTN, private voice dispatch networks, and data networks respectively. In addition, the network control centre (NCC) is also a fixed earth station. Functions of the NCC include network surveillance, performance monitoring, system administration, service billing, and the management of satellite transponder resources [9,15,21,22]. The NCC also incorporates a DAMA control system (DCS) to handle call processing and channel assignment functions as specified by MRS and MTS voice service protocols [9,23].

5
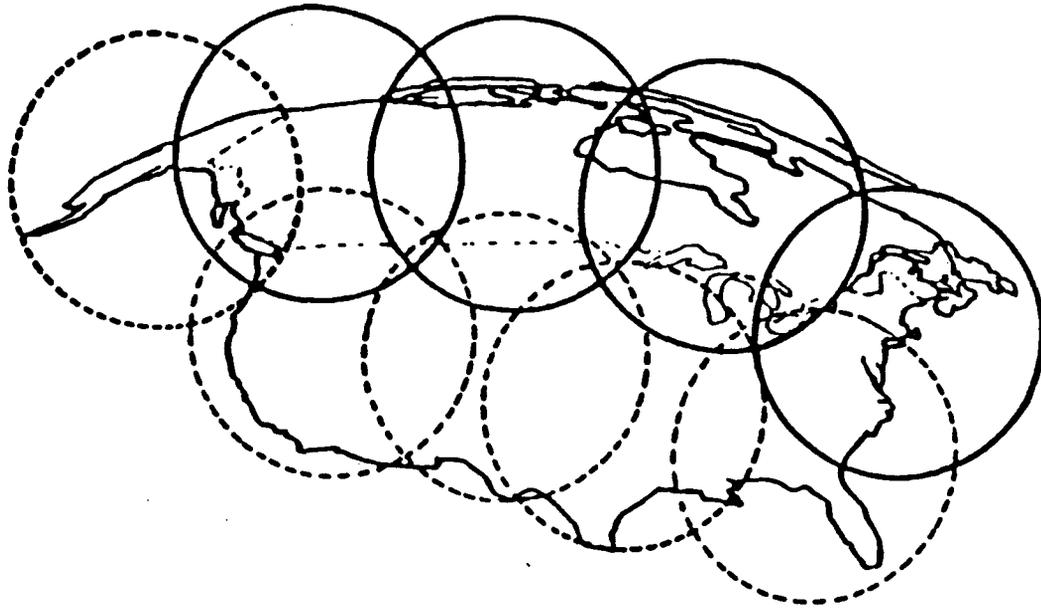
Figure 3 North American Spot-beam Coverage

The NCC communicates with mobile terminals over L-/Ku-band signalling links, and with other fixed earth stations over Ku-band signalling links for management and call control purposes. Mobile telephone service is offered to each mobile telephone terminal (MTT) user on an individual basis. Mobile radio service, on the other hand, accommodates private dispatch subnets, and service is extended to each individual mobile radio terminal (MRT) through its associated dispatch centre. Each dispatch centre is, in turn, attached to an appropriate base station which is under DCS control.

1.2 Motivations and Objectives

1.2.1 Motivations

A need for LMSS services has been firmly established, and government and industry world wide are committed to providing first generation services in the near future. Even the

most conservative projections for the first generation of Canadian LMSS estimate over 60,000 voice service subscribers. Worldwide, twelve satellites dedicated to MSS are scheduled for launch by the mid-1990s, and global estimates well in excess of one million terminals within five years of initial implementation are common [13,24]. Of the total number of terminals, voice users are expected to constitute thirty to forty percent, most of whom are likely to be MRS subscribers.

Faced with a large, rapidly expanding demand for LMSS, and limited available spectrum, efficient utilization of satellite transponder bandwidth is essential [10,22,25,26]. For circuit switched voice services, DAMA protocols are particularly suitable, as they provide a satisfactory grade of service for a large number of users with respect to the number of available channels. The major advantage of DAMA protocols is that they restrict resource allocation to only those users which have immediate requirements.

1.2.2 Objective 1: MRS DAMA Strategy

The objective is to evaluate the delay performance for a recently proposed blocked-calls-dropped, batch processing MRS DAMA protocol in [9] under the above network and spot-beam configurations (Figures 2 and 3). Various numbers of dispatchers and several batch service disciplines are examined. In particular, the objective of analyzing the MRS DAMA strategy is to show that the scheme for handling dispatch radio traffic which is based on the blocked-calls-queued service discipline, makes more efficient use of available bandwidth than other techniques proposed to date. Efficiency is measured by how well the overall performance objectives of both system and user are met.

From a users perspective, performance is based on ease of access to network channel resources. Since blocked MRS calls are queued for service, the measure of user performance

is given by the mean delay between call initiation and the time service commences [27]. Network management, on the other hand, endeavors to maximize its return on investment. While in theory, the latter is roughly equated with throughput (ie. maximizing carried traffic load), revenue will decrease if users find services unsatisfactory. Thus, there is often a trade-off between the amount throughput and the desired level of user performance.

If both the throughput and delay vary, it may be difficult to determine how effective a DAMA system works, particularly if one aspect improves while the other declines. Therefore, the approach taken in determining the amount of efficiency gained by the new scheme is to choose a fixed level of user performance criteria, then observe the amount of increased throughput obtained by the new system.

1.2.3 Objective 2: Integrated Network Strategy

The objective here is to find an appropriate technique for radio and telephone traffic integration and to evaluate its performance improvements in order to further satisfy LMSS performance criteria. A new technique for integrating MRS and MTS under a single dynamic resource sharing system using a common DAMA channel pool is introduced. Analogous to the previous discussion, the objective of the analysis of the proposed integrated strategy is to show that it is more efficient than any other applicable channel allocation scheme found in the literature. A similar approach is taken with the integrated strategy also — desired levels of user performance are selected, then it is shown how additional throughput may be achieved by employing the integrated strategy over the fixed channel allocation method.

1.2.4 Overall Contribution to LMSS Communications

The current demand for LMSS and the overwhelming potential magnitude of LMSS services combined with the scarcity of channel resources clearly indicates the necessity for

the development of efficient DAMA protocols of a highly practical nature. The new DAMA protocol for MRS and the introduction of an efficient method for integrating heterogeneous sources of voice traffic constitute fundamental contributions to MSAT communications in both efficient and practical resource management.

1.3 Review of Previous Work

Since there are two major topics covered in this thesis, it is appropriate that an exposition of previous work be made in two separate divisions. Hence, subsequent subsections of this section deal separately with the new MRS protocol and the proposed technique for heterogeneous traffic integration. Before discussion commences, however, some background on MRS and MTS call characteristics is necessary.

1.3.1 Comparison of MRS and MTS Traffic

The MRS segment of LMSS will serve a collection of independent closed user group dispatch subnets [28]. On the other hand, the MTS segment will serve a number of independent telephone subscribers. Dissimilar applications give rise to different service requirements. These differences are reflected in a variation of call characteristics between the two services.

Differences between MRS and MTS traffic are in call duration, call frequency, and network connectivity. Telephone traffic typically has a mean call holding time of 2–3 minutes [29–31] while dispatch calls vary between 8–30 seconds, depending on application [32–34]. While radio call holding times are generally shorter, they are often more frequent, and thus total offered traffic (measured in Erlangs) per MRT may be similar to that offered by each MTT. Connectivity of MTS, like conventional telephony, is variable as a large population of users generates calls independently [35]. However, MRS caters to private dispatch

9

applications with closed user groups where connectivity is inherently fixed. Furthermore, MRS communication is generally between individual MRTs and their respective dispatcher(s), unlike that of MTS which is between mobiles or a mobile and a PSTN subscriber [10,21].

In addition to differences in traffic characteristics, MRS and MTS calls are serviced differently. Blocked MRS calls are queued for service at their respective dispatchers, whereas blocked MTS calls are dropped. Since no dispatch radio calls are dropped, the primary measure of MRS performance is given by the mean delay between the time when a call is initiated and the time service commences [27]. On the other hand, MTS performance is determined by its blocking probability — the probability that an attempted call finds all channels busy and is subsequently dropped [15].

1.3.2 Mobile Radio Service DAMA Protocol

In general, DAMA protocols require signalling for call management through call request, channel allocation, channel verification, and channel release upon call termination [36]. Signalling represents overhead in system operation and costs in terms of revenue-producing traffic carrying capacity. Therefore, efficient DAMA protocol design is concentrated in two areas — minimizing signalling overhead, and maximizing the utilization of the demand assigned channel pool under constraints of customer satisfaction.

Conventional methodology for handling radio dispatch systems is to treat radio calls in the same manner as telephone calls, by employing a blocked-calls-dropped service discipline [9,30]. However, when this technique is adapted to MRS, private dispatch networks endeavouring to make efficient use of their respective dispatchers find highly utilized dispatchers forming a resource bottleneck; this translates into a high blocking probability for a blocked-calls-dropped radio service. In this situation, calls are not only

10

limited by available channel resources, but also by the availability of their own subnet dispatchers [35]. Due to the higher call rate of MRTs over MTTs, repeated call requests from blocked MRTs result in an excessive number of retries, and hence wasted signalling capacity [9]. Signalling is particularly costly in satellite communications where propagation delay is inherently long. Therefore, the handling of MRS by traditional telephony techniques is unacceptable for LMSS applications. Nevertheless, very little research has been dedicated specifically to modelling and analysis of land mobile radio (LMR) traffic [35]; however, that which is found in the literature is now summarized.

Deferred channel assignment has been proposed as one possible modification to conventional call handling [23,37]. However, deferred channel assignment is not applicable to LMSS. In the case of MTS, deferred channel assignment, which does not assign a channel until the called party is off hook, is unacceptable since the one who initiates a call from the PSTN is charged for the PSTN segment of the call even though the LMSS network might not be able to complete its portion of connection [9]. This type of deferred channel assignment is also inapplicable to MRS since all calls *must* go through a dispatch center. Dispatchers only make channel requests to serve MRTs that have call requests pending so they could not possibly be busy or unavailable. Thus, anytime the DCS responds to a dispatcher channel request, there is certainty that the connection can be completed end to end. However, call queuing at the dispatcher could be considered an alternate form of deferred channel assignment since a channel is not assigned when an MRT initiates a call but when a dispatcher is able to handle the call.

Quite a number of new protocols were proposed during the upshot of terrestrial cellular communications, particularly in the early 1970s through to the early 1980s in the design

11

and implementation stages of cellular development. However, methodology arising from the cellular revolution is directly related to conventional terrestrial telephony and, therefore, inherits the same shortcomings when adapted to MRS, because the techniques are still based on the assumptions of a large independent population, longer less frequent call characteristics, and variable connectivity. Among those strategies, a number propose a blocked-calls-queued discipline based on the Erlang-C traffic model [31,38–41].

Although the Erlang-C model could be considered appropriate for telephone traffic, where there is a large independent population, it does not accurately represent MRS which consist of a collection of closed user groups. While the total MRT population is large, each subnet has a small population which must contend for proprietary dispatcher service [35]. Nevertheless, call queueing is appropriate for mobile dispatch applications since it eliminates the need for repeated call requests, and thereby saves on signalling overhead. Accordingly, the new protocol outlined in the next section incorporates call queuing as a fundamental component of its design.

In addition to work done in mobile telephony, some research has been done trunk radio applications. With regard to subnet configuration, some trunk radio applications are very similar to proposed LMSS dispatch radio services and therefore tend to be more relevant than telephony models. However, the underlying assumption of Erlang-C traffic is often the basis for modelling of trunk radio applications as well [32,35,33]. Furthermore, trunk radio applications typically employ a drop out phase for reclaiming idle channels [42], where the drop out period is usually significantly less than round trip propagation delay for geosynchronous satellite communications, so the corresponding methods are inappropriate for LMSS. Finally, none of the systems attempt batch processing to save on signalling

12

overhead, which is particularly relevant in a satellite environment where long propagation delays warrant special consideration. Therefore, a need for further research concentrated in this area still exists.

1.3.3 Integrated MRS and MTS Dynamic Channel Allocation

The objective of integrating two separate networks into one is that, in combination, better performance can be realized by both services [32]. Alternatively, it would be acceptable to improve the performance of one service, provided that the other suffered no service degradation. It is, however, important that a resource sharing strategy not only achieve the objective, but achieve it in a simple and practical manner. Simplicity is of paramount importance in implementing satellite systems [27,36].

There have been several attempts to model systems with heterogeneous traffic sources. Of those systems which are applicable to voice traffic, there are two distinct types. One type assumes Erlang-B traffic is generated from both sources, although each source is assumed to have different call characteristics [43]. However, as previously discussed, the blocked-calls-dropped service discipline is inappropriate for dispatch applications. The other type of system assumes Erlang-B traffic for telephone calls and Erlang-C traffic for dispatch radio. The Erlang-B is an acceptable model for telephone traffic, but once again, the Erlang-C model does not accurately represent the M. y system being modelled. Nevertheless, problems faced when integrating an Erlang-B (telephone) and an Erlang-C (dispatch radio) system are relevant to the proposed strategy, because a common problem is encountered when blocked-calls-dropped and blocked-calls-queued disciplines are combined.

The problem is most simply explained by examining a simple method for combining the two traffic sources. Accordingly, the most obvious technique is considered which is to open

13

all available channels to both services on a first come first served basis. This system has the advantage of being very simple; however, even when the two heterogenous systems perform well independently, the combined system exhibits undesirable behavior. An acceptable level of performance is maintained by both traffic sources in such a network only when overall traffic is low. When loads are increased, the Erlang-C source often maintains an acceptable level of performance but only at the expense of the Erlang-B traffic. As observed through simulation modelling in [44], performance degrades quickly for telephone traffic even at a modest channel utilization level of about 70 percent. Since a queue is most often present at higher Erlang-C channel utilization levels, most free channels are consumed by queued dispatch calls, leaving few available for the blocked-calls-dropped telephone traffic. In a similar analytical model developed in [45], telephone traffic is again shown to deteriorate rapidly as most of the available channels are seized by queued traffic.

One successful attempt to circumvent this problem was proposed by Peritsky [46][45]. Peritsky tries to maintain a more evenly balanced system by introducing an artificial delay, τ, to the dispatch traffic to afford a chance for incoming telephone calls to pick up a free channel. Peritsky's premise is that forcing incoming dispatch calls to delay for a short time before requesting a channel gives an incoming Erlang-B call a chance to obtain a free channel before the channel is seized by Erlang-C traffic which most always has queued calls pending. If, after an initial delay, a channel is not available to serve a queued dispatch call, the lead call in the queue must wait another τ seconds before once again attempting to seize a free channel. Peritsky also shows that an optimal τ may be estimated effectively.

Some other attempts to integrate heterogeneous traffic sources have been presented [32,43,44,47]; however, none others have been found which actually show improvements

over separate channel allocation schemes. Furthermore, most are complex or make unrealistic assumptions in order to focus on some special case for which promise is shown. The reserved margin strategy proposed in chapter 4, on the other hand, not only works well for the proposed LMSS network but can also be applied directly to the more classical problem of combining the ubiquitous Erlang-B and Erlang-C traffic models, and in fact, is the best solution for solving this problem to date. A comparison between Peritsky's method and the newly proposed network integration strategy may be found in Chapter 4.

1.4 Overview

A complete description and thorough analysis of the proposed MRS DAMA system are given in Chapters 2 and 3 respectively. Due to the highly intractable nature of the queuing models required for closed form analytical results, much of the protocol analysis is done through computer simulation modelling. This a common technique since the derivation of steady-state distributions is rarely possible with new communication system models [48]. However, when applicable analytical models or numerical results are available, they are used in simulation model verification. Chapter 4 looks at the proposed strategy for integrating MRS and MTS in dynamic shared channel allocation. Methodology is similar to that for the MRS system. The last chapter, Chapter 5, contains a summary and discussion of Chapter 3 and Chapter 4 results, and includes suggestions for further research and development.

Verification of simulation models, including analytic model comparisons, is covered in Appendix A. Appendix B includes a brief description of general simulation modelling methodology and discusses the generation of confidence intervals for simulation results. Finally, Appendix C includes some examples of simulation model source code.

2. Description of MRS Protocol

The new MRS protocol employs a blocked-calls-queued discipline and batched call processing. By employing a call queuing discipline, the proposed protocol saves signalling overhead attributed to frequent retries of blocked calls. A further reduction in signalling overhead achieved by the batch processing scheme is demonstrated below.

2.1 Dispatch Service Disciplines for Batch Processing

Batch processing is where dispatcher service for calls under the same spot-beam, and belonging to the same subnet, are delayed until a predetermined threshold, with respect to the number of calls, is reached. Once the threshold is reached, a dispatcher request is made to process the entire batch of calls in succession by pipelining. If a dispatcher is not immediately available to process the new batch, it must wait in queue for the first available dispatcher. Figure 4 depicts a model of the batch formation and dispatcher queuing process: notice how batches are formed independently under each spot-beam, but batches are channelled into the same queue for dispatcher service; there is no distinction between subnet dispatchers — any dispatcher belonging to a particular subnet may serve any batch belonging to the same subnet, regardless of the spot-beam under which the batch is formed.
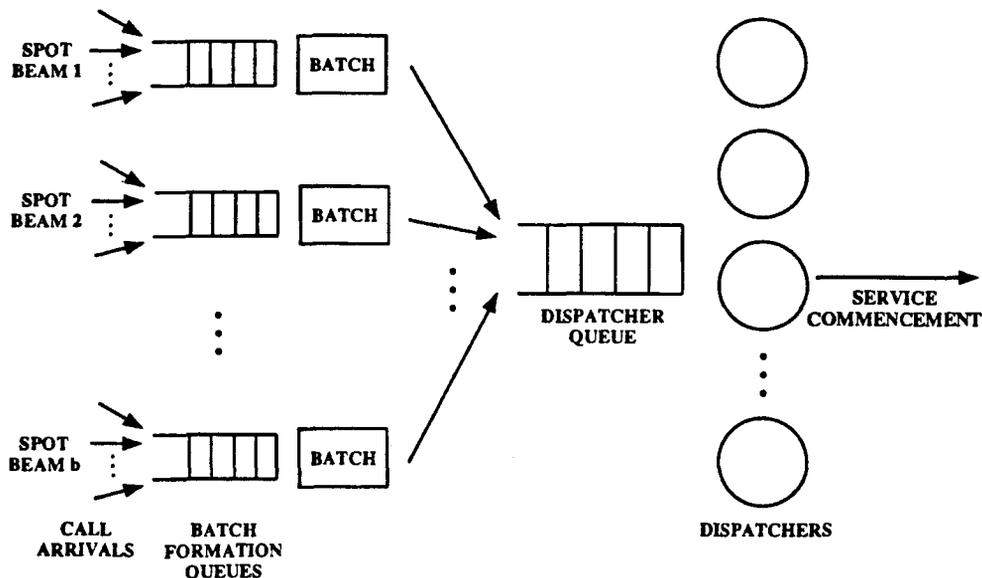
Figure 4 Batch Formation Queuing Diagram

The rationale for batch processing is that savings in signalling overhead can be achieved by eliminating some inband signalling requirements for channel assignment, confirmation, and relinquishment. A timing diagram for single call channel assignment is shown in Figure 5. Single call channel assignment corresponds to a batch size of one, and is also applicable to MTS call servicing where each call is handled individually. Figure 6 presents a timing diagram for more general batch call servicing, where the batch size may be greater than unity. With regard to the two timing diagrams (Figures 5 and 6), T_m is the control message serializing delay, T_p is the propagation delay between two earth terminals, and b is the batch size.

Four alternative batching servicing strategies are considered here, although there are many possible variations on these. The alternatives presented here encompass two broad classes: closed batch (CB) and open batch (OB). In either system a channel is requested when a batch is formed, and once a channel is seized, it is not relinquished until all calls

in the batch have been processed. In a CB system, if subsequent calls arrive during the processing of a batch, the new arrivals must wait to form a new batch, and the new batch must queue for the next available dispatcher. Open batch systems are the same as CB systems except subsequent call arrivals join the batch in dispatcher queue and are pipelined through before the channel is relinquished.
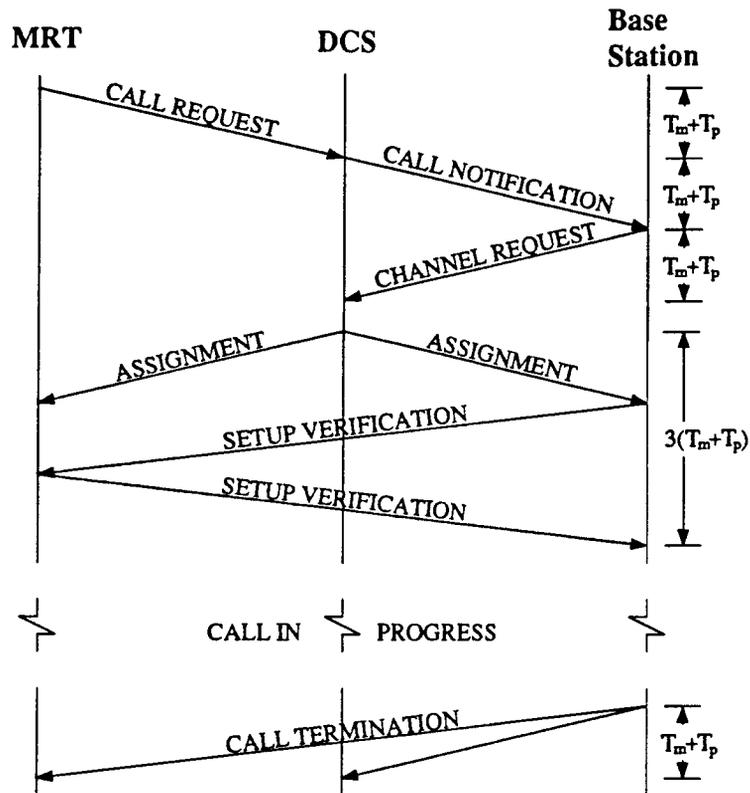
Figure 5 Single Call Processing Timing Diagram

When several batches from a single subnet are queued for dispatcher service, the order in which they are served may be made according to a number of different disciplines. Two batch ordering disciplines are investigated here: first come first served (FCFS), and oldest job first (OJF). The FCFS discipline serves batches in the order in which they complete formation, whereas the OJF discipline first serves the completed batch which contains the oldest individual call request.

18
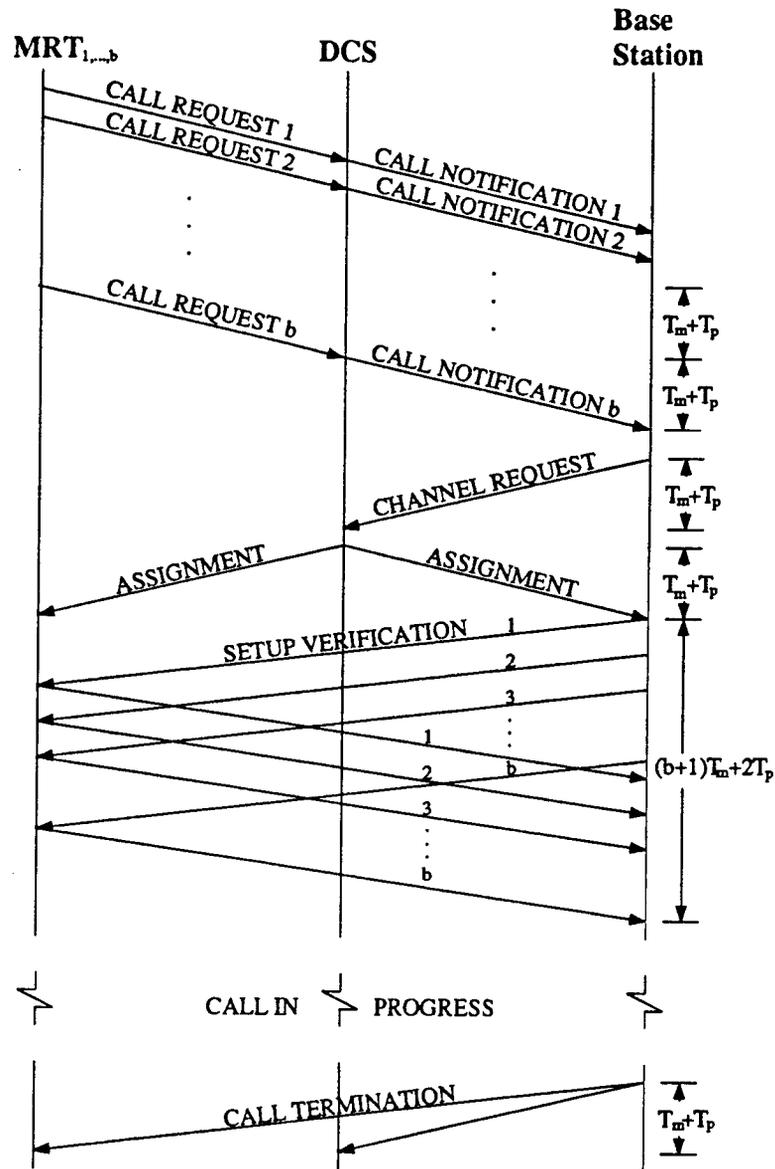
Figure 6 Closed Batch Call Processing Timing Diagram

2.2 MRS Subnet Operation

In operation, a single subnet consists of a number of MRTs distributed over the satellite coverage area. MRTs illuminated by different spot-beams (Figure 3) must communicate with the dispatch centre through their respective satellite transponders. When a mobile user initiates a call, the MRT sends a call request to the DCS at the NCC via the L-/Ku-band

signalling channel of the user's respective spot-beam, using the slotted Aloha or sloppy Aloha protocol [49]. Instead of allocating a channel in response to the call request, the DCS simply acknowledges receipt by returning a standby indication to the MRT, and relays the request to the dispatch centre via the base station to which the dispatch centre is attached [8]. Call requests are queued at the dispatch centre and displayed on dispatchers' consoles, sorted according to the originating spot-beam, and ordered by the time of arrival. When a sufficient number of call requests from the same spot-beam have accumulated in queue to form a batch, the completed batch is entered into a dispatcher request queue and ordered according to the active service discipline (either the FCFS or OJF). When a dispatcher comes available, the dispatch centre requests the DCS to allocate a channel under the appropriate spot-beam.

Channel requests could be generated automatically by the dispatch centre using a pre-selected batch size, or manually under dispatcher control, before being relayed to the DCS via the base station over the Ku-band signalling channel. Each channel request is queued at the DCS for the first available voice channel under the required spot-beam. Once a channel has been assigned, the DCS broadcasts the channel allocation to all the MRTs in the batch, as well as signalling the base station, which then sequentially verifies the presence of the MRTs on the assigned channel by pipelining a series of queries. Once channel verification is complete, the dispatcher can then select each MRT in sequence to talk with the mobile user. Once each MRT completes its call, it is free to initiate a new call. Upon clearing a batch of calls, the dispatcher proceeds with call termination, and via its associated base station, the dispatch centre reports to the DCS to relinquish the channel. The channel is then available to another user, and the dispatcher is ready to handle the next batch of calls. Once a dispatcher is again seized, it contacts the DCS to request another channel from the DAMA channel pool.

20

Figure 5 illustrates signalling timing for individually processed call requests, whereas Figure 6 illustrates the more general timing of the CB MRS DAMA signalling sequence. Timing for the OB system is more complex because the batch size is not fixed (it has a minimum size only), and additional calls may join the batch currently being served. For those call requests arriving in an OB system before channel assignment, batch treatment is similar to a CB system with the exception of the variable batch size. A call request arriving at an OB system after service has begun requires signalling confirmation on an individual basis; in this case, confirmation is done immediately prior to call servicing.

Since larger batch sizes result in greater savings in signalling overhead, dispatchers could be encouraged to use the largest batch size possible that meets the objectives of their subnet — perhaps by implementing a tariff structure. In Figure 6, it can be seen that the inband L-band signalling overhead for a CB system is given by

$$(b + 3)T_m + 4T_p \tag{2.1}$$

. Note that call requests are also made in L-band but use sideband signalling channels, and that call notifications, channel requests, and channel assignments made to dispatchers are all in Ku-band. The amount of bandwidth saved by reducing signalling overhead using a CB system is shown as a percentage of the bandwidth used for actual call servicing in Table 1, where it is assumed that $T_m = 50$ ms and $T_p = 250$ ms [9,27]. The amount saved, as a percent, is calculated from the difference between the amount of signalling overhead required by batches consisting of a single call and that required by batches of the given size. The savings in overhead are expressed as a percentage of the expected channel resources required to service the batch of calls. Given an expected call holding time of 20 seconds,

21

the percentage of overhead for a CB system is calculated as follows:

$$\frac{(b+3)T_m + 4T_p}{20b} \times 100\% \tag{2.2}$$

where the percentage of overhead is determined on a per call basis. Note that by using

the above values for T_m and T_p and taking the limit as $b \rightarrow \infty$ in (2.2), the percentage of

overhead is bounded below by 0.25%. Correspondingly, the maximum achievable increase

in usable bandwidth is bounded above by 5.75% — the difference between the maximum

percent overhead using single call processing and the minimum obtainable. These figures

represent significant resource savings. As can be seen in Table 1, the majority of possible

saving is achieved with a relatively small batch size.

Batch Size	Overhead (%)	Saving (%)	% of Total Possible Saving
1	6.00	0.00	0
2	3.13	2.67	46.4
3	2.17	3.83	66.6
5	1.40	4.60	80.0
8	0.95	5.05	87.8
12	0.73	5.27	91.7
20	0.54	5.46	95.0

Table 1 Savings in Signalling Overhead with a CB System

2.3 Description of MRS Delay Characteristics

In analyzing a dispatch subnet, the objective is to determine delay characteristics as a

measure of required user performance. The average call setup delay, T_{Setup}, is defined as

the mean time between call initiation at an MRT and commencement of conversation with

a dispatcher; it consists of the following components:

$$T_{Setup} = T_R + T_{WB} + T_{WD} + T_{WC} + T_V + T_S \qquad (2.3)$$

where

- T_R is the mean time for the call request to reach the dispatch centre,

- T_{WB} is the mean waiting time associated with batch formation,

- T_{WD} is the mean time waiting for a dispatcher to service the batch,

- T_{WC} is the mean time waiting to receive a channel assignment,

- T_V is the mean time to verify all MRTs in the batch are set up on the assigned channel,

- T_S is the mean time to complete service for leading calls in the same batch.

Figures 7 and 8 show the flow of a single call through the proposed CB and OB systems respectively, where nonzero instances of the components of T_{Setup} are indicated in bold square boxes.

As previously defined, let T_p be the satellite propagation delay between any pair of fixed or mobile earth stations, and let T_m be the serializing delay for a signalling message. Assuming signalling messages do not encounter collisions and are free of errors, then

$$T_R = 2(T_m + T_p) \qquad (2.4)$$

is the time required for round trip messages (a request and acknowledgment).

Similarly,

$$T_{WC} = T_{CR} + T_{QC} + T_{CA} \qquad (2.5)$$

where $T_{CR} = T_{CA} = T_m + T_p$. T_{CR} and T_{CA} are the times required for a channel request and channel assignment respectively, and T_{QC} is the additional time required for channel

23
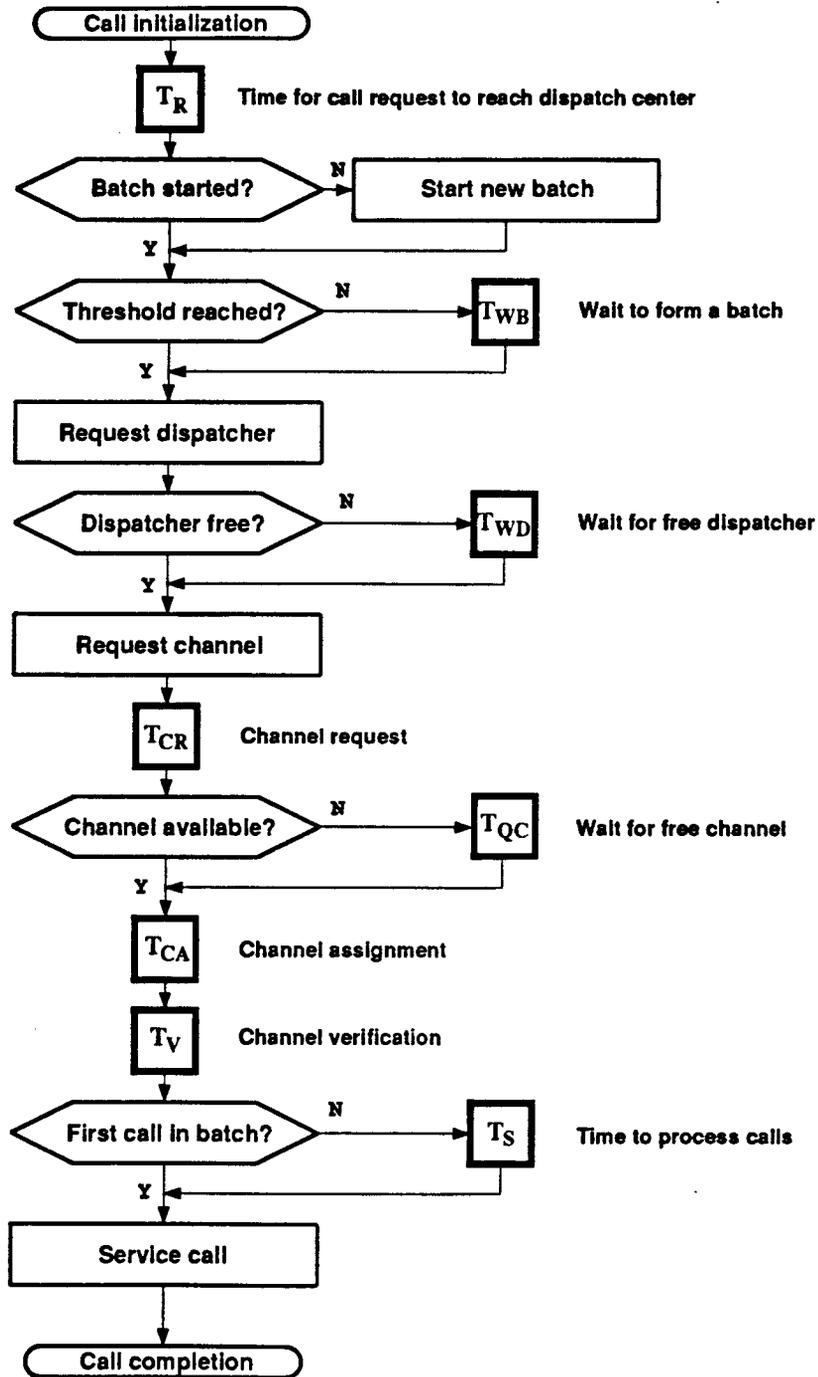
Figure 7 Closed Batch Flowchart

Call initialization

T_R — Time for call request to reach dispatch center

Batch started? — N → Start new batch

Y

Batch in service? — N → Threshold reached? — N → T_{WB} — Wait to form a batch

Y

T_S

Y → Request dispatcher

Dispatcher free? — N → T_{WD} — Wait for free dispatcher

Y

T_V

Request channel

T_{CR} — Channel request

Channel available? — N → T_{QC} — Wait for free channel

Y

T_{CA} — Channel assignment

T_V — Channel verification

First call in batch? — N → T_S — Time to process calls

Y

Service call

Call completion

Figure 8 Open Batch Flowchart

assignment queueing at the DCS should no channels be readily available for assignment. In the case of a CB discipline with pipelined channel verification signalling,

$$T_V = (b+1)T_m + 2T_p \qquad (2.6)$$

where b is the number of MRTs in a batch (see Figure 6).

For OB systems, T_V is considerably different. Let T_{V_A} be the mean verification delay for a call joining a batch before the batch service begins, and let T_{V_B} be the mean verification delay for a call joining a batch already in service. For calls that come in before batch service commences, T_{V_A} is given by equation (2.6) with the batch threshold size, b, replaced by the mean number of calls actually in the batch before service commences. For each call that arrives after batch service has begun,

$$T_{V_B} = 2(\bar{x}+1)(T_m + T_p) \qquad (2.7)$$

where $\bar{x}$ is the mean of a random variable, X, where for each new call request that arrives after a batch has started service X represents the number of calls that arrive after batch service starts but still have not commenced service at the time the new call arrives. The factor of $2(T_m+T_p)$ represents round trip verification signalling done for each new arrival on an individual basis, as discussed in Section 2.2. New call arrivals continue to be processed until an idle period occurs under the spot-beam being serviced, at which time the dispatcher surrenders the channel. For an OB system, T_V is a weighted average of T_{V_A} and T_{V_B}. However, since neither the expected number of calls in an open batch before service commencement nor the probability density function (p.d.f.) for the random variable X are known, a closed form solution for T_V for OB systems cannot be derived.

For the ith call in CB system, the time spent waiting for leading calls in the same batch to complete is the sum of the service times of those $i-1$ previous calls, which, on average

26

is simply $(i - 1) \cdot E[call\ holding\ time]$. Further averaging over the waiting times for all b calls in a batch yields

$$T_S = \frac{b-1}{2} \cdot E[call\ holding\ time] \tag{2.8}$$

. Again, however, for OB systems the lack of knowledge about the call arrival distribution prohibits the development of a closed form solution for T_S. If the call arrival distribution were known, then T_S could be calculated similarly to the open batch calculation with the *expected number of leading calls* substituted for b in equation (2.8), where the expectation accounts for the expected residual service time for a call in service when a new call arrives.

Finally, calculation of T_{WB}, T_{WD}, and T_{QC} are not straightforward since they are interdependent. T_{WB} could be calculated similarly to T_S with the substitution of *expected call interarrival time* for *expected call holding time* if the appropriate probability distributions could be determined. By using a finite population model, however, each time a call request enters the system the MRT that generated the call is no longer free to generate new calls, which results in a slowing of overall arrival rate. Similarly, each time a call is completed, an MRT that was engaged in a call becomes free to generate calls again, so the overall call rate increases. Therefore, a varying arrival rate due to the finite population model, means that the *expected call interarrival time* is dependent on queue length distributions for batch formation, dispatcher waiting, and channel assignment queues. Interdependencies are further complicated by the introduction of propagation and message delays, and channel resource competition between different private dispatch networks, each of which might have a different configuration. Even under the assumption of infinite population homogeneous subnets, multiple tiered queuing, batch processing, and long propagation delays result in a formidable queuing model not given to closed form analysis. Hence, in the next chapter the

27

delay characteristics of single subnet with various configurations are analyzed through the use of computer simulation.

3. Modelling and Analysis of a Single MRS Subnet

The analysis of a single dispatch subnet focuses on determining delay characteristics as the measure of user performance. Analysis concentrates on the expected performance under the assumption of stationarity where it is well known that channel utilization in stochastic systems must be strictly less than unity for stability [29,50]. In order to determine relevant delay characteristics, a network model is constructed.

3.1 Single MRS Subnet Delay Model

This chapter considers an instance of a single private subnet in order to gain insight into the general operation and performance of the proposed MRS DAMA protocol. Specifically, call processing delay of a single dispatch subnet, defined now as $T_C = T_{Setup} - T_{QC}$, is considered. The overall delay of an MRS network which includes T_{QC}, and is made up of many independent subnets operating under the proposed DAMA protocol competing for a fixed number of channels, is subsequently considered in Chapter 4. The analysis of call processing delay for a single subnet provides an understanding of the relationships between delay and offered dispatch traffic, batch size, and batch servicing discipline. For a more complete discussion of the advantages of call queuing on the sideband signalling channels, the reader is referred to [9].

The initial analysis is done under the assumption that there is a sufficient number of channels available under each spot-beam such that channels are always available upon dispatcher request. Thus, $T_{QC} = 0$ as channel assignment is immediate. As a result, the

29

T_{WC} component in T_C is equal to $2(T_m + T_p)$. Ignoring T_{QC} for the moment does not detract a great deal from the usefulness of the individual subnet analysis, since system behavior is such that T_{QC} remains relatively small when considering subnets of a practical size (see Chapter 4).

3.2 Single MRS Subnet Operation

The model under consideration has a number of both fixed and variable parameters. Fixed model parameters are related to the general MRS satellite network architecture and to call characteristics; they are summarized in Table 2. Variable parameters, shown in Table 3, are related to an individual dispatch subnet where a wide variety of configurations are required for comparative performance analysis.

Call rate per MRT (λ_R)	.001 calls/sec.
Mean call holding time $(1/\mu_R)$	20 sec.
Number of spot beams (s)	4
Control message serializing delay (T_m)	50 ms
Propagation delay (T_p)	250 ms

Table 2 Fixed MRS Model Parameters

A dispatch subnet is represented as a collection of n independent MRTs, each of which is assumed to generate calls with exponentially distributed interarrival times, at rate $\lambda_R = 0.001$ calls/s, and have a mean call holding time of $1/\mu_R = 20$s, where call holding times are also exponentially distributed [9,30,33,37]. Overall, the total offered traffic per MRT is 0.02 Erlangs in the absence of queuing, which is typical for LMR applications [51]. Call queuing temporarily stops queued MRTs from generating calls and thus slows the call arrival process so that a lower level of offered traffic per terminal is actually realized.

Number of dispatchers (d)	1 - 8
Batch size (b)	1 - 12
Number of MRTs (n)	$12d$ - $92d$

Table 3 Variable MRS Model Parameters

The MRT population is distribu...d uniformly under $s = 4$ spot-beams, and is served by d identical dispatchers. In keeping with the small number of terminals in each individual subnet, a finite population model must be used in MRS subnet modelling; similar network configurations employed in LMR applications are also modelled in this fashion [35]. A consequence of the finite population model is that the overall call rate varies over time as MRTs either waiting for service, or in service, cannot generate new calls (see Section 2.3). The assumptions of exponential interarrival and service times are accepted as good approximations derived from empirical observations of real LMR traffic [30,52].

The number of spot-beams is fixed, and is consistent with the proposed satellite coverage in Canada (Figure 3). Subnet configuration is varied by changing the number of dispatchers and the number of MRTs. The latter change also has the effect of varying the offered traffic. In subsequent analysis, both the subnet configuration and the batch size are varied, and the resultant effects on delay are observed for each service discipline. It is important to note that for a CB system, d may be greater than s as several batches might be served simultaneously under one spot-beam. On the other hand, for an OB system d must be less than or equal to s since at most one batch may be present under each spot-beam, and a dispatcher serving a particular spot-beam continues to serve all calls arriving at that beam until there is an idle period. If d were greater than s, then $d - s$ dispatchers would never be utilized in an OB system.

3.3 Delay Analysis

In order to determine an optimal strategy, the effects of dispatch network traffic, batch size, subnet size, number of dispatchers per subnet, and batch service discipline are examined. In some restricted cases of much simplified network models, some numerical or closed form solutions exist for delay analysis. However, due to the complexity of the proposed system, and the uncertainty in related probability distributions, no general closed form solutions exist. As an alternative, discrete event computer simulation is used as the primary tool of analysis. Nevertheless, simplified analytic models have been used to verify simulation model execution where possible (see Appendix A).

3.3.1 Comparison of FCFS and OJF Batch Service Ordering

Empirical results obtained via simulation indicate virtually no significant statistical difference in delay characteristics between systems employing the FCFS discipline and those using OJF. These results are consistent with the Markovian nature of the traffic and the call service distributions. Therefore, in subsequent analysis, OJF models are not considered, and FCFS models are selected for their efficiency in execution and ease of implementation. Thus, the only two models used for subsequent analysis are the FCFS CB and FCFS OB systems, but the results apply equally well to OJF systems. Henceforth, discussion will only refer to CB and OB systems but it is understood that either FCFS or OJF service ordering may equally well be in effect.

3.3.2 Closed Batch Subnet Configurations

Each variable system parameter may assume a range of values in order that many subnet configurations may be examined. Figures 9 through 12 show resultant delay characteristics in relation to the number of MRTs per dispatcher in a CB subnet with 1, 2, 4, and 8 dispatchers.

The respective curves in each figure show results from experiments with batch sizes of 1, 2, 3, 5, 8, and 12, and thereby illustrate the effects of different batch sizes on T_C.
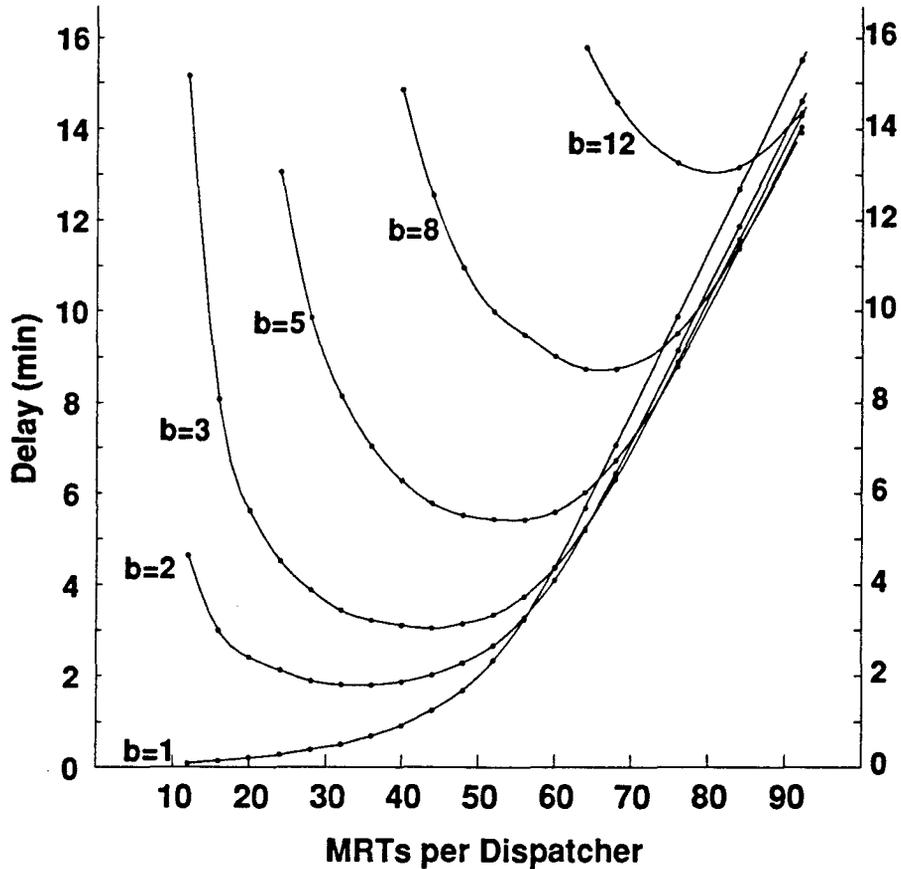
Figure 9 CB Subnet: Single Dispatcher Delay

Except for the case where $b = 1$, the general shape of the curves indicates that delay is long for small MRT populations with respect to the number of dispatchers, but delay decreases as the number of MRTs per dispatcher (offered traffic) increases. Eventually, a minimum delay is reached, and further increases in the MRT population cause delay to rise once again.
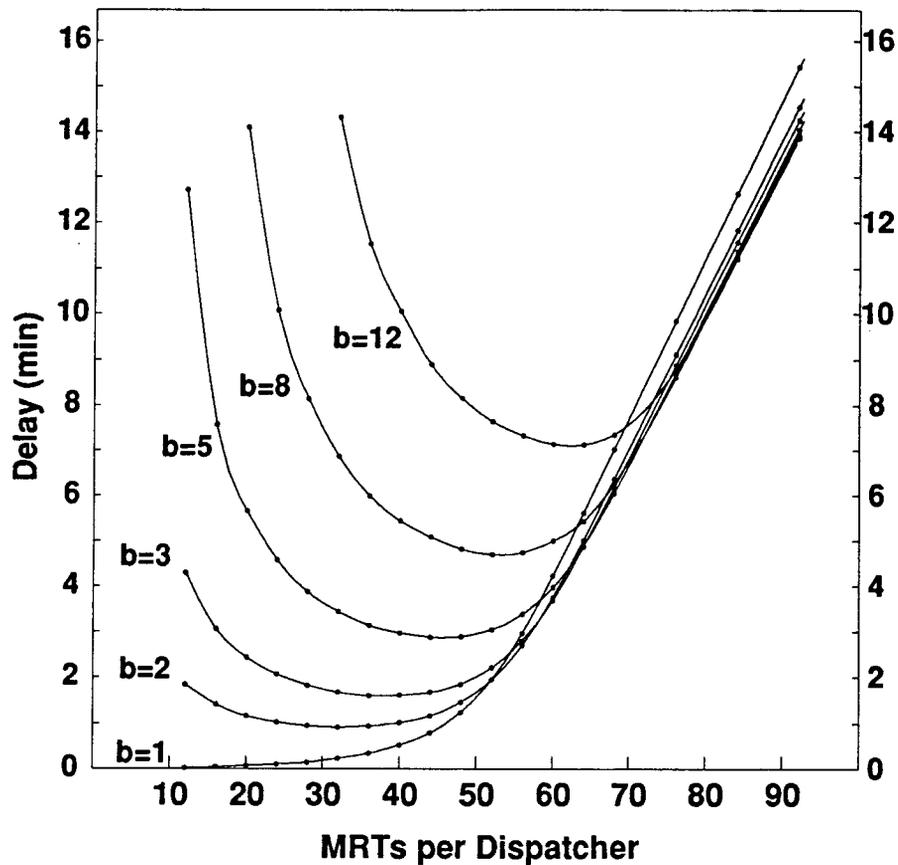
Figure 10 CB Subnet: Two Dispatcher Delay

For small populations, most of the delay is attributed to the time it takes to form a batch, T_{WB}. When $b = 1$, there is no batch formation delay. In the extreme (implausible) case, where there is a nonzero population of fewer MRTs (under each spot-beam) than required to form a single-batch, the resultant batch formation delay is infinite since no batch can be formed.

When the MRT population is increased to a number larger than the batch size, delay becomes finite. Further increases in population cause delay to decrease as batches are formed more quickly. However, if the population is increased too much, delay begins to rise once again as dispatchers become fully utilized, and T_{WD} becomes the dominant component of the overall delay. For very large populations, note that T_C increases linearly with respect

34
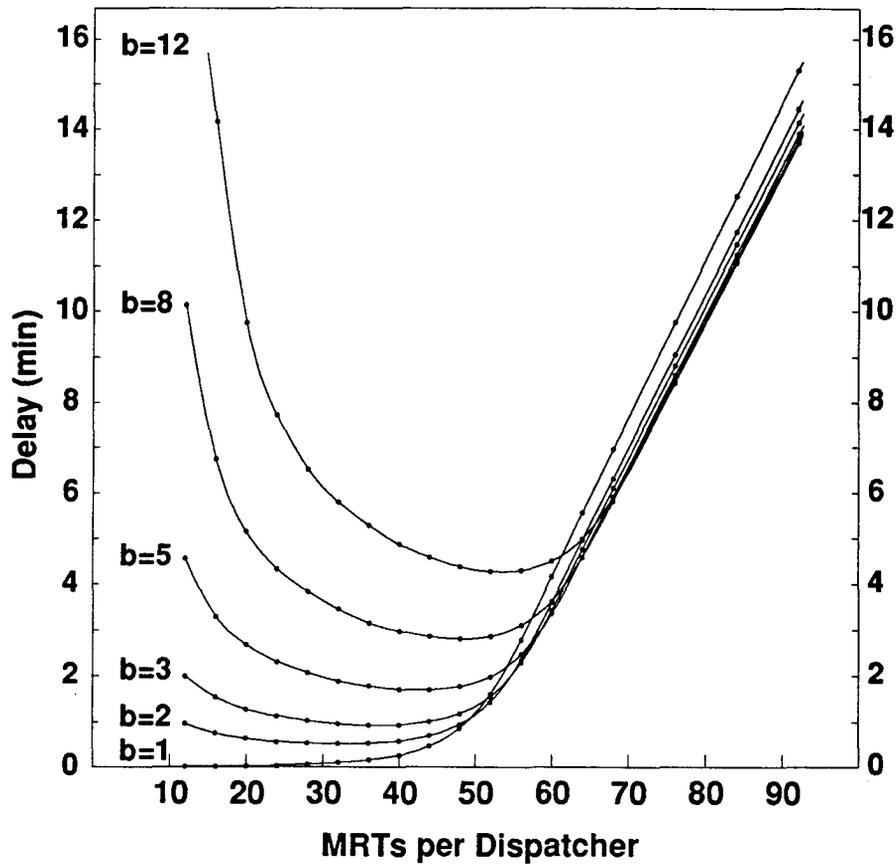
Figure 11 CB Subnet: Four Dispatcher Delay

to the MRT population due to the finite population model. Another consequence of the finite population model is that the delay remains bounded regardless of how large a subnet population becomes because call request simply accumulate in queue until equilibrium is reached. At equilibrium where the rate of calls generated by those MRTs not in queue nor engaged in calls is balanced by the rate of call completions.

Further note that the lines at the right side of each graph are slightly splayed due to overhead processing delays being proportionally larger for smaller batch sizes. The spacing between the lines narrows as the batch size is progressively increased due to less dramatic savings in signalling overhead for each successive increase in batch size (Table 1).
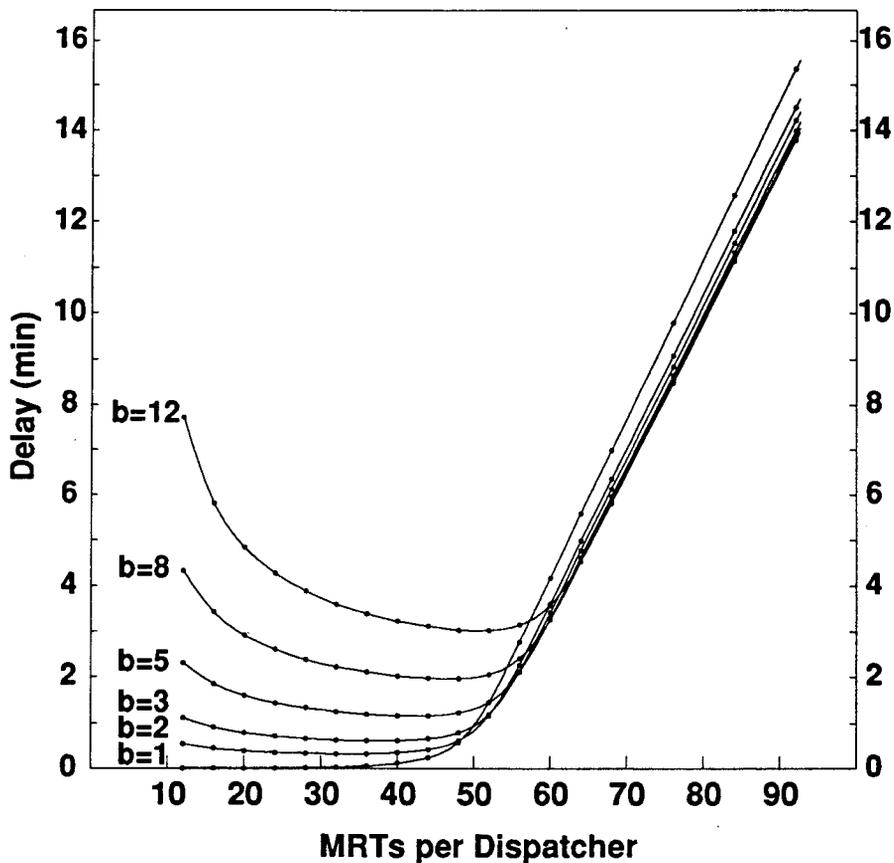
Figure 12 CB Subnet: Eight Dispatcher Delay

Closer examination of Figures 9 through 12 reveals that for any given batch size, the optimal number of MRTs per dispatcher corresponding to the minimum delay is fairly constant with respect to the number of dispatchers (see Table 4). Hence, it appears that the ratio of MRTs to the number of dispatchers is an important indicator of subnet delay performance. This is because a high number of MRTs per dispatcher results in long dispatcher delays (large T_{WD}). This finding is, in fact, consistent with other research investigating similarly configured LMR dispatch networks [35,53].

Batch	Number of Dispatchers			
Size	1	2	4	8
1	12	12	12	12
2	34	32	32	32
3	44	36	36	36
5	56	44	44	40
8	66	52	48	48
12	80	62	52	52

Table 4 Location of Minimum Delay in MRTs per Dispatcher for CB Systems

While the delay minima are of a similar range with respect to MRTs per dispatcher, the absolute delay is shorter for larger subnets, due to the much accelerated batch formation process, as shown in Figure 13 The largest difference in delay occurs between single and double dispatcher systems, then progressively smaller differences are observed with each additional dispatcher. For any constant ratio of MRTs per dispatcher and fixed batch size, as the number of dispatchers is increased (yielding a proportional rise in absolute population) the rate at which the delay decreases begins to diminish. Initially, additional dispatchers result in great improvements in overall delay, but after four dispatchers, improvements come more slowly. It appears that little is gained by having more than eight dispatchers, and as previously noted, LMSS are likely to serve a collection of smaller private dispatch subnets so large subnets are not consistent with the system being modelled.
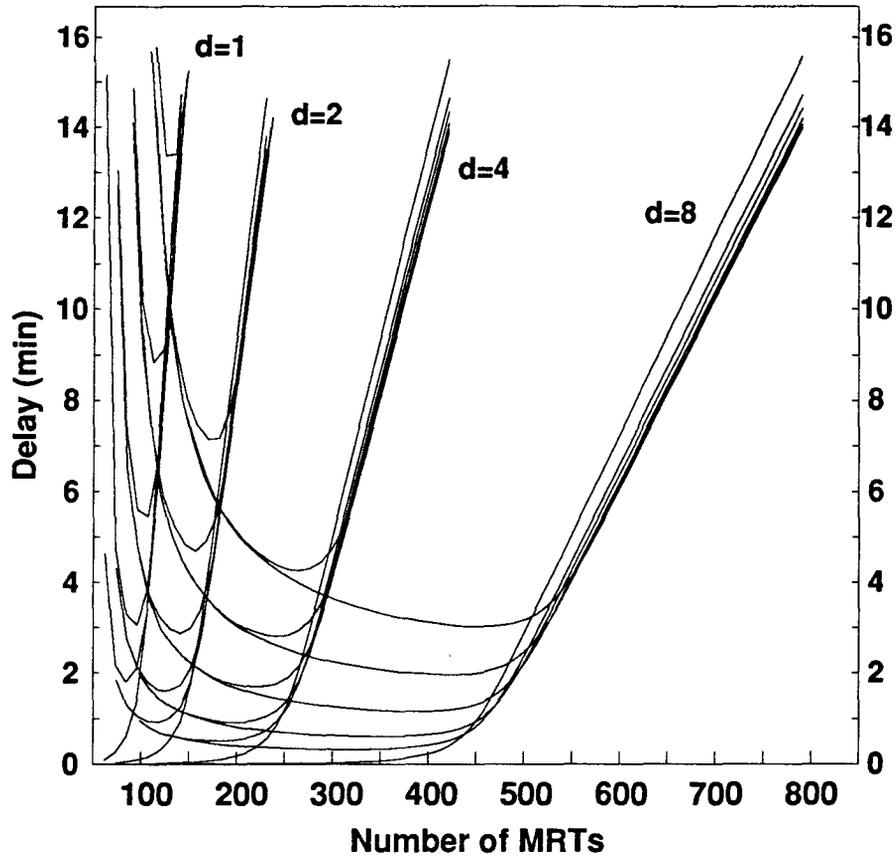
Figure 13 CB Subnet Delay Curve Comparison for 1, 2, 4, and 8 Dispatchers

3.3.3 Open Batch Subnet Configurations

Delay curves for OB subnets with $d = 1, 2, 3, 4$ are shown in Figures 14 through 17 respectively. Subnets with more than four dispatchers are not considered since at most four can be utilized at any moment. If more than four dispatchers were assigned to a single subnet with a fixed MRT population, the delay curves would be identical to Figure 17, regardless of how many more dispatchers were added.

The general rationale for the shape of the OB curves is essentially the same as for CB systems; however, there are some behavioral differences. First of all, as subnet size gets large (right hand side of figures), the delay seems to grow almost identically in Figures 14, 15, and

38
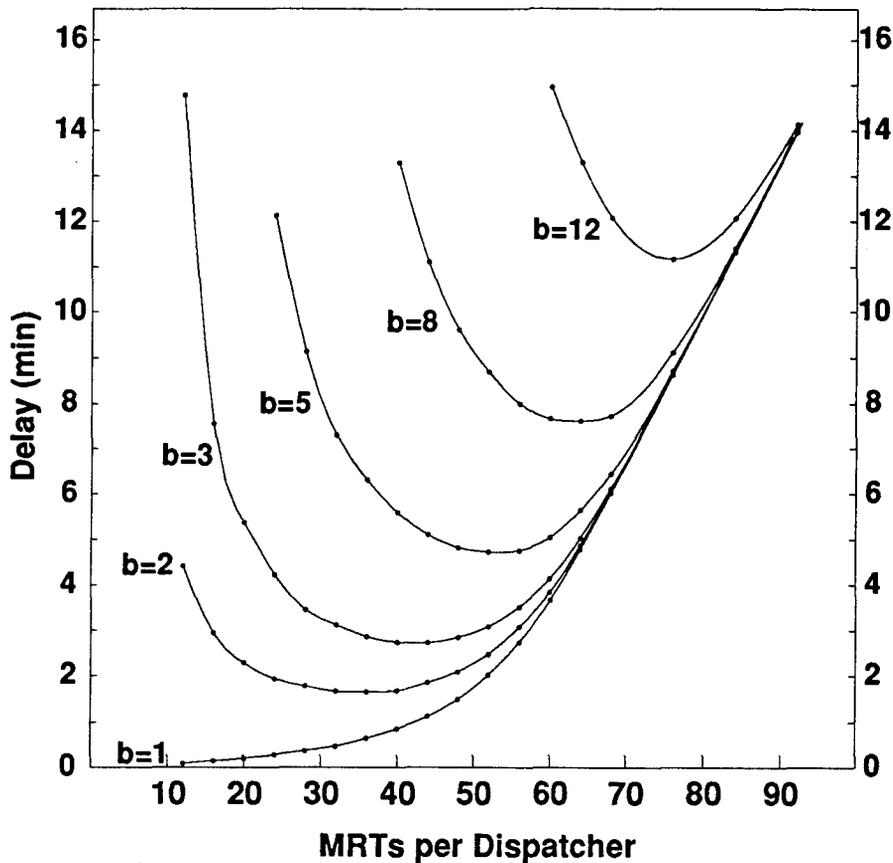
Figure 14 OB Subnet: Single Dispatcher Delay

17, regardless of batch size. This is in contrast to CB systems whose linear tails were splayed due to differences in the average overhead per call for different batch sizes. Since overloaded OB systems tend to hold channels open for long periods of time, the vast majority of calls that arrive do so while a batch is already in service; therefore, the number of unserviced calls remaining in an open batch when a new call arrives tends to be the same regardless of the number of calls required to form the batch initially. As a result, the overhead associated with most OB calls is also the same, regardless of size designated for batch formation. If a batch stays open for a very long time, the signalling overhead saved due to the initial batch size becomes much less relevant in the long run. However, once again it is recognized
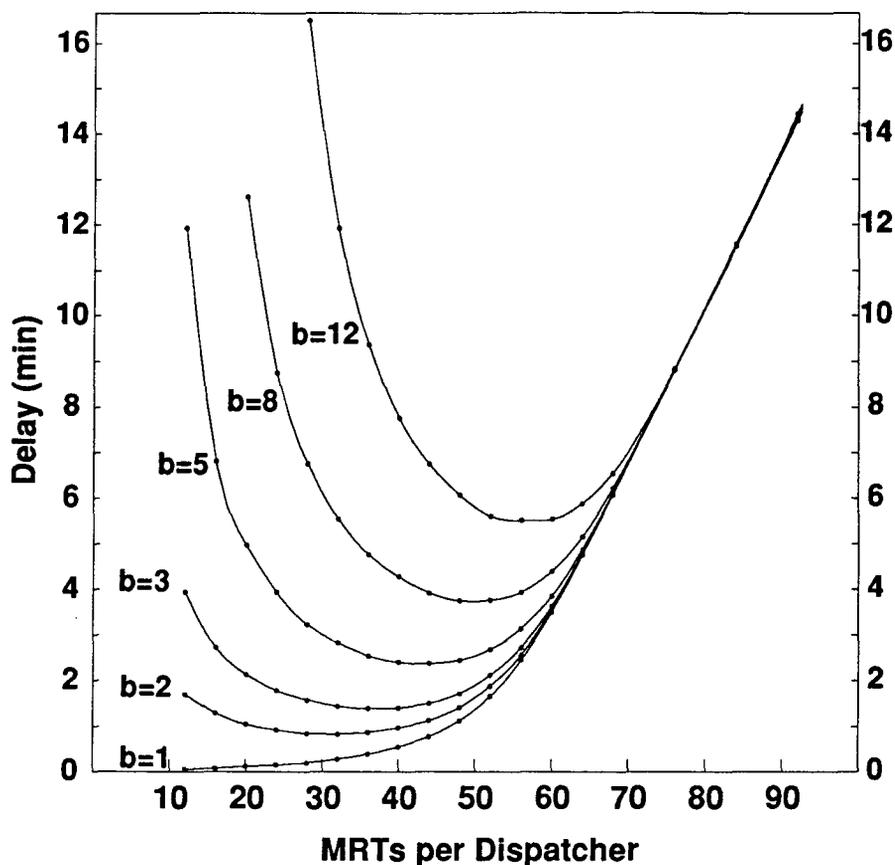
Figure 15 OB Subnet: Two Dispatcher Delay

that subnets with over-utilized dispatchers are not realistic, and degenerate into subnets with permanently assigned channels, which is simply frequency division multiplexing (FDM).

In Figure 16, while the delay curves still have the same general shape as the preceding figures, a slightly different phenomenon is observed at very high utilization: the right ends of the delay curves in Figure 16 turn upward in much more erratic fashions than those in previous figures. This rapid rise in delay is the result of spot-beam starvation. Spot-beam starvation occurs when there is a sufficient amount of traffic generated under a proper subset of the spot-beams to keep all subnet dispatchers continually busy, while call requests generated under the other spot-beams go without service.
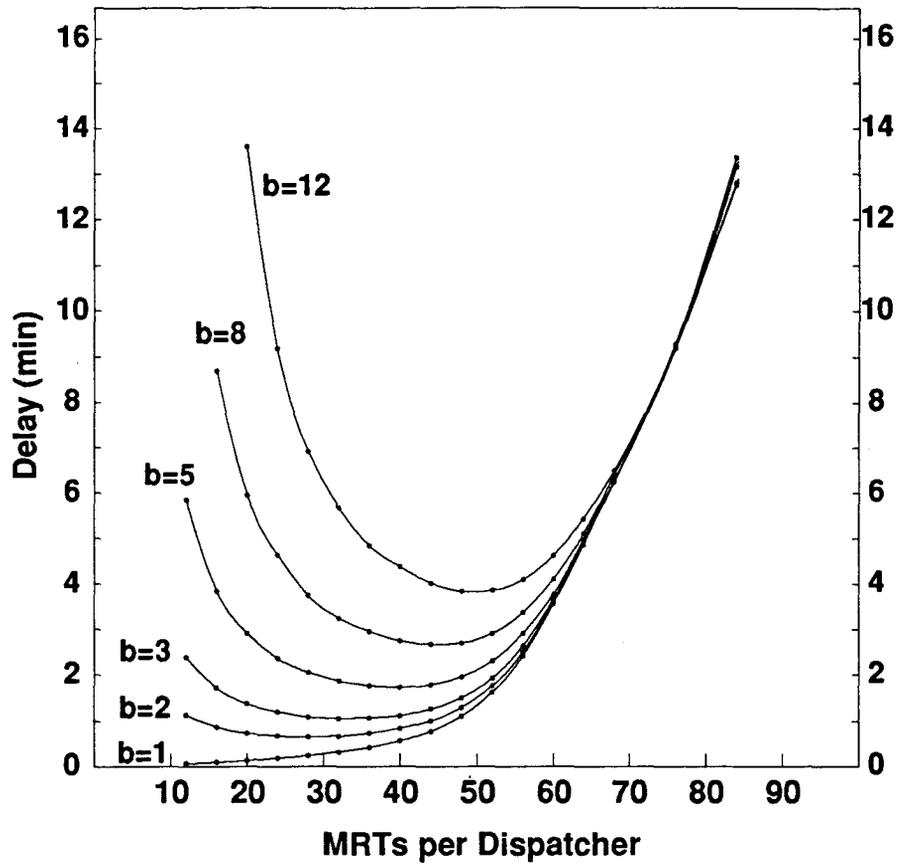
Figure 16 OB Subnet: Three Dispatcher Delay

Spot-beam starvation is not observed in either Figure 14 or Figure 15 because the range of values on the horizontal axis does not extend to a sufficiently large population. Correspondingly, starvation occurs at a much higher MRT per dispatcher ratio in a single dispatcher subnet than it does in a two dispatcher subnet, and the ratio is higher for a two dispatcher subnet than for one with three dispatchers. The reason that starvation only occurs at much higher MRT per dispatcher ratios when there is one or two dispatchers is that the absolute number of terminals per spot-beam is the critical factor. For example, suppose the are two subnets, each with 80 MRTs per dispatcher, but one has a single dispatcher and the other has three dispatchers. The subnet with one dispatcher has only 20 MRTs
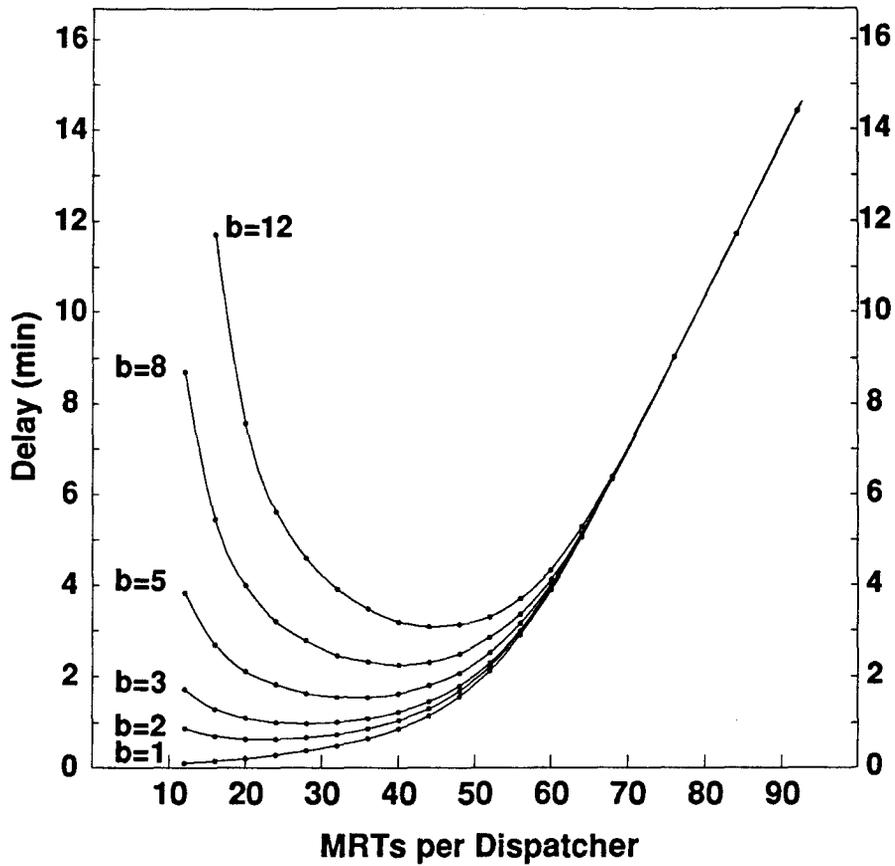
Figure 17 OB Subnet: Four Dispatcher Delay

under spot-beam which do not generate enough traffic to continually tie up the dispatcher, so no starvation occurs. The subnet with three dispatchers has 60 MRTs under each spot-beam which generate enough traffic to tie up a dispatcher for extended periods of time, so starvation may occur. On the other hand, starvation is not observed in Figure 17 where there are four dispatchers because there is exactly one dispatcher available for each spot-beam, so it is not possible for any to go without service. Nevertheless, spot-beam starvation does not cause a problem for subnets of any practical size, regardless of the number of dispatchers, because the population level at which it occurs is where dispatchers are hopelessly over-utilized and far above the population that minimizes the call setup delay.

Similar to those cases of CB subnets, it is observed that on an MRT per dispatcher basis the location of the minimum delay on OB delay curves is fairly consistent with regard batch size, though not quite as consistent as for CB systems (see Table 5). In addition, the loci of minima for OB systems tend to be where there is a smaller number of MRTs per dispatcher. Figure 18 combines Figures 14 through 17 on the same scale for ease of comparison, and once again it is seen that the minimum delay for a given batch size is always found with subnet with a greater number of dispatchers, although for OB systems d is constrained to be less than or equal to s.
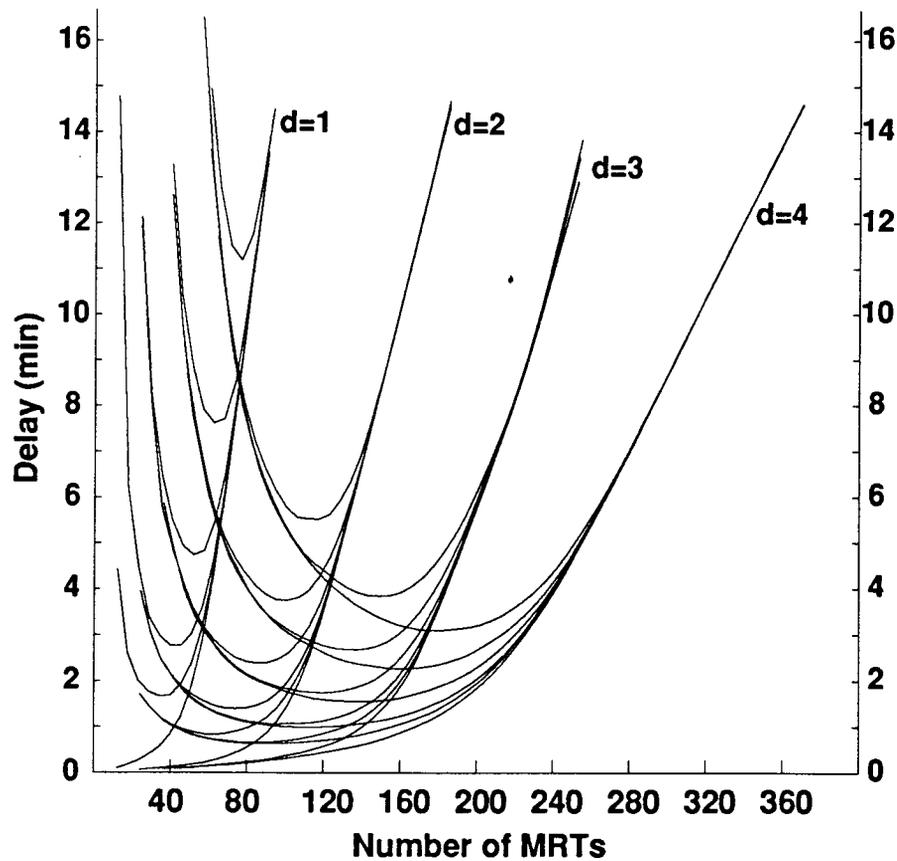
Figure 18 OB Subnet Delay Curve Comparison for 1, 2, 3, and 4 Dispatchers

Batch	Number of Dispatchers			
Size	1	2	3	4
1	12	12	12	12
2	36	32	28	24
3	42	36	32	28
5	52	44	40	36
8	62	48	44	40
12	74	56	48	44

Table 5 Location of Minimum Delay in MRTs per Dispatcher for OB Systems

3.3.4 Open and Closed Batch System Comparison

In order to facilitate comparison between CB and OB systems, Figures 19 through 21 are provided. Figures 19 through 21 each combine two previous figures: respectively they combine Figures 9 and 14, 10 and 15, and 11 and 17.

For small batch sizes, the minimum delays are roughly equal for both CB and OB systems. It appears, however, that CB systems offer larger subnet sizes at optimal delay, although OB systems always yield lower minimum delay for larger batch sizes. For a single dispatcher subnet (Figure 19), OB systems always perform better than CB systems in terms of delay characteristics, with increasing advantage for larger batch size with respect to the absolute difference in delay. This is primarily due to the ability of OB systems to occasionally service new incoming calls without waiting for a new batch to form, so a dispatcher has opportunity to flush all calls through under the spot-beam it is currently serving before relinquishing the channel. Closed batch call arrivals, on the other hand, must always wait for batch formation, and this delay is most often long when the total MRT population is small or batch size is large. A two dispatcher subnet (Figure 20) behaves very similarly to a single dispatcher

subnet, except minimum delays achieved by the larger population two dispatcher subnets
are significantly less.
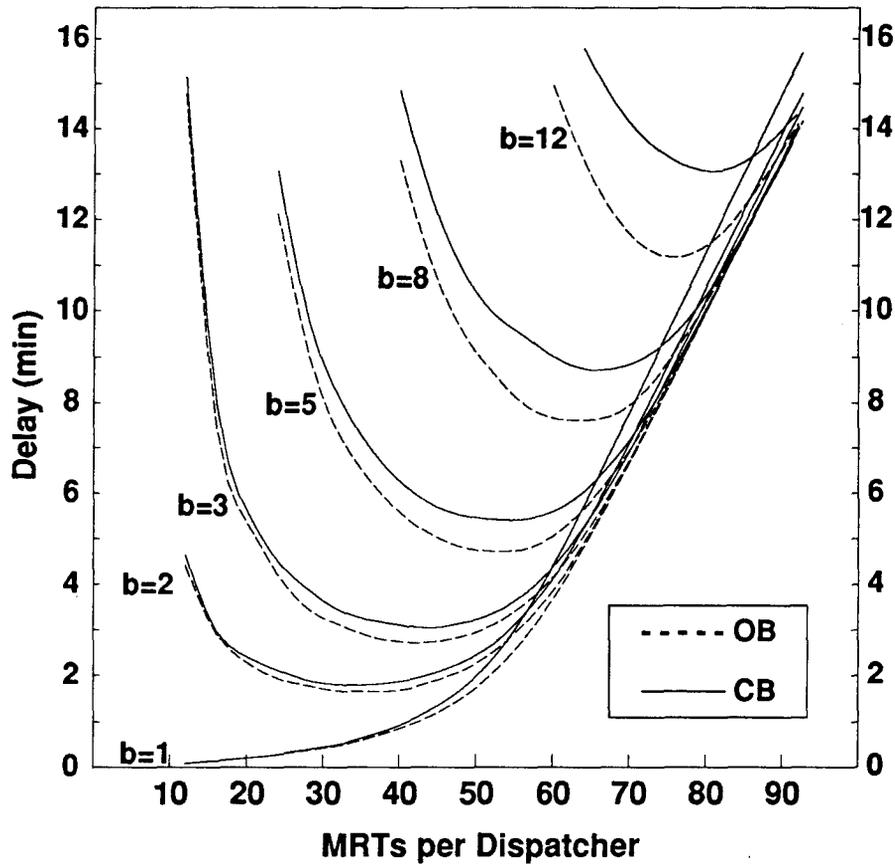
Figure 19 CB and OB Single Dispatcher Delay

In the case of four dispatcher subnets (Figure 21), the delay curves are somewhat
different. For small populations, the OB systems still perform better, but as the population
increases, CB systems outperform their OB counterparts when small batch sizes are employed.
For larger batch sizes, OB systems begin to show better performance once again.

The reason for good OB performance at low traffic levels is the same as that given for
one and two dispatcher subnets. As the population is increased, however, small batch CB
systems have the advantage of being able to use more than one dispatcher to service a single
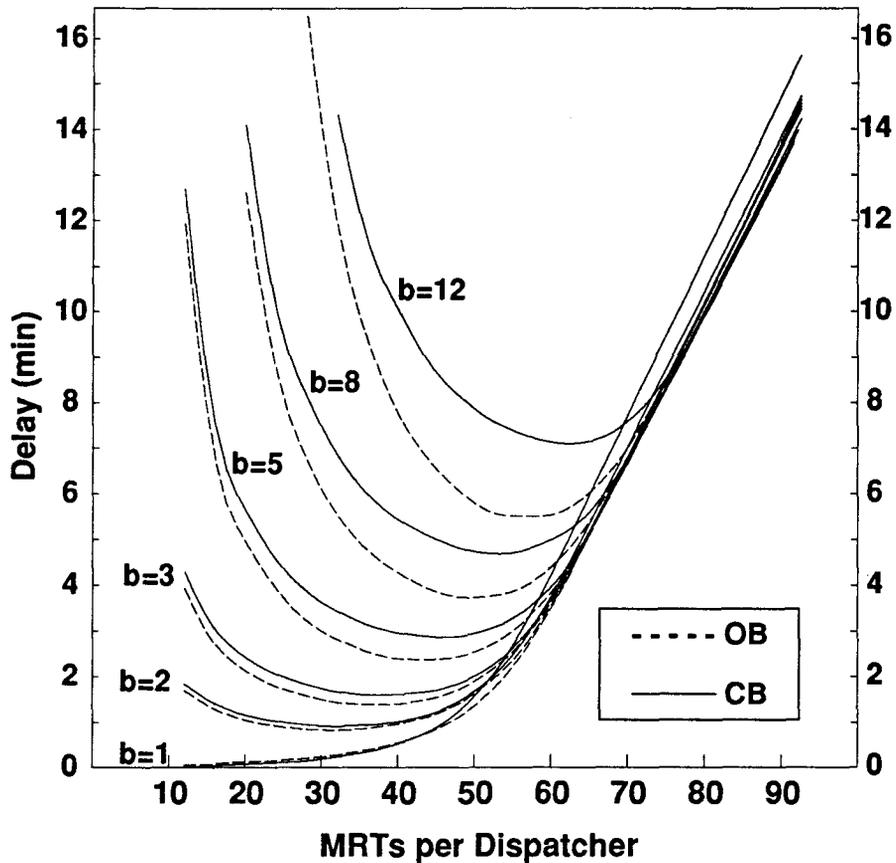
Figure 20 CB and OB Two Dispatcher Delay

spot-beam if high traffic fluctuations so warrant. The ability to apply any combination of dispatchers to any spot-beam is equivalent to statistical multiplexing for closed batches, since "batch arrivals" may be served by the first available dispatcher. The advantage of statistical multiplexing is well documented [29], and plays an important role here in reducing the overall delay for CB systems.

An important observation of Figure 21 is that the minimum delay for the CB systems is often the same or less than for OB systems when small batch sizes are used. Moreover, the loci of the CB minima tend to occur at larger population levels than their OB counterparts. As batch size increases, OB minima become smaller than those of CB systems. Therefore,
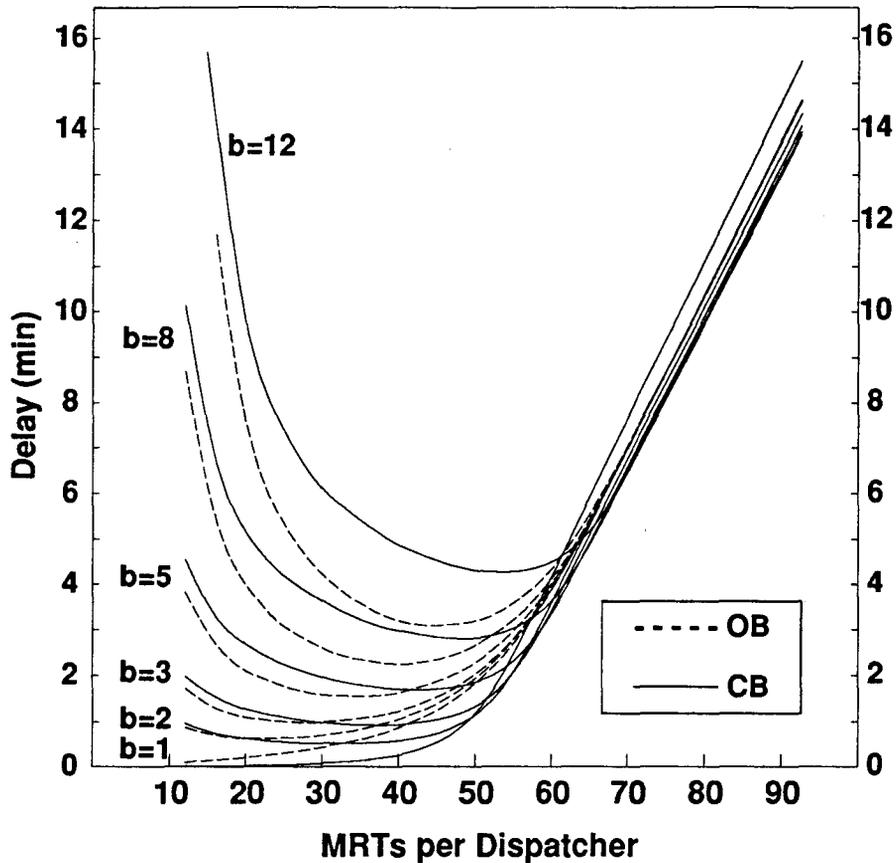
Figure 21 CB and OB Four Dispatcher Delay

it appears that either a CB or an OB system may be selected for optimal performance in a four dispatcher subnet, depending on its configuration and batch size.

With larger batch sizes, batch formation delay increases dramatically. For CB systems, T_{WB} becomes more significant than the saving in T_{WD} due to statistical multiplexing. Thus, as batch size increases, OB systems tend to have shorter overall delay than CB systems. Open batch systems gain advantage by being able to process *all* calls queued under a particular spot-beam before relinquishing a channel. Therefore, the delay of those calls which arrive during batch service is shortened greatly, since they do not have to wait for a new batch to form. With larger batch sizes this effect becomes more important; not only is batch formation

delay longer for larger batches, but a greater number of additional calls are likely to arrive during batch servicing of a larger batch. As more calls arrive during batch servicing, more calls per batch avoid the long delay associated with large batch formation.

For very high population levels, where dispatchers are fully utilized, T_{WD} is the dominant delay component. The amount of overhead associated with an OB system at full dispatcher utilization is slightly smaller than the overhead associated with a CB system with a batch size of two, as is reflected in the relative positions of the OB and CB linear tails at the right hand side of Figure 21. The overhead associated with each call in an OB system with fully utilized dispatchers who never realize breaks in servicing once batches are open is

$$2(T_m + T_p) \tag{3.1}$$

For CB systems, the overhead per call is given by dividing Equation (2.1) by the batch size, b:

$$\frac{(b+3)T_m + 4T_p}{b} \tag{3.2}$$

By equating the two formulae given in Equations (3.1) and (3.2), substituting the values given in Table 2 for T_m and T_p, and solving for b, the following result is obtained: $b \simeq 2.1$. This result corresponds to the observation that the linear tails of the OB systems in Figure 21 converge just below the tail of the CB system using a batch size of two. For larger batch sizes, the smaller overhead associated with CB systems result in lower delays relative to OB systems at full dispatcher utilization.

3.4 Summary

By dealing with the simplified case of a single subnet which does not face channel competition from other subnets, several insights into the behavior of the new MRS protocols

have been obtained. Insights have been gained by experimenting with a wide range of combinations of MRT population size, the number of dispatchers, batch formation strategy, batch service ordering discipline, and batch size.

It is observed that the number of MRTs per dispatcher (dispatcher load) is often more relevant in considering delay performance than the absolute MRT population itself, because competition for dispatcher service within a subnet can result in large delay. However, very low or very high MRT to dispatcher ratios constitute extreme, implausible network configurations and are therefore not considered realistic candidates for network configuration. When there are only a few MRTs for each dispatcher, dispatchers are under-utilized and batch formation delay is often large. Very high MRT per dispatcher ratios, on the other hand, result in over-utilized dispatchers where long queues for dispatcher service result and delay is also long. It is important to note, however, that small absolute populations (Figures 19 and 20) tend to show large delay characteristics even for a reasonable MRT per dispatcher ratio because calls are still generated too slowly; thus, dispatcher utilization is often very low even when delay is high.

Whether batches are ordered for service by the FCFS or the OJF discipline has little effect on subnet delay characteristics. However, considerable differences are observed between CB and OB systems. When subnets are small (one or two dispatchers), OB systems always exhibit superior delay characteristics over their CB counterparts, regardless of batch size or number of MRTs per dispatcher. As the number of dispatchers increases, the minimum delay for CB systems is often smaller than that of similar OB systems, depending on batch size and offered traffic.

Open batch systems always perform much better than CB systems when traffic is low.

When the batch size is large, OB systems again have an advantage by being able to clear all calls under a given spot-beam, which tends to reduce batch formation delay. However, for large subnets with high traffic, CB systems often perform better due to the advantage of being able to statistically multiplex batch arrivals. In addition, the nature of CB systems allows more dispatchers than spot-beams to be employed, although very large networks are unlikely.

Increasing batch size also results in greater savings in signalling overhead per call, although the most dramatic increases in savings occur at relatively small batch sizes (see Table 1. Moreover, the utilization of small batch sizes generally contributes to more favorable delay characteristics, particularly at low traffic levels. A unit batch size performs best at low traffic levels. Although single sized batches do not offer any signalling overhead saving, at very low traffic levels resource conservation does not present a problem. Alternatively, batch formation time-outs might be used for larger batch sizes when traffic is low and channel resource competition is minimal. A batch formation time-out would allow partially formed batches to request a dispatcher some specified time after the first call request in the batch has been received.

In consideration of all of the above factors, the results enable the estimation of the optimal batch size and service discipline to achieve a desired performance objective for a given network size and number of dispatchers. Overall, an understanding of the effects of the new MRS protocol applied to a single network has been achieved. In addition, the delay analysis of a single subnet affords a good point of departure in further investigating the behavior of the protocol when applied to multiple subnets in a network where competition exist for a finite number of channels.

4. Integrated MRS and MTS Network

Having analyzed the delay characteristics of a single subnet, the analysis of delay characteristics associated with an MRS network consisting of a collection of similar subnets in channel resource competition remains to be done. Still further, an investigation of the effects of integrating MRS and MTS traffic into a single network must also be performed. The behavior of an MRS network in isolation is obtained in the course of analyzing an integrated network, where the behavior of separate MRS and MTS networks with fixed channel pool allocations form the basis of comparison for the proposed integrated strategy.

If given a finite number of channels to distribute between two traffic sources, the simplest allocation method would be to partition the channels into two fixed size channel pools — one for each service. However, if the two sources generate the same type of traffic then, as is well documented, overall performance for both networks improves by combining them into a single network with equal access to all channels [50]. Improved performance is due to a statistical smoothing which occurs as high and low fluctuations in one traffic source tends to cancel fluctuations in the other. However, as previously discussed, when networks with heterogeneous traffic sources are combined, one might improve at the expense of the other unless some measure of control is introduced into the system.

In particular, it has been shown that Erlang-B traffic may be sacrificed to improved Erlang-C performance [44]. While LMSS MRS traffic is not modelled as Erlang-C, the MRS system does employ call queuing, and simulation results indicate that, once again, blocked-calls-cleared telephone traffic performance deteriorates rapidly in favour of queued dispatch

radio service, as queued calls seize channels as soon as they come available. The new strategy proposed here, however, balances the two services and allows more total throughput without a degradation of the performance of either system. In addition, the new reserved margin system can be applied to other blocked-calls-dropped and blocked-calls-queued services equally well as the proposed LMSS voice network.

4.1 Dynamic Channel Allocation

4.1.1 The Reserved Margin Strategy

In examining the problem of integrating MRS and MTS traffic, the behavior of the telephone traffic is first examined, where an Erlang-B traffic model is assumed. The blocking probability of the Erlang-B model, p_B, is related to the offered traffic, ρ, and the number of available channels, n, by the following formula:

$$p_B = \frac{\rho/n!}{\sum_{k=0}^{n} \rho/k!} \tag{4.1}$$

. Suppose there are 30 channels dedicated to telephone traffic and a blocking probability of 0.01 is desired. Given these parameters, the offered traffic utilizes $\rho = 20.3$ channels on average. Conversely, an average of 9.7 channels — almost one third of the available resources — are left unused. Since calls are only dropped when all 30 channels are in service, there are periods where the number of free channels is considerable larger than the mean. It is apparent that a large margin of free channels is maintained only in order to ensure that all channels are not busy more than 1% of the time. In theory, an average of an additional 9.7 channels of capacity could be made available to dispatch radio without degrading telephone service, provided preemptive priority could be given to telephone service

for up to the full 30 channels. However, preemptive priority is entirely infeasible, so a more moderate system for sharing is in order.

The basic premise of the new channel allocation scheme is to reduce the mean number of free channels required when combined with radio service, and thereby provide additional channel capacity for call servicing. In keeping with the overall objective, however, the utilization of the available free channels must be such that overall service improves. In an integrated network, the idea is to tentatively maintain a small margin of free channels at all times for telephone traffic, where the mean number of free channels is considerably less than that given by the Erlang-B model. If there are more free channels available than stipulated, dispatch traffic may use them to service calls in queue. Blocking of telephone traffic only occurs when telephone calls or batch services do not complete quickly enough to maintain the desired margin, but new telephone traffic is generated quickly enough to consume those channels in reserve.

Combining both MRS and MTS by sharing a common channel pool, rather than allocating separate channel pools, allows channels unused by one service due to a traffic lull to be used by the other service if the latter is simultaneously experiencing a high statistical fluctuation in its traffic level. The ability of the system to statistically multiplex may be more clearly illustrated by example. Suppose there are two separate 30 Erlang-B networks, each requiring a blocking probability of 0.01. Given these constraints, a maximum of 20.3 Erlangs of traffic can be handle by each network, for a total of 40.6 Erlangs of traffic. On the other hand, a single network with 40.6 Erlangs of traffic requires only 53 channels to maintain the same blocking probability of 0.01. Therefore, the combined network provides the same level of service with 7 fewer channels, which may be allocated elsewhere. Though more difficult

to analyze, an analogous situation occurs when an Erlang-B and Erlang-C networks are combined. Therefore, a smaller free channel margin is required by the combined network than two networks in isolation.

With regard for telephone service, it should be noted that the margin of free channels is independent of current traffic load (the number of channels currently dedicated to MTS call servicing) because of the memoryless property of exponentially distributed interarrival times under the Erlang-B traffic model. In other words, the expected rate of telephone traffic is constant regardless of how much is currently in service. Hence, it is reasonable to expect that the free channel margin should be held fairly constant, regardless of current load. Differences may occur, however, when the balance between dispatch and telephone traffic changes the channel relinquish process (discussed below). Thus, the problem is reduced to finding the optimal value for the tentative free channel margin when given particular levels of telephone and radio traffic and their respective performance criteria in terms of blocking probability and delay.

Since an analytical closed form solution for the system is not known, the optimal free channel margin must be determined experimentally through discrete event computer simulation.

4.1.2 Comparison of Integrated Service Strategies

An application of the reserve margin is now shown in comparison with two other methods for integrating Erlang-B and Erlang-C traffic: separate channel pool divisions and Peritsky's method [46]. Some of the numerical values given in Peritsky's paper are contradictory, so they cannot be verified. Nevertheless, by reproducing his method as it is described, rather than accepting the results as presented, it may be verified that the system does in fact work.

However, the system proposed above performs better than Peritsky's method, even when applied to his own traffic model. In addition, the new system is much simpler to implement.

Peritsky analyses a model with 40 channels available to share between both services. The traffic model he uses for his analysis assumes a mean telephone call holding time of 3 minutes, 1/30 Erlangs of traffic generated per MTT, and a resultant telephone blocking probability of 5 percent. For dispatch traffic, a mean call holding time of 15 seconds is assumed, where the mean delay is less than one holding time, and traffic generated by each MRT is 1/120 Erlangs. Given these traffic characteristics, a pure Erlang–B network with 32 channels accommodates 802 MTTs, and a pure Erlang-C network with 8 channels also accommodates 802 MRTs. The numbers of terminals that can be accommodated by separate channel pool divisions are compared to the numbers of terminals that can be accommodated by shared channel strategies at the same performance levels in Table 6, the effectiveness of the new method is clearly shown.

Channel Allocation Method	Number of MTTs (Erlang-B traffic)	Number of MRTs (Erlang-C traffic)
Separate Channel Pools	802	802
Peritsky's Method	819	819
Proposed Method	832	832

Table 6 Comparative Results of Various Channel Allocation Strategies

4.2 Integrated Network Operation

When a dispatcher requests a channel to service a batch, the DCS allocates a channel only if the number of free channels is greater than the predetermined telephone reservation margin, otherwise the channel request is placed in queue until a sufficient number of channels become

available such that the margin can be maintained, at which time queued channel requests are served in order of arrival. A requests for telephone service, on the other hand, is served immediately if any channel is available.

Channels are reallocated at the time of channel release. As each MRS batch or MTS call completes service a channel is relinquished and returned to the DAMA channel pool for subsequent reallocation. If the number of free channels is less than the specified margin, a released channel is used to reinstate the margin, otherwise it is used to serve any queued radio calls (see Figure 22).
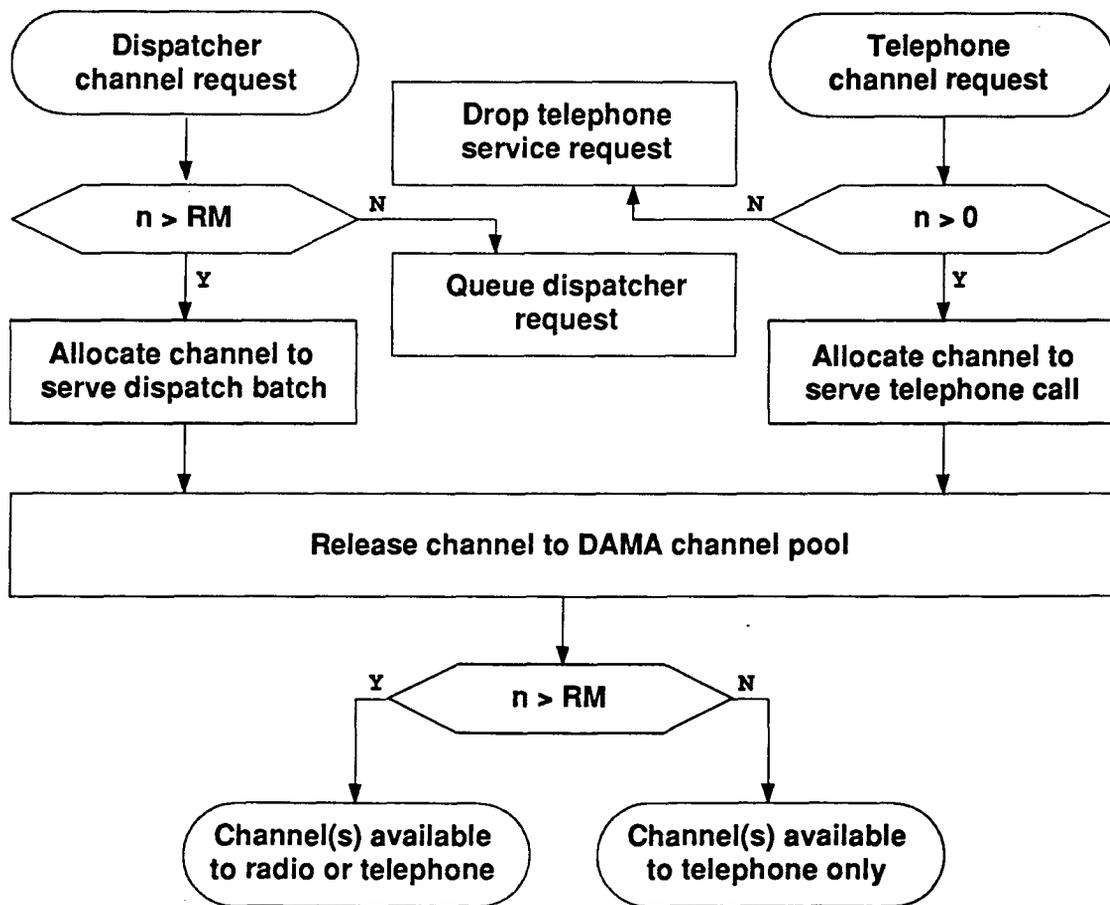
Figure 22 Reserved Margin Channel Allocation

4.3 Integrated Network Model

The model used for analyzing the reserved margin channel allocation scheme in an MTS and MRS integrated LMSS network consists of an Erlang-B traffic source for telephone and a collection of subnets operating under the new MRS protocol detailed above. The definition of the MTS traffic model is straightforward; however, the determination of an appropriate model for a collection of MRS subnets operating in a combined network with a finite number of channels is much more complex. In order to achieve the optimal performance for an MRS network, it is necessary to determine the best subnet configuration for a desired level of performance. Given a fixed number of channels and a mean delay requirement, the optimal network configuration is the one which allows the most throughput with respect to the number of calls that can be handled at the given level of performance.

4.3.1 MTS Traffic Model

The Erlang-B traffic model is generally accepted as a realistic representation of independent telephone traffic when the number of terminals is large in comparison with the number of available channels [35]. However, MTS traffic is not modelled identically as Erlang-B; slight differences are inherent because MTS traffic is subject to signalling and propagation delays which add to the exponentially distributed holding time. With respect to overhead, each telephone call is handled as if it were a single sized closed batch radio call as shown in Figures 5 and 6.

4.3.2 MRS Network Model for Multiple Dispatch Subnets

In the face of a combinatorial explosion, and limited computational resources, an exhaustive search for the optimal network configuration cannot be performed. Hence, the first step in limiting the number of possible combinations is to assume homogeneous subnet

configurations. This is a reasonable assumption since subnets that are either too small or too large exhibit unfavourable delay characteristics, and are considered implausible. In practice, more moderately sized subnets are more suitable for the MRS services being offered, and are likely to be more easily manageable from an administration standpoint. Very small subnets could be serviced simply by MTS, whereas large subnets will demand dedicated channels under each spot-beam.

Given the assumption of homogeneous subnet configurations, systematic attempts to achieve a globally optimal network configuration further result in the derivation of implausible subnets where there is an inordinately large number of dispatchers. This is a consequence of a predominant trend to completely eliminate dispatcher wait by allowing enough dispatchers for the maximum possible number of batches. In other words, if no constraints are placed on configuration parameters, the number of dispatchers increases until no dispatcher contention exists. Hence, intelligent choices must be made in selecting a subnet configuration. Since actual subnets desire well utilized dispatchers, the number of dispatchers must be fixed to a reasonable number relative to the offered traffic. Upon fixing the number of dispatchers, the number of MRTs per dispatcher is estimated effectively by examining results in the previous chapter.

The question still remains as to how the number of dispatchers is determined. A subnet must not be too small, nor excessively large, and for realistic analysis, it must facilitate comparison between CB and OB systems. Therefore, a subnet configuration with four dispatchers is selected. This moderate sized subnet configuration allows CB systems to take advantage of statistical multiplexing, and OB systems to serve batches simultaneously under all spot-beams if necessary. In addition, delay performance has been shown to improve most

rapidly when the first increases in the number of dispatchers are made, so four dispatchers provides good overall performance. Furthermore, OB systems cannot take advantage of more than four dispatchers when there are four spot-beams, but with exactly four there is no risk of starvation. Overall, the four dispatcher subnet model strikes the best balance between feasibility, comparability, and performance.

Given subnets with four dispatchers, the best operating range in consideration of both delay and dispatcher utilization perspectives is in the range of 35 to 55 MRTs per dispatcher (see Figure 21). For the network model, 44 MRTs per dispatcher is selected since it is closest to the middle of the range where the MRTs can be distributed uniformly over the four spot-beams, and this configuration performs well over a wide range of batch size. The determination of the best batch size for a given performance level is investigated by experimentation.

4.3.3 Integrated Network Model Parameters

A list of fixed model parameters for both MRS and MTS in a combined system is shown in Table 7, and a list of variable parameters is shown in Table 8. The batch servicing discipline may also be either closed or open.

The mean MTS call holding time is typical for telephone service [31], but this value has been selected specifically such that the offered traffic per MTT is 0.02 Erlangs — the same as the per MRT offered traffic for MRTs in the absence of queuing. Having the same offered traffic per terminal is not essential, but it makes some comparisons between the two services more simple. In the literature, comparisons of traffic handling capacities between heterogeneous traffic sources are often made by comparing the number of terminals that are accommodated; however, these same comparisons are often made between terminals which

59

offer considerably different levels of traffic and, therefore, might be misleading [54,53].
When each type of terminal offers approximately the same amount of traffic, comparisons
can be made easily because the number of terminals for a given service is a reasonable
representation of the level of offered traffic. Nevertheless, there is no one standardized
method for comparing heterogenous traffic, and there is considerable disagreement in the
literature as to which method is best [35,53]. However, the direct comparison between
MRS and MTS traffic handling capacities is of secondary consideration; the performance
enhancement of combined network services is of paramount importance.

MTS mean call holding time, $1/\mu_T$	120 s
Control message serializing delay, T_m	50 ms
Propagation delay, T_m	250 ms
MRT call rate, λ_R	.001 calls/s
MRS mean holding time, $1/\mu_R$	20 s
Number of spot beams, s	4
Number of dispatchers per subnet, d	4
Number of MRTs per dispatcher	44
Total number of channels	35

Table 7 Fixed Network Model Parameters

Number of subnets in the network	1 - 50
MTS call rate, λ_T	.02 - .20
Batch size, b	1 - 20
Size of reserve channel margin	1 - 10

Table 8 Variable Network Model Parameters

60

The propagation delay, message serializing delay, MRT mean call rate, MRT mean call holding time, and the number of spot-beams are identical to those used in Chapter 3, and the rationale for selecting these parameters is discussed there. The rationale for choosing the selected subnet configuration, including the number of dispatchers, the number of MRTs per dispatcher, and the use of homogeneous subnets has been discussed previously.

In actual implementation, LMSS transponders are expected to provide several hundred channels for voice services [15]. Nevertheless, the total number of channels used in the simulation model analysis here is 35. While less than the expected number of channels in actual implementation, 35 channels is the upper functional limit given the constraints of the available computer resources (see Appendix B). This is a sufficient number to illustrate the dynamic channel allocation scheme.

4.4 Simulation Modelling

Before it is possible to determine the degree of improved performance offered by the proposed MRS and MTS integrated network strategy, a basis for comparison must be established. The basis is established by first observing the performance of separate MRS and MTS networks with a separate channel pool for each service. The merit of the proposed system may then be determined by observing the level of improvement over the separate channel pool method.

4.4.1 MRS and MTS Performance Objectives

A level of performance, in terms of both blocking probability for MTS and delay for MRS, must be established for comparison purposes. Two different combinations of network performance are examined here:

1. MRS delay = 2 minutes, MTS blocking Probability = 0.01

2. MRS delay = 5 minutes, MTS blocking Probability = 0.05

For convenience, these two models are subsequently referred to as the high performance model (shortest delay and smallest blocking probability) and the low performance model respectively.

A blocking probability of 0.01 is often used in the analysis of telephone traffic, and 0.05 is not uncommon. In fact, in many mobile networks, the blocking probability far exceeds 0.05 during the busy period [55]. The delay of two to five minutes is assumed acceptable for MRS. It is common within heavily utilized urban dispatch environments to be faced with dispatcher delays of this order. Furthermore, since LMSS is rurally oriented, delay of a few minutes is a great improvement in many circumstances, especially considering that no service of any kind may have been previously provide.

4.4.2 Methodology

In examining the fixed allocation scheme, several channel divisions are used. Starting with zero channels for MRS and 35 for MTS, the number of MRS channels is incremented by 5 while the number of MTS channels is correspondingly decremented by 5, until there are no channels available to MRS and 35 available to MTS — eight static divisions in all. For each division, as much MRS and MTS traffic as possible is applied to each of the two respective channel pools until the desired levels of performance can no longer be maintained. The experiment is repeated for each of the four possible combinations of performance levels and batch servicing disciplines. Experimentation with various batch sizes is also undertaken to ensure the most effective batch size is chosen in minimizing delay for each case.

In order to determine the additional number of terminals that can be serviced using dynamic allocation, the above process is repeated with some modification. For each experiment, the channel pools are merged into a single pool of 35 channels and traffic levels are set according to the results obtained under the separate channel allocation scheme. Then, only the MRS traffic is increased while the batch size and free channel margin are varied until the highest level of throughput is achieved. Experimentation ceases when the specified performance level can no longer be maintained in the face of increasing traffic. The experiment is then repeated, except the MRS traffic is held constant and the MTS traffic is increased until performance degrades below the specified level.

The level of MRS traffic is varied by increasing or decreasing the number of subnets present in the network, whereas the MTS traffic level is varied by increasing the rate at which calls come in, λ_T (see Appendix B). Experimentation over a wide range of batch sizes and free channel margins provides the best performance that may be realized for each combination of offered MRS and MTS traffic. In practice, a trade-off between an increase in one type of offered traffic against the other could be made, but for illustration purposes changes in each type of traffic are examined independently.

4.5 Results

Figures 23 through 26 show the results obtained for each respective performance level and service discipline combination. Static and dynamic allocation results are included in the figures for comparison. The downward sloping lines in the figures represent the number of MTTs while the upward sloping lines represent the number of MRTs. The horizontal axes represent the size of the MRS channel pool, c_R, under the fixed allocation scheme, where it follows that the size of the MTS channel pool, c_T, under the fixed allocation is $35 - c_R$

(for an overall total of 35 channels). As more channels are dedicated to MRS, more MRTs are accommodated; at the same time, more MRS channels means fewer for MTS, and a resultant decrease in the number of MTTs. Hence, as the number of terminals under one service increases, the number under the other declines.
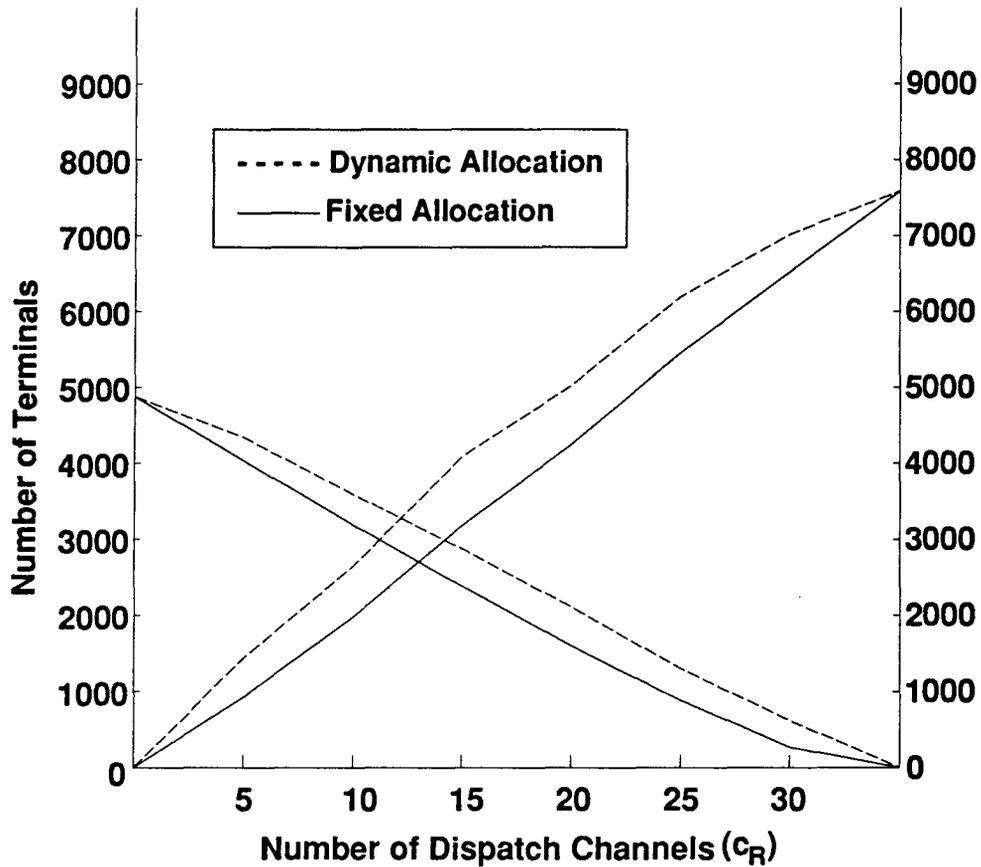
Figure 23 High Performance CB Channel Capacities

The solid lines in each pair of sloped lines represents the number of terminals that can be accommodated under the fixed allocation scheme, whereas the dashed lines represent the number of terminals that can be handled using the dynamic strategy. As outlined in the methodology of the previous section, a change shown in the number of terminals of one type is dependent upon the traffic being held constant for the alternative service. Numerical

64
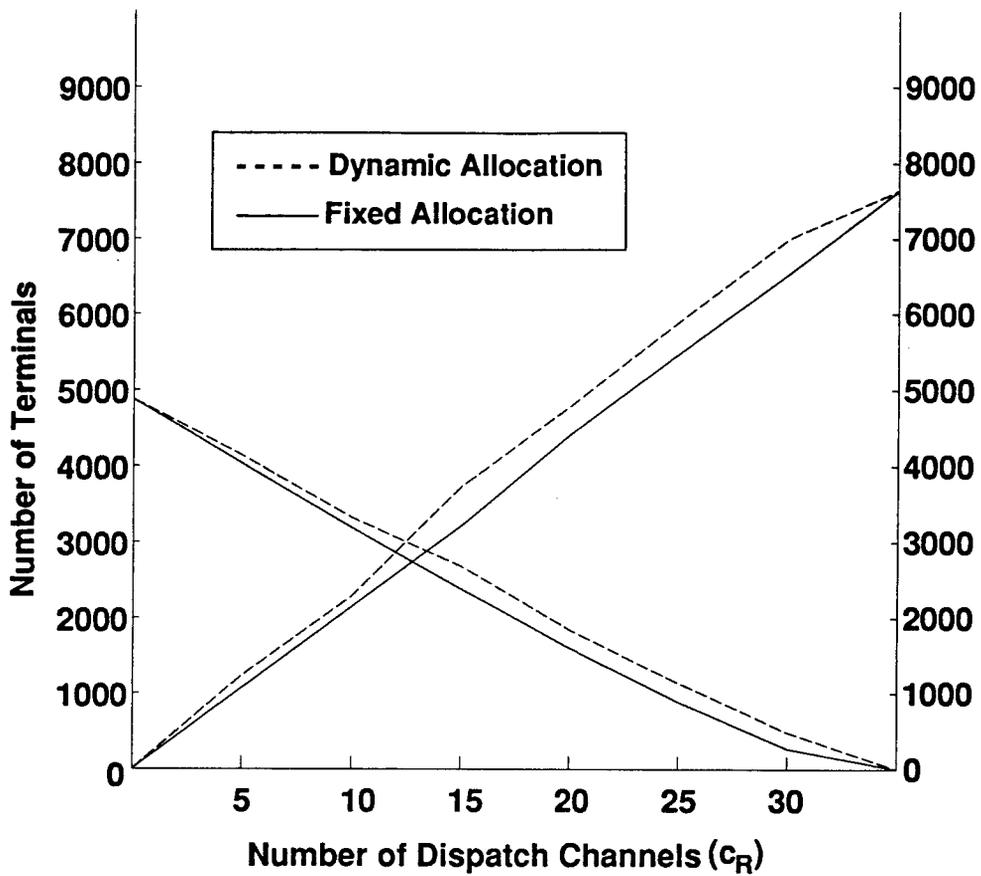
Figure 24 High Performance OB Channel Capacities

quantities corresponding to the graphs in Figures 23 through 26 are found in Tables 9 through 12 respectively. The definition of symbols used to head columns in Tables 9 through14 are as follows:

- n — number of MRTs

- Δn — increase in n

- m — number of MTTs

- Δm — increase in m

- $\Delta \%$ — percent increase

| Radio | | | | Telephone | | | |
| Static | | Dynamic | | Static | | Dynamic | |
c_R	n	Δn	$\Delta \%$	c_T	m	Δm	$\Delta \%$
0	0	0	0	35	4879	0	0
5	927	487	53	30	4028	312	8
10	1980	663	33	25	3192	401	13
15	3174	908	29	20	2382	492	21
20	4245	777	18	15	1606	503	31
25	5459	738	14	10	883	422	48
30	6524	492	8	5	270	315	116
35	7590	0	0	0	0	0	0

Table 9 High Performance CB Dynamic Allocation Improvements

| Radio | | | | Telephone | | | |
| Static | | Dynamic | | Static | | Dynamic | |
c_R	n	Δn	$\Delta \%$	c_T	m	Δm	$\Delta \%$
0	0	0	0	35	4879	0	0
5	1072	162	15	30	4028	110	3
10	2143	138	7	25	3192	132	4
15	3209	516	16	20	2382	299	13
20	4406	369	8	15	1606	240	15
25	5475	423	8	10	883	254	29
30	6538	465	7	5	270	211	78
35	7646	0	0	0	0	0	0

Table 10 High Performance OB Dynamic Allocation Improvements

Figures 23 and 24, and Tables 9 and 10 show that significant increases in traffic handling capacity can be achieved by employing dynamic channel sharing between the two services. Results are excellent for both CB and OB service disciplines with the high performance model. When the low performance model is examined, it appears that the CB discipline continues to work well (Figure 25, Table 11), although improvements are not quite as pronounced. On the other hand, the OB systems show little or no improvement; in fact, they often show degradation in performance (Figure 26, Table 12). This phenomenon is further discussed below.

For CB and OB comparison, Figures 23 and 24 have been combined in Figure 27 (high performance models), and Figures 25 and 26 have been combined in Figure 28 (low performance models). Notice that in each of the new figures, the line showing the number of MTTs under static allocation is the same for CB and OB systems, so those lines are superimposed. In Figure 27 it can be seen that, while both CB and OB systems offer higher throughput with dynamic allocation, there are some performance differences. In particular, an OB system offers greater capacity than a CB system under fixed allocation. However, under dynamic allocation, CB systems significantly surpass their OB counterparts in traffic handling capacity. Differences in traffic handling capacity are even more noticeable with the low performance model shown in Figure 28; here it can be seen that a CB system outperforms an OB system regardless of which channel allocation method is employed.
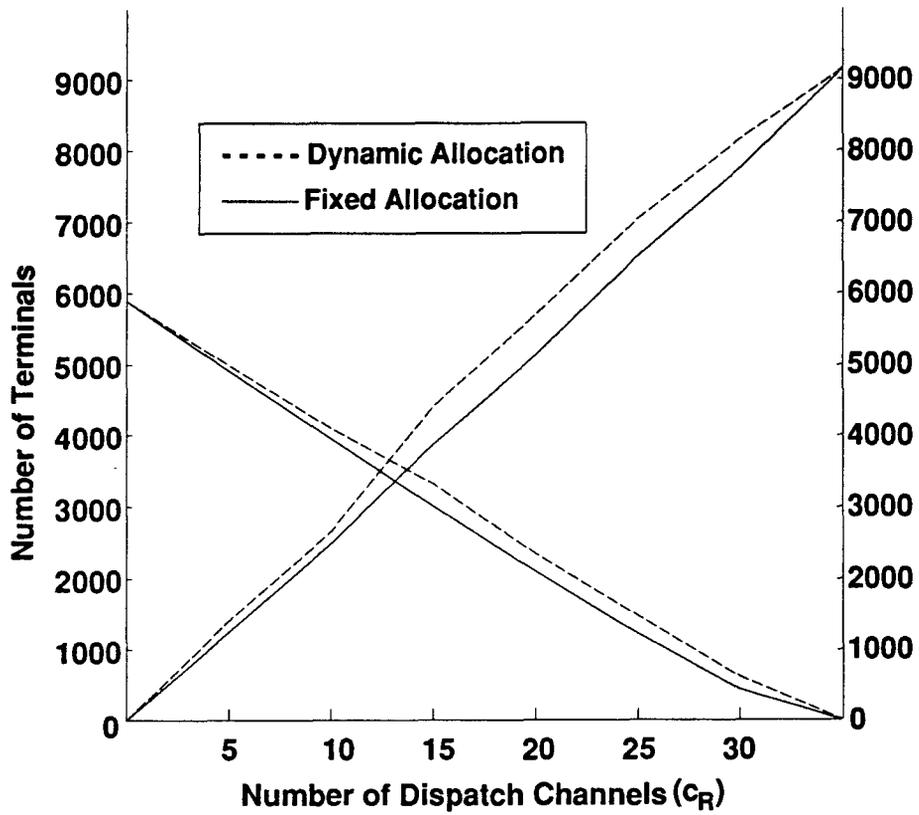
Figure 25 Low Performance CB Channel Capacities

Radio				Telephone			
Static		Dynamic		Static		Dynamic	
c_R	n	Δn	$\Delta\%$	c_T	m	Δm	$\Delta\%$
0	0	0	0	35	5882	0	0
5	1251	159	13	30	4908	82	2
10	2494	170	7	25	3960	143	3
15	3877	531	14	20	3019	314	10
20	5122	571	11	15	2105	248	12
25	6516	524	8	10	1231	250	20
30	7755	409	5	5	439	176	40
35	9154	0	0	0	0	0	0

Table 11 Low Performance CB Dynamic Allocation Improvements
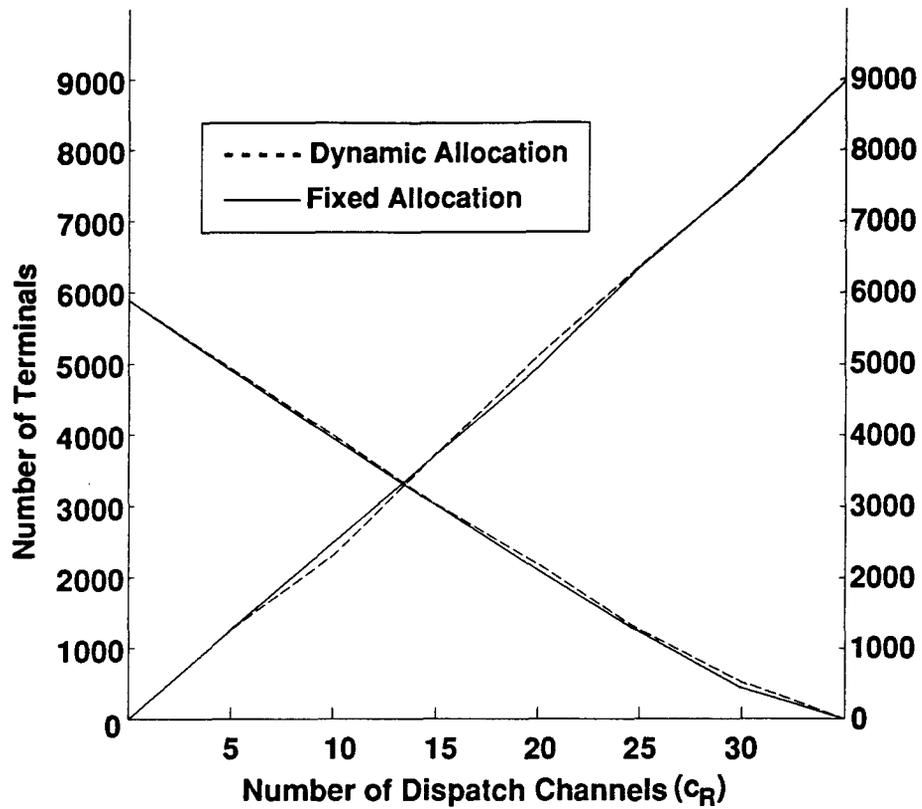
Figure 26 Low Performance OB Channel Capacities

Radio				Telephone			
Static		Dynamic		Static		Dynamic	
c_R	n	Δn	$\Delta\%$	c_T	m	Δm	$\Delta\%$
0	0	0	0	35	5882	0	0
5	1258	-4	0	30	4908	30	1
10	2491	-180	-7	25	3960	40	1
15	3726	3	0	20	3019	11	0
20	4936	158	3	15	2105	87	4
25	6349	23	0	10	1231	332	27
30	7584	-11	0	5	439	83	19
35	8981	0	0	0	0	0	0

Table 12 Low Performance OB Dynamic Allocation Improvements

When considering CB systems, greater improvements in traffic handling capacity are seen in the high performance model than in the low performance model. The reason for this is largely due to factors inherent in the achievement of the desired MTS blocking probability. There must be a much higher mean percentage of under-utilized channels in an Erlang-B model when the blocking probability is small. For example, if there are 30 channels available for service requiring a blocking probability of 0.01, then an average of 9.7 channels would be free; alternatively, a service offering a blocking probability of 0.05 results in a mean of only 5.2 free channels. Hence, as one might expect from this example, a small Erlang-B blocking probability affords a higher number of potential free channels for sharing than a large blocking probability. Tables 13 and 14 show the number of mean free channels for the high performance and low performance models, respectively.

In the tables, following the columns indicating the number of channels given under the static allocation scheme, is a column showing the mean number of free channels using a fixed MTS channel pool of the given size when meeting the given blocking probability. The next columns show first the specified free channel margin for which an attempt is made to maintain, and second the mean free margin that is actually obtained. Tables 13 and 14 use the following symbols (not previously defined) in column headings:

- $(1 - \rho)c_T$ — mean number free channels for a fixed channel allocation telephone network with the given number of channels

- R — mean number of free telephone channels realized after MRS traffic had been increased as much as possible, while MTS traffic was held constant

- T — mean number of free telephone channels realized after MTS traffic had been increased as much as possible, while MRS traffic was held constant

70
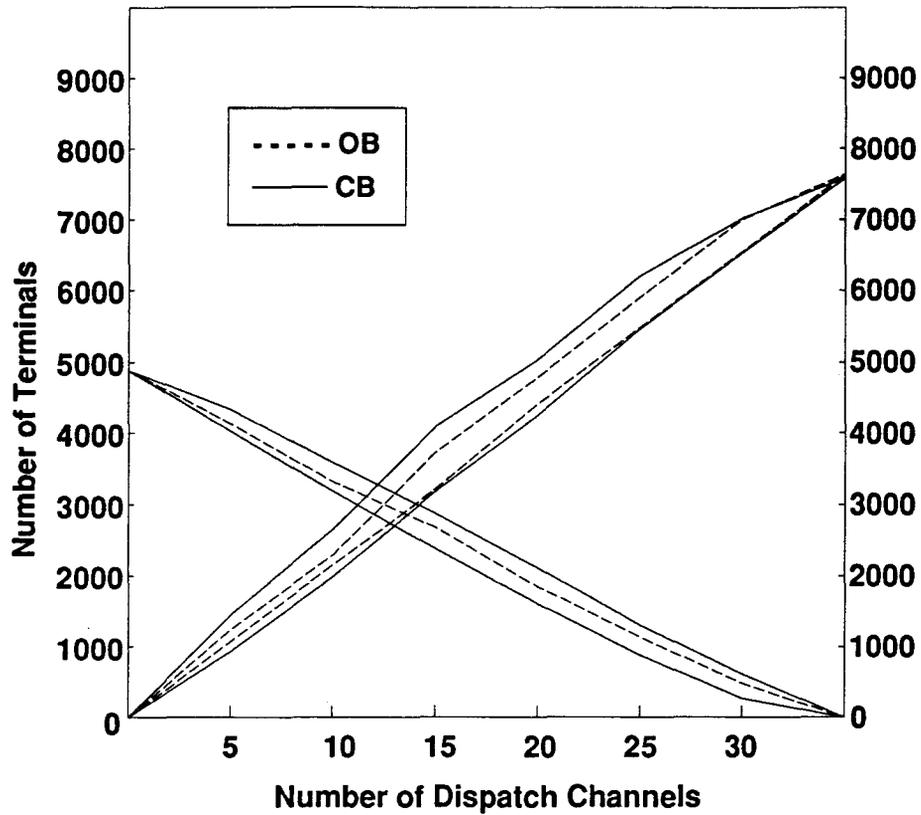
Figure 27 High Performance CB and OB Channel Capacities

High Performance Model							
Static Allocation Strategy		Dynamic Allocation Strategy					
		Closed Batch			Open Batch		
c_T	$(1-\rho)c_T$	R	T	RM	RM	T	R
30	9.7	6.9	8.1	4	9	8.6	8.3
25	8.9	4.8	6.1	3	8	7.9	7.5
20	8.0	3.1	4.3	3	6	5.9	5.1
15	6.9	2.8	3.4	3	5	5.0	4.3
10	5.5	1.9	2.4	2	4	3.5	3.5
5	3.6	1.6	1.7	1	2	2.4	2.4

Table 13 Mean Number of Free Channels for High Performance Model
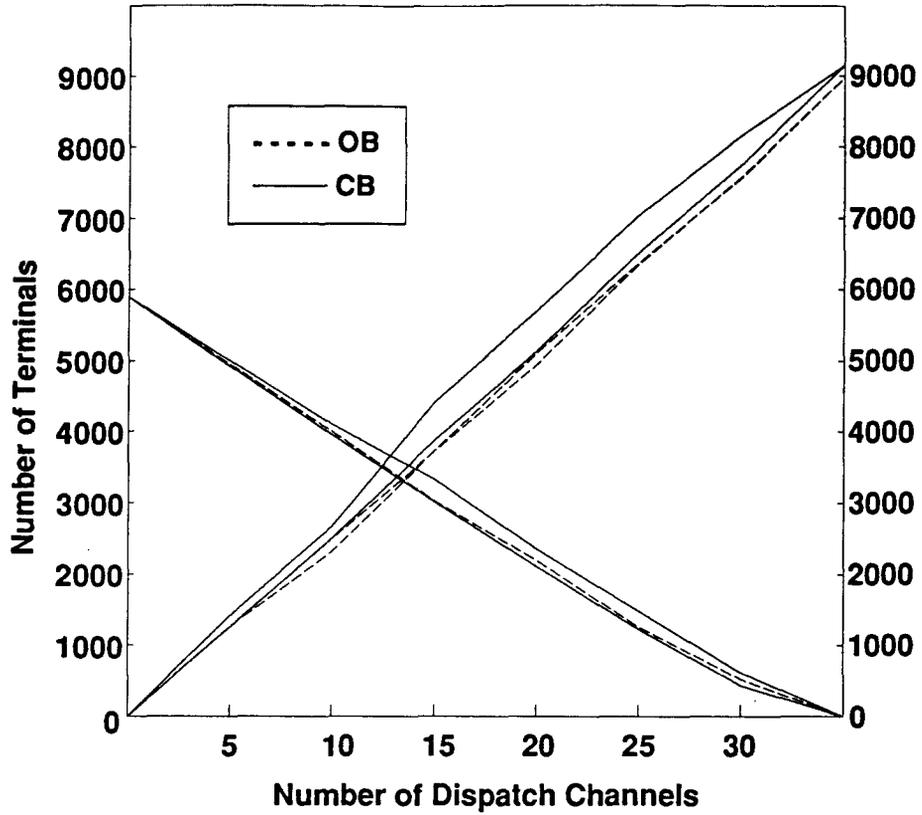
Figure 28 Low Performance CB and OB Channel Capacities

Low Performance Model							
Static Allocation Strategy		Dynamic Allocation Strategy					
		Closed Batch			Open Batch		
c_T	$(1 - \rho)c_T$	R	T	*RM*	*RM*	T	R
30	5.2	4.7	4.9	5	11	5.8	5.9
25	5.0	3.7	3.9	4	10	5.6	5.7
20	4.8	2.8	3.0	3	8	4.8	4.8
15	4.4	2.5	2.4	3	6	4.6	4.3
10	3.8	1.8	1.7	2	4	3.5	3.4
5	2.8	1.5	1.9	1	3	2.4	2.2

Table 14 Mean Number of Free Channels for Low Performance Model

Note that the number of free channels which the system attempts to maintain (RM) is not generally equal to the actual mean number of free channels that is realized (R or T). Suppose that M is the number of free channels desired and that at some time there are exactly RM free channels on reserve. Further suppose that all other channels are being used. If a telephone call arrives before any of the calls in service is complete, then one channel is used to serve the incoming call and only $RM - 1$ free channels remain. More incoming telephone calls may reduce the margin even further. Blocking occurs when additional telephone calls arrive but there are no more free channels in reserve. On the other hand, if traffic is low and there is no queue to service dispatch batches, the actual number of free channels may be greater than RM. Hence, it is apparent that the observed mean number of free channels is generally not equal to the specified minimum number of free channels which the system strives to maintain.

Open batch systems, however, do not behave as predictably as CB systems. The reason for the apparent inconstant behavior of OB systems is due to their variable batch size capability. Since batch sizes are not fixed, calls joining a batch which has not completed service add to the time a dispatcher hold a channel. When channels are often held for relatively long periods of time, dispatchers queued for channel allocation find larger batches accumulating in queue due to extended channel waiting delays. Batches which are larger at service commencement ultimately result in still longer channel holding time and, hence, the problem compounds until equilibrium again is reached.

Table 15 shows the optimal batch sizes used for each service discipline and performance model in the experiments. In the case of OB systems, there are two numbers for each case. The first number is the minimum batch formation threshold, and the second number (in parenthesis) is the mean number of calls processed in a batch before the dispatcher actually

relinquishes an assigned channel. Note that for the high performance model, the actual number of calls processed in an OB averages between 10 and 16, and the number of calls per OB in the low performance model is approximately four times as high.

c_R	High Performance		Low Performance	
	Closed	Open	Closed	Open
5	4	1 (16)	7	2 (65)
10	4	1 (15)	7	3 (62)
15	4	1 (12)	8	4 (60)
20	4	1 (11)	8	4 (50)
25	4	1 (12)	8	5 (47)
30	4	1 (10)	8	5 (40)
35	4	1 (10)	8	5 (38)

Table 15 Optimal Batch Sizes

When performance levels are kept high, open batches remain relatively small, but as performance is allowed to drop, large batches result and detrimental effects become evident. However, the underlying cause of performance degradation is more easily understood by considering the process of channels being relinquished on batch service completion. Because large batches cause dispatchers to hold channels for extended periods of time, the time periods between channel relinquishment are extended. As a result of an infrequent channel relinquishment process, the dynamic channel allocation system cannot react quickly enough to adjust to changing traffic intensities, and both services can end up adversely affected due to frequent traffic imbalances that cannot take advantage of the statistical smoothing of the dynamic allocation system.

Open batch systems have another major disadvantage associated with them. The problem may be explained by examining the variance in the mean delay. Table 16 shows comparisons

between the delay variances of OB and CB systems in relative terms, where the variance ratio is defined as the OB delay variance divided by the CB delay variance.

c_R	Variance Ratio (CB variance (min^2))	
	High Performance	Low Performance
5	6.23 (1.02)	6.75 (3.48)
10	2.98 (0.93)	5.34 (2.51)
15	1.43 (0.97)	3.82 (2.72)
20	2.23 (1.07)	5.49 (2.35)
25	1.93 (1.02)	4.65 (2.43)
30	4.54 (0.39)	4.61 (2.20)
35	2.21 (0.89)	5.15 (2.30)

Table 16 CB and OB Variance Comparison

One reason for the larger OB variance is the channel queuing delay. Define t_{QC} as the mean absolute channel request delay at the dispatcher between the time a channel is requested and the time it is assigned. Note the difference between t_{QC} and T_{QC}, where the latter is the mean channel request delay *per call*. Table 17 shows the t_{QC} and T_{QC} for each number of radio channels. For CB systems, t_{QC} translates directly into T_{QC}, since all calls in a closed batch face the same absolute channel request delay.

On the other hand, OB calls that arrive after a dispatcher has been seized (it is assumed a channel is requested immediately upon receiving a dispatcher) do not have to wait the full length of the absolute channel request delay. Open batch calls that arrive after a channel request has been made do not have to wait the full t_{QC}, and those calls which arrive after a channel has been assigned face no channel queuing delay at all. This wide relative variability of channel queuing delays combined with a large t_{QC} accounts for the much larger delay

variance in OB systems. Table 17 shows a comparison between t_{QC} for CB and OB systems under the fixed channel allocation scheme. An analogous situation occurs with the dispatcher delay, T_{WD}.

Number of Radio Channles	t_{QC}			
	High Performance Model		Low Performance Model	
	CB	OB	CB	OB
5	11.6	104.8	67.2	555.3
10	12.4	63.9	31.4	421.1
15	12.4	51.1	68.1	408.3
20	18.4	139.9	68.2	435.3
25	18.0	95.4	73.7	460.5
30	15.6	60.3	63.4	428.0
35	14.8	117.9	65.4	506.1

Table 17 Channel Queuing Delay

The variance for OB systems is always larger than similar CB systems, and the differences are more pronounced with the low performance model. The problem with a large variance, even with systems exhibiting favourable performance in terms of mean delay, is that wide variations in delay at the MRT level are likely to cause users to viewed the system as unreliable. Therefore, inconsistent service is likely to lead to customer dissatisfaction — a key concern in performance evaluation for overall network design.

Since the batch size is fixed in CB systems, they do not experience the same shortcomings as OB systems. Consistently small batch servicing times allow for fast adjustment under the dynamic allocation strategy, and provide a more uniform service to MRTs.

Aside from comparisons between CB and OB systems, a comparison relating the low
and high performance models for each of the two batch service disciplines is also considered.
Figure 29 combines Figures 23 and 25 to compare low and high performance models for CB
systems, and Figure 30 combines the OB performance model graphs from Figures 24 and
26. It is observed that even though the high performance model offers greater improvement,
but as expected the low performance model is ultimately capable of handling more traffic.
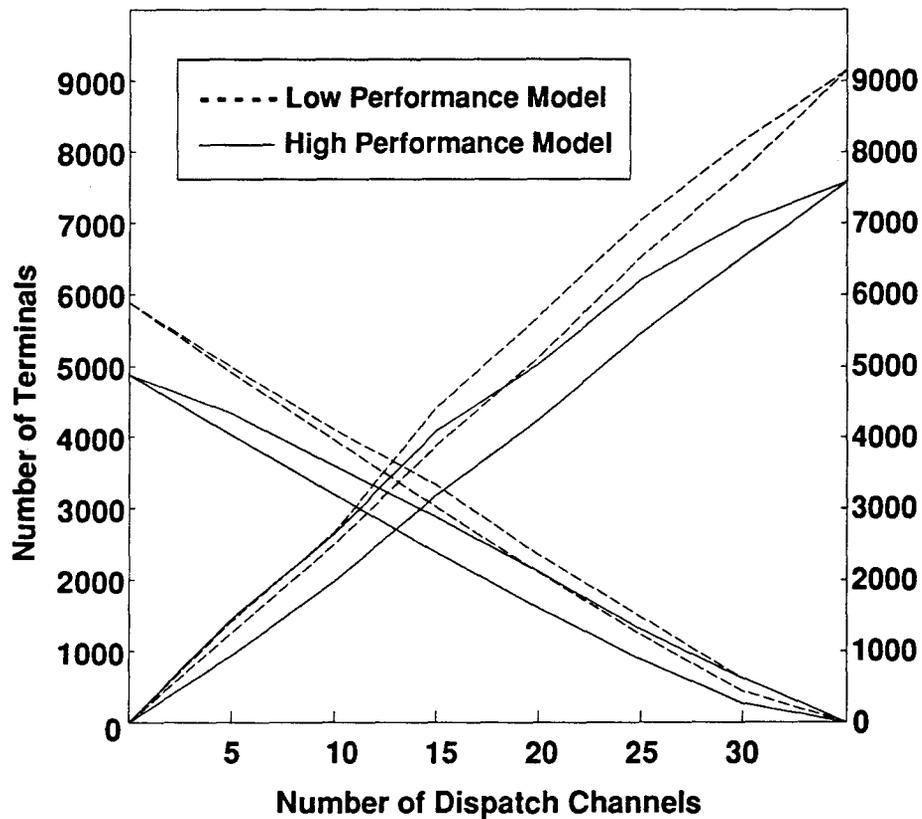
Figure 29 Closed Batch High and Low Performance Models
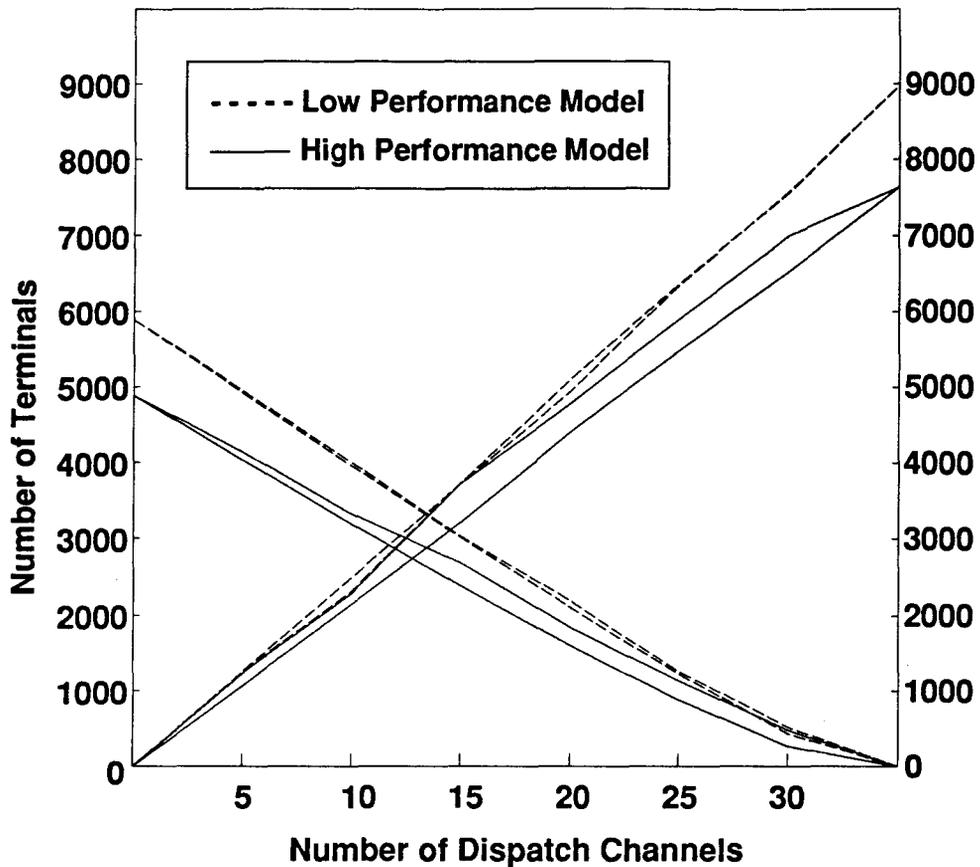
Figure 30 Open Batch High and Low Performance Models

4.6 Summary and Observations

A model of the integrated MRS and MTS network is constructed which is consistent with the proposed LMSS network configuration. The MRS segment operates under the new dispatch radio protocol discussed above, and the ᵐMTS segment is modelled as an infinite population Erlang-B traffic source. The networks are combined under two different channel allocation strategies (static and dynamic), and the results are compared.

Under all traffic divisions and performance model combinations, CB systems using dynamic allocation show improvements in dispatch and telephone traffic throughput over static channel pool division. Increased throughput in CB systems is achieved with no sacrifice

in performance. Qualitatively, OB systems show similar behavior with the high performance model, but no real improvement with the low performance model.

Closed batch systems using the proposed dynamic channel allocation scheme perform better than any static system or any OB system model. In addition, CB systems offer a smaller variance in call delay. While it would seem that CB systems are always the clear favourite, OB systems still offer better service when traffic levels are relatively low.

The dynamic channel allocation system works by reducing the mean number of free channels required to ensure adequate blocking probability for blocked-calls-dropped traffic. Table 13 and 14 summarize the reductions in the mean number of idle channels for the different levels of offered traffic. From these tables, it can be seen that the tentative margin of free telephone channels, M, can be either smaller or larger than the actual mean number of free channels realized depending on service discipline and performance level. Increases in traffic due to varying MRS or MTS traffic also have an effect on the mean number of free channels. The difference is often small, but usually results in a smaller number of free channels being necessary when radio traffic is increased. This is due to the fact that radio calls are considerably shorter than telephone calls. Shorter calls means more frequent call completions and, ultimately, more rapid adjustments in the free channel margin to accommodate changing traffic. The speed of margin adjustment is fundamentally related to the size of the free channel margin; if adjustments could be made instantaneously, then the tentative free channel margin would never have to be greater than unity, since as soon as a telephone call seized the free channel, another channel would be made available.

5. Conclusion

Both the new MRS protocols and the proposed reserve margin dynamic channel allocation strategy provide means to make more efficient use of available bandwidth for LMSS voice services. The MRS protocol conserves bandwidth by reducing inband signalling requirements by processing calls in batches, and reducing sideband signalling by queuing blocked calls to avoid repeated call requests. Further conservation of bandwidth is achieved by employing the dynamic channel allocation scheme use to combine radio and telephone which takes advantage of channels that often remain idle when used in a telephone only, blocked-calls-dropped, network.

5.1 Summary

Given an integrated MRS and MTS network, the selected subnet configuration, any level of offered traffic, and one of the two performance levels, CB systems under the reserved margin dynamic allocation system perform better than any OB system or any system under static allocation. In addition, it appears that CB systems are robust when faced with various performance levels.

When offered traffic is low, OB systems might offer lower delay. Regardless of which service discipline is employed, time-outs might be used to provide better service at low traffic levels, and priorities might be incorporated to service urgent messages at any traffic level. The batch size might also be adapted dynamically to reflect the current traffic load. If the load is light, a smaller batch size can be used to reduce the batch formation delay. With

regard to the dynamic channel allocation system, the tentative free channel margin might also be adjusted to adapt to changes in the load balance between MRS and MTS traffic.

An interesting consequence of the integrated system, is that greater additional throughput is achieved when required blocking probability is small. This is due to the fact that, on average, more channels must be free to maintain a small blocking probability and therefore more channels can be utilized under the reserve margin dynamic allocation scheme.

Intelligent choices have been made in constructing a model from that which is known about planned LMSS voice services and in consideration of available computing resources. While no model is ideal, the results here clearly indicate the superiority of the analyzed systems. The effectiveness of the reserved margin strategy and the CB systems during the busy period allows us to relax some of the network parameter constraints selected earlier. It follows that subnets need not be constrained to having fewer dispatchers than the number of spot-beams. The proposed reserve margin dynamic allocation strategy not only facilitates additional throughput over static allocation for all CB systems, but is also intrinsically simple to implement. In addition, the strategy is easily adapted to other systems of heterogeneous traffic where offered traffic is a combination of blocked-calls-dropped and blocked-calls-queued disciplines. By virtue of the fact that CB systems exhibit optimal throughput when using a batch size greater than unity, the batching strategy is shown to work effectively. The reserved margin dynamic channel allocation strategy is also shown to improve throughput. Overall, the systems constitute innovative and practical contributions to efficient use of scarce channel resources for LMSS.

5.2 Future Work

Due to the constraints of both time and computational power, an exhaustive study has by

no means been done. Nevertheless, insights gained here into system behavior may be used as a foundation for further study. Several aspects of the system may be further investigated.

First of all, only two possible batch service disciplines have been tested, and there might be some better than the FCFS-OB system. One possibility is a hybrid of the CB and OB systems where batches remain open until service begins, but once a channel is allocated the batch closes to new arrivals. This system might provide a combination of advantages of the CB and OB systems, allowing statistical multiplexing within subnets and at the same time preventing excessively long channel holding times.

A greater range of performance models could also be investigated. Additionally, a method for relating blocking probability to delay in order that the channel margin may be adjusted to maintain equal levels of performance at all times. This would allow system performance to be truly balanced between services. In reality, the most effective performance measure would be a cost function which can be applied to maximize LMSS revenue. In order to achieve the best cost function, a degree of adaptability might also be incorporated, where time-outs and priorities might be used, and where the batch size, batching strategy, and free channel margin might also be modified dynamically also.

Larger simulations might be considered to more accurately represent an entire LMSS system. Since larger systems tend to behave more efficiently than proportionally smaller ones due to statistical smoothing, the benefits of the dynamic allocation strategy may not be as pronounced in a large network. On the other hand, a larger network has a much faster and much more uniform channel relinquishment process which would allow the system to adapt much more quickly. Hence, it might be the case that even a proportionally much smaller number of free channels is required to achieve the desired result. Furthermore, the proposed

system performs better when a smaller blocking probability is desired; therefore, it might be possible to considerably enhance system performance without sacrificing throughput.

A system that also incorporates a third service into a shared dynamic channel allocation scheme might also be considered. In particular, a method for integrating mobile data service to further enhance overall LMSS network bandwidth utilization may be possible..

Bibliography

[1] K. G. H., "Yesterday's Dream — Today's Reality," in *Proc. Mob. Sat. Conf.*, p. 5, 1988.

[2] A. C. Clarke, *The Nine Billion names of God*. Signet, New York, 1974.

[3] G. E. Lewis, *Communication Services Via Satellite*. BSP, London, 1988.

[4] J. Martin, *Communications Satellite Systems*. Prentice Hall, Englewood Cliffs, N.J., 1978.

[5] D. Gilvery, "Review of Canadian Mobile Satellite Systems Institutional Arrangements Policy," in *Proc. Intl. Mob. Sat. Conf.*, pp. 468–472, 1990.

[6] E. Fthanakis, *Manual of Satellite Communications*. McGraw Hill, New York, 1984.

[7] D. Athanassiadis, "Canadian Development and Commercialization of North American Mobile Satellite Service," in *Proc. Intl. Mob. Sat. Conf.*, pp. 438–442, 1990.

[8] C. E. Agnew et al, "The AMSC Mobile Satellite System," in *Proc. Mob. Sat. Conf.*, pp. 3–9, 1988.

[9] V.C.M. Leung, M.O. Ali and A. Spolski, "An Efficient Demand-Assigned Multiple-Access Scheme for Satellite Mobile Radio Dispatch Networks," *IEEE Trans. Veh. Tech.*, vol. 38, no. 4, pp. 204–210, 1990.

[10] D. Sward, "Mobile Satellite Service for Canada," in *Proc. Mob. Sat. Conf.*, pp. 23–29, 1988.

[11] J. Freibaum, "International and Domestic mobile Satellite Regulatory Proceedings: A Comparison of Outcomes and Discussion of Implementations," in *Proc. Mob. Sat. Conf.*, pp. 71(a)–71(f), 1988.

[12] R.J. Arnold, "US Development and Commercialization of a North American Mobile Satellite Service," in *Proc. Intl. Mob. Sat. Conf.*, pp. 431–436, 1990.

[13] C. S. Kim, "Omni-Directional L-band Antenna for Mobile Communications," in *Proc. Mob. Sat. Conf.*, pp. 255–260, 1988.

[14] Federal Department of Communications, "Personal Communication," *Ottawa, Canada*, 1991.

[15] W. Rafferty, K. Dessouky, and M. Sue, "NASA's Mobile Satellite Development Program," in *Proc. Mob. Sat. Conf.*, pp. 11–22, 1988.

[16] W. Nowland, M.Wagg, and D. Simpson, "AUSSAT Mobile Satellite Services," in *Proc. Mob. Sat. Conf.*, pp. 31–36, 1988.

[17] A. Pedersen, "Canadian MSAT Field Trial Program User Requirements," in *Proc. Intl. Mob. Sat. Conf.*, pp. 717–721, 1990.

[18] A. Arcidiacoco, "Integration Between Terrestrial-Based and Satellite-Based Land Mobile Communication Systems," in *Proc. Intl. Mob. Sat. Conf.*, pp. 39–45, 1990.

[19] B. Azarbar, "An Upward Compatible Spectrum Sharing Architecture for Existing Actively Planned and Emerging Mobile Satellite Systems," in *Proc. Intl. Mob. Sat. Conf.*, pp. 456–461, 1990.

[20] O. Lundberg, "Mobile Satellite Services: International Coordination, Cooperation and Competition," in *Proc. Mob. Sat. Conf.*, pp. 71–78, 1988.

[21] M. Wachira, "Domestic Mobile Satellite Systems in North America," in *Proc. Intl. Mob. Sat. Conf.*, pp. 19–27, 1990.

[22] V. C. M. Leung, "Management of Transponder Resources in Mobile Satellite Systems," *IEEE Trans. Aerospace and Electronic Systems*, vol. 26, no. 2, pp. 273–281, 1990.

[23] M. Razi, A. Shoamanesh, and R. Azarbar, "L-band and SHF Multiple Access Schemes for the MSAT System," in *Proc. Mob. Sat. Conf.*, pp. 373–379, 1988.

[24] G. Boulay, "MSAT: A Booster for Land Based Mobile Telecommunications Networks," in *Proc. Intl. Mob. Sat. Conf.*, pp. 712–716, 1990.

[25] P. Bartholome and R. Rogard, "A Satellite System for Land-Mobile Communications in Europe," in *Proc. Mob. Sat. Conf.*, pp. 37–42, 1988.

[26] R. A. Wiedeman et al, "System Capacity and Economic Modelling Computer Tool for Satellite Communications Systems," in *Proc. Mob. Sat. Conf.*, pp. 43–49, 1988.

[27] M. Joseph and D. Raychaudhuri, "Simulation Models for Performance Evaluation of Satellite multiple Access Protocols," *IEEE JSAC*, vol. 6, no. 1, pp. 210–221, Jan. 1988.

[28] A. Salmasi, "An Over View of the Onmitracs — The First Operational Mobile Ku-band Satellite Communications," in *Proc. Mob. Sat. Conf.*, pp. 63–68, 1988.

[29] D. Bertsekas and R. Gallager, *Data Networks*. Prentice Hall, Englewood Cliffs, N.J., 1987.

[30] G. Hess and J. Cohn, "Communication Load and Delay in Mobile Trunked Radio Systems," in *Proc. 31th IEEE VTC*, pp. 269–273, 1981.

[31] J. K. S. Sin and N. D. Georganas, "A Simulation Study of a Hybrid Channel Assignment Strategy for Cellular Land-Mobile Radio Systems with Erlang-C Service," *IEEE Trans. Comm.*, vol. 29, no. 2, pp. 143–147, Feb. 1981.

[32] Y. Doganata, "A Shared Service Algorithm for a Trunked Mobile System," *IEEE Trans. Veh. Tech.*, vol. 35, no. 2, pp. 93–99, Aug. 1986.

[33] H. H. Hoang and P. C. Cohen, "A Model for Channel Sharing in Land Mobile Radio Dispatch Services," *IEEE Trans. Veh. Tech.*, vol. VT-34, no. 2, pp. 76–85, 1985.

[34] P. Cohen, "Traffic Analysis for Different Classes of Users of Land Mobile Radio Communication Systems," in *IEEE Veh. Tech. Conference*, pp. 283–285, May 1983.

[35] R. M. G. Chan, H. H. Hoang, "Traffic Engineering of Trunked Land Mobile Radio Dispatch Systems," in *Proc. 41st Veh. Tech Conference*, pp. 251–256, 1991.

[36] D. Raychauduri and K. Joseph, "Ku-band Satellite Data Networks Using Very Small Aperture Terminals — Part I: Multi-access Protocols," *International Journal of Satellite Communications*, vol. 5, pp. 195–212, 1987.

[37] S.S. Kamal, "Demand Assignment for Mobile Systems," in *Proc. IEEE Globcom*, pp. 1221–1225, 1984.

[38] M. Zhang and T. P. Yum, "Comparisons of Channel-Assignment Strategies in Cellular Mobile Telephone Systems," *IEEE Trans. Veh. Tech.*, vol. 26, no. 2, pp. 211–215, 1990.

[39] D. C. Cox, D. O. Reudink, and M. L. Honig, "Increasing Channel Occupancy in Large-scale Mobile Radio Systems: Dynamic Channel REassignment," *IEEE Trans. Veh. Tech.*, vol. 22, no. 4, pp. 218–225, Nov. 1973.

[40] M. L. Honig, "Some Effects on Channel Occupancy of Limiting the Number of Available Servers in Small Cell Mobile Radio Systems Using Dynamic Channel Assignment," *IEEE Trans. Comm.*, vol. 27, no. 8, pp. 1224–1225, Aug. 1989.

[41] M. L. Honig, "Analysis of a TDMA Network With Voice and Data Traffic," *AT&T Bell Laboratories Technical Journal*, vol. 63, no. 8, pp. 1537–1563, Oct. 1984.

[42] A. Chrapkowski and G. Grube, "Mobile Trunked Radio System Design and Implementation," in *Proc. 41st IEEE VTC*, pp. 245–250, May 1991.

[43] R. S. Lunayach, S. S. Rao, and S. C. Gupta, "Analysis of a Mobile Radio Communications System with Two Types of Customers and Priority," *IEEE Trans. Comm.*, vol. 30, no. 11,

pp. 2470–2474, 1982.

[44] T. J. Kahwa and N. D. Georganas, "A Hybrid Channel Assignment Scheme in Large-scale, Cellular-structured Mobile Communication System," *IEEE Trans. Comm.*, vol. 26, no. 4, pp. 432–438, April 1978.

[45] U. N. Bhat and M. J. Fischer, "Multichannel Queuing Systems with Heterogeneous Classes of Arrivals," *Naval Research Logistics Quarterly*, vol. 23, pp. 271–282, June 1986.

[46] M. Peritsky, "Traffic Engineering of Combined Mobile Telephone and Dispatch Systems," *IEEE Trans. Com*, vol. COM-21, no. 11, pp. 1107–1109, 1973.

[47] V. Li and C. Wu, "Integrated Voice/Data Protocols for Satellite Channels," in *Proc. Mob. Sat. Conf.*, pp. 413–421, 1988.

[48] U. R. Krieger, B. Muller-Clostermann, and M. Sczittnick, "Modeling and Analysis of Communication Systems Based on Computational Methods for Markov Chains," *IEEE JSAC*, vol. 8, no. 9, pp. 1630–1648, Nov 1990.

[49] S. N. Crozier, "Sloppy Slotted Aloha," in *Proc. Intl. Mob. Sat. Conf.*, pp. 357–362, June 1990.

[50] L. Kleinrock, *Queuing Systems Volume 1: Theory*. John Wiley & Sons, New York, 1975.

[51] R. J. Holbeche, *Land Mobile Radio Systems*. Peter Pereginus Ltd., New York, 1985.

[52] N. J. Haslett and A. J. Bonney, "Loading Considerations for Public Safety Dispatch on Trunked Radio Systems," in *Proc. 37th IEEE VTC*, pp. 24–28, 1987.

[53] D. N. Hatfield, "Measures of Spectral Efficiency in Land Mobile radio," in *Proc. 25th IEEE VTC*, pp. 23–26, 1975.

[54] R. D. Rosner, *Packet Switching*. Wadsworth Inc., Belmont California, 1982.

[55] G. Calhoun, *Digital Cellular Radio*. Artech House, Norwood, MA, 1988.

[56] R. Fairly, *Software Engineering Concepts*. McGraw Hill, 1985.

[57] A. M. Law and W. D. Kelton, *Simulation Modeling and Analysis, 2nd Ed.* McGraw Hill, New York, 1991.

[58] S. M. Ross, *Probability Models*. John Wiley & Sons, 1980.

[59] S. M. Ross, *Stochastic Processes*. John Wiley & Sons, 1983.

[60] J. A. Rice, *Mathematical Statistics and Data Analysis*. Wadsworth & Brooks, Pacific Grove, California, 1988.

[61] P. Heidelberger and P. D. Welch, "A Spectral Method for Confidence Interval Generation and Run Length Control in Simulations," *ACM*, vol. 24, no. 4, pp. 233–245, April 81.

[62] W. D. Kelton, "Transient Exponential-Erlang Queues and Steady-State Simulation," *ACM*, vol. 28, no. 7, pp. 741–749, July 1985.

[63] L. W. Schruban, "Control of Initialization Bias in Multivariate Simulation Response," *ACM*, vol. 24, no. 4, pp. 246–252, April 1981.

A. Model Validation and Verification

In order to determine the correct operation of a software project, both validation and verification are required. Validation is to ensure that the correct model is being built, whereas verification is to ensure the model is being built correctly [56,57].

There are no actual systems against which to compare either the subnet or integrated network model, since the systems are new. Therefore, MRS model validation is implicit in the model description which is provided in previous chapters and validated against the open literature to the extent possible. Validation of the MTS model is complete by accepting the assumption of Erlang-B traffic which is widely regarded as a reasonably accurate representation of telephone traffic.

Model verification, on the other hand, is still essential to ensure there are no software bugs which produce errors in program output. Some standard software verification techniques include independent program module testing, redundant collection of program statistics to ensure consistency, program execution step-throughs, and the use of predetermined test suites. All of these techniques have been employed as a matter of routine coding, except for the last one. Predetermined test suites cannot be used because no results can be determined in advance for any input. In addition, no closed form analytic solutions or numerical results exist for the particular G/G/m/n queuing systems involved in subnet modelling. Nevertheless, if the models are simplified, some mathematical models may be constructed and their results compared to the simulation models. This is the objective in the next sections.

A.1 Open Batch MRS Subnet Model

A single spot-beam OB MRS subnet employing a single dispatcher with no signalling overhead (propagation and serializing delays) and no channel assignment delay may be modelled by the imbedded Markov chain as shown in Figure 31. In the figure, n is the total number of MRTs in the subnet, b is the batch size, λ is the call rate per individual MRT, and μ is the call service rate. Before proceeding to analyze the Markov chain, some theoretical
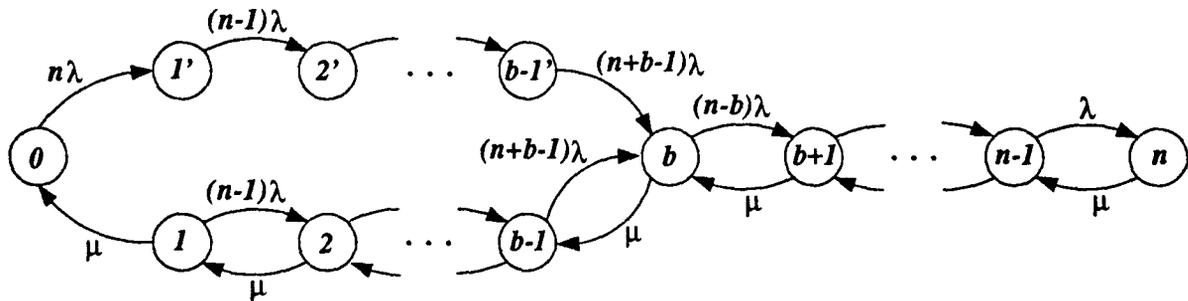
Figure 31 Open Batch, Single Dispatcher Markov Chain

results are required. First, by inspection it is noted that the Markov chain of Figure 31 is a finite state, irreducible, aperiodic chain.

Proposition B.1. All states of an irreducible aperiodic Markov chain are either:

i. positive recurrent

ii. transient

Corollary B.2. All states of a finite state irreducible aperiodic Markov chain are positive recurrent.

Corollary B.2 indicates that the Markov chain of Figure 31 is positive recurrent.

Proposition B.3. Sufficient conditions for the existence of a stationary distribution of a Continuous-time Markov Chain are:

i. the Markov Chain is irreducible

ii. the Markov Chain is positive recurrent

The Markov chain is irreducible by inspection and positive recurrent by Corollary B.2; therefore, the limiting distribution exists.

Propositions B.1 and B.3, and Corollary B.2 are offered without proof. For a more complete discussion of limiting distributions of Continuous-time Markov chains, including proofs see references [20,50,58,59].

Define p_j as the limiting probability of the chain being in state S_j. Now, since the limiting distribution exists, the Chapman-Kolmogorov equations are reduced to the following set of balance equations: For states S_1' through S_{b-1}'

$$p_1' = n\lambda p_0 + [1 - (n-1)\lambda]p_1' \tag{A.1}$$

$$p_2' = (n-1)\lambda p_1' + [1 - (n-2)\lambda]p_2' \tag{A.2}$$

$$\cdots$$

$$p_j' = (n-j+1)\lambda p_{-1j}' + [1 - (n-j)\lambda]p_j' \quad 3 \le j \le b-1 \tag{A.3}$$

The equations (A.1) through (A.3) may be more simply expressed as

$$p_j' = \frac{(n-j+1)}{(n-j)}p_{j-1}' \quad \bullet \; 1 \le j \le b-1 \tag{A.4}$$

and, solving recursively in terms of p_0,

$$p_j' = \frac{n}{(n-j)}p_0 \quad 1 \le j \le b-1 \tag{A.5}$$

Investigating states S_0 through S_{b-1}, the balance equations are

$$p_0 = (1 - n\lambda)p_0 + (1 - \mu)p_1 \tag{A.6}$$

$$p_1 = [1 - (n-1)\lambda - \mu]p_1 + \mu p_2 \tag{A.7}$$

$$p_2 = (n-1)\lambda p_1 + [1 - (n-2)\lambda - \mu]p_2 + \mu p_3 \tag{A.8}$$

$$...$$

$$p_j = (n-j+1)\lambda p_{j-1} + [1 - (n-j)\lambda - \mu]p_j + \mu p_{j+1} \quad 3 \leq j \leq b-1 \tag{A.9}$$

. Simplification of equations (A.6) through (A.9) yields

$$p_1 = n\left(\frac{\lambda}{\mu}\right)p_0 \tag{A.10}$$

$$p_2 = \left[(n-1)\left(\frac{\lambda}{\mu}\right) + 1\right]p_1 \tag{A.11}$$

$$...$$

$$p_j = \left[(n-j+1)\left(\frac{\lambda}{\mu}\right) + 1\right]p_{j-1} - (n-j+2)\left(\frac{\lambda}{\mu}\right)p_{j-2} \quad 3 \leq j \leq b \tag{A.12}$$

recursively solving equations (A.10) through (A.12) in terms of p_0,

$$p_j = \frac{np_0}{(n-j)!}\sum_{k=0}^{j-1}\left(\frac{\lambda}{\mu}\right)^{k+1}(n-j+k)! \quad 1 \leq j \leq b \tag{A.13}$$

Finally, for states S_b through S_{n-1} the balance equations are as follows:

$$p_b = (n-b+1)\lambda(p_{b-1}' + p_b) + ([1 - (n-b)\lambda - \mu])p_b + \mu p_{b+1} \tag{A.14}$$

$$p_j = (n-j+1)\lambda p_{j-1} + [1 - (n-j)\lambda - \mu]p_j + \mu p_{j+1} \quad b < j < n \tag{A.15}$$

and by simplifying and solving recursively in terms of p_0

$$p_j = \frac{np_0}{(n-j)!} \sum_{k=j-b}^{j-1} \left(\frac{\lambda}{\mu}\right)^{k+1} (n-j+k)! \qquad b < j \le n \qquad (A.16)$$

Further, the dispatcher utilization factor, ρ, may be calculated as follows:$\rho = 1 - \left(p_0 + \sum_{j=1}^{b-1} p_j'\right).$

Now define

$$\pi_j = \begin{cases} p_0 & j = 0 \\ p_j' + p_j & 1 \le j < b \\ p_j & b \le j \le n \end{cases} \qquad (A.17)$$

so that π_j represents the limiting probability that there are j customers (calls) in the system at any time. By combining equations (A.5), (A.13), and (A.16) the following result is obtained:

$$\pi_j = \begin{cases} \frac{np_0}{(n-j)} \left[1 + \frac{1}{(n-j-1)!} \sum_{k=0}^{j-1} \left(\frac{\lambda}{\mu}\right)^{k+1} (n-j+k)!\right] & 1 \le j < b \\ \frac{np_0}{(n-j)!} \sum_{k=j-b}^{j-1} \left(\frac{\lambda}{\mu}\right)^{k+1} (n-j+k)! & b \le j \le n \end{cases} \qquad (A.18)$$

. However, there are only n equations, but $n+1$ unknowns. In order to solve for the π_j an additional equation is required. By recognizing the fact that all probabilities must sum to 1.0, the additional equation,

$$\sum_{j=0}^{n} \pi_j = 1 \qquad (A.19)$$

may be used. The limiting distribution for the Markov chain may now be solved by combining equations (A.18) and (A.19):

$$\pi_0 = \left[1 + \sum_{j=1}^{b-1} \frac{n}{(n-j)} \left(1 + \frac{1}{(n-j-1)!} \sum_{k=0}^{j-1} \left(\frac{\lambda}{\mu}\right)^{k+1} (n-j+k)!\right) + \right.$$
$$\left. \sum_{j=b}^{n} \frac{n}{(n-j)} \sum_{k=j-b}^{j-1} \left(\frac{\lambda}{\mu}\right)^{k+1} (n-j+k)!\right]^{-1} \qquad (A.20)$$

The expected number of calls in the system is simply

$$N = \sum_{j=1}^{n} j\pi_j \qquad (A.21)$$

. Still further, the average call arrival rate, Λ, may be calculated

$$\Lambda = \mu\rho \qquad (A.22)$$

. Finally, by Little's law, $N=\Lambda T$, the mean time spent in the system, T, can be found. The mean delay, T_Q, is simply the mean time spent in the system minus the mean service time which is given by $1/\mu$. Thus, the mean delay is given by

$$T_Q = \frac{N}{\Lambda} - \frac{1}{\mu} \qquad (A.23)$$

or

$$T_Q = \frac{N - \rho}{\mu\rho} \qquad (A.24)$$
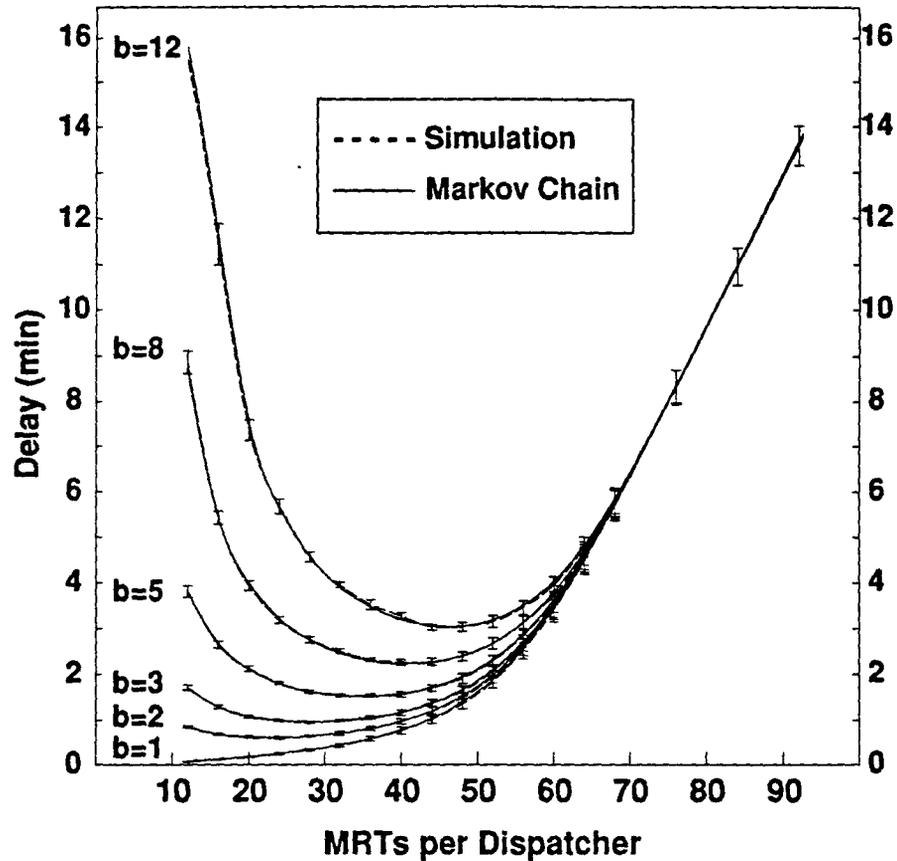
Figure 32 Comparison of OB Simulation Model Results with Markov Chain Model

For $b=1$, note that the result is identical to an M/M/1/n queuing system. Figure 32 shows a comparison of the results of the delay derived from the Markov chain, and that determined empirically through simulation modelling when signalling overhead is omitted, where λ and μ have been set to .001 and .05 respectively, in accordance with Table 2. The error bars on the simulation model curves represent 95% confidence intervals. Slight aberrations in the curves are largely due to limitations of the graph plotting software used. It is clear, however, that the simulation model corresponds almost identically with the theoretical model.

A.2 Closed Batch MRS Subnet Model

A Markov chain representation of a single spot-beam CB subnet without signalling overhead may also be constructed. Figure 33 show one possible representation. Each state in this model has two associated subscripts; a state S_{ij} is defined such that i is the number of call requests in the system, and j is the number of calls requests eligible for service. A call is eligible for service if it is belongs to a completely formed batch. A closed form solution for this system is not given; however, numerical solutions may be determined.
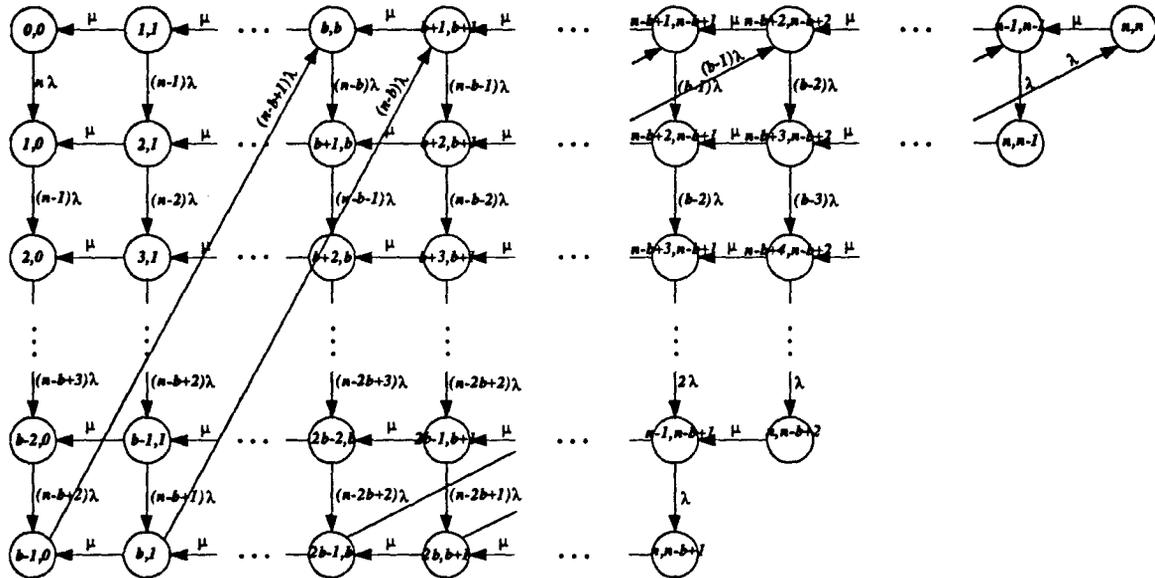
Figure 33 Closed Batch, Single Dispatcher Markov Chain

First of all, recognize that the Markov chain of Figure 33 is a finite state irreducible aperiodic chain and, therefore, has a unique limiting distribution. Thus, the Chapman-Kolmogorov differential equations describing the rates of state transitions may be reduced to a set of balance equations. Let p_{ij} be the limiting probability that the chain is in state S_{ij},

and assume $b \geq 2$, then for $0 \leq k \leq n - b - 1$, there are the following set of equations:

$$-n\lambda p_{0,0} + \mu p_{1,1} = 0 \tag{A.25}$$

$$n\lambda p_{0,0} - (n-1)\lambda p_{1,0} + \mu p_{2,1} = 0 \tag{A.26}$$

$$-[(n-1)\lambda + \mu]p_{1,1} + \mu p_{2,2} = 0 \tag{A.27}$$

$$\ldots$$

$$(n-k-b+1)\lambda p_{k+b-1,k} - [(n-k-b)\lambda + \mu]p_{k+b,k+1} + \mu p_{b+k+1,k+2} = 0 \tag{A.28}$$

$$(n-k-b+1)\lambda p_{k+b-1,k+1} - [(n-k-b)\lambda + \mu]p_{k+b,k+2} + \mu p_{b+k+1,k+3} = 0 \tag{A.29}$$

$$\ldots$$

$$(n-k-b+1)\lambda p_{k+b-1,k+b-2} - [(n-k-b)\lambda + \mu]p_{k+b,k+b-1} + \mu p_{b+k+1,k+b} = 0 \tag{A.30}$$

$$(n-k-b+1)\lambda p_{b+k-1,k-1} - [(n-k-b)\lambda + \mu]p_{b+k,b+k} + \mu p_{b+k+1,b+k+1} = 0 \tag{A.31}$$

$$\ldots$$

$$\lambda p_{n-1,n-2} - \mu p_{n,n-1} = 0 \tag{A.32}$$

$$\lambda p_{n-1,n-b} - \mu p_{n,n} = 0 \tag{A.33}$$

If m is the total number of states, then there are m equations and m unknowns. In matrix notation,

$$\mathbf{Q}\mathbf{p} = 0 \tag{A.34}$$

98

where Q is an $m \times m$ infinitesimal generator matrix, and p is an $m \times 1$ matrix of unknown state probabilities, and where

$$m = b(n - b + 2) + \frac{1}{2} \sum_{j-0}^{b-2} (j + 1)(j + 2) \qquad (A.35)$$

However, the first of the balance equations, (A.25), is linearly dependent on the others. Therefore it is replaced with

$$\sum_{all\ i,j} p_{i,j} = 1 \qquad (A.36)$$

which holds true since all probabilities sum to unity. The resulting system of linear equations is

$$\acute{Q}p = e_1 \qquad (A.37)$$

where e_1 is a zero vector with the exception of a 1.0 entry in the first position. This system of equations is solved using standard numerical methods employing Gaussian elimination and back substitution with row and column pivoting for numerical stability. Then, in order to determine the limiting probability distribution, note that

$$\pi_j = \sum_k p_{j,k} \qquad k = \begin{cases} 0, 1, 2, ..., j & 0 \le j < b \\ j - b + 1, j - b + 2, ..., j & b \le j \le n \end{cases} \qquad (A.38)$$

The utilization factor is calculated as follows

$$\rho = 1 - \sum_{i=0}^{b-1} p_{i,0} \qquad (A.39)$$

Finally, the delay is calculated similarly to that of OB systems yielding

$$T_Q = \frac{N - \rho}{\mu \rho} \qquad (A.40)$$

Once again, for $b=1$, note that the system is identically an M/M/1/n queuing system. The results of the delay derived numerically from the Markov chain model are compared to the results of the simulation model in Figure 34. The individual MRT arrival rate and dispatcher service rate are set to .001 and .05 respectively, in accordance with Table 2. The error bars on the simulation model curves represent 95% confidence intervals. Once again, the simulation model performs almost identically to the theoretical model.
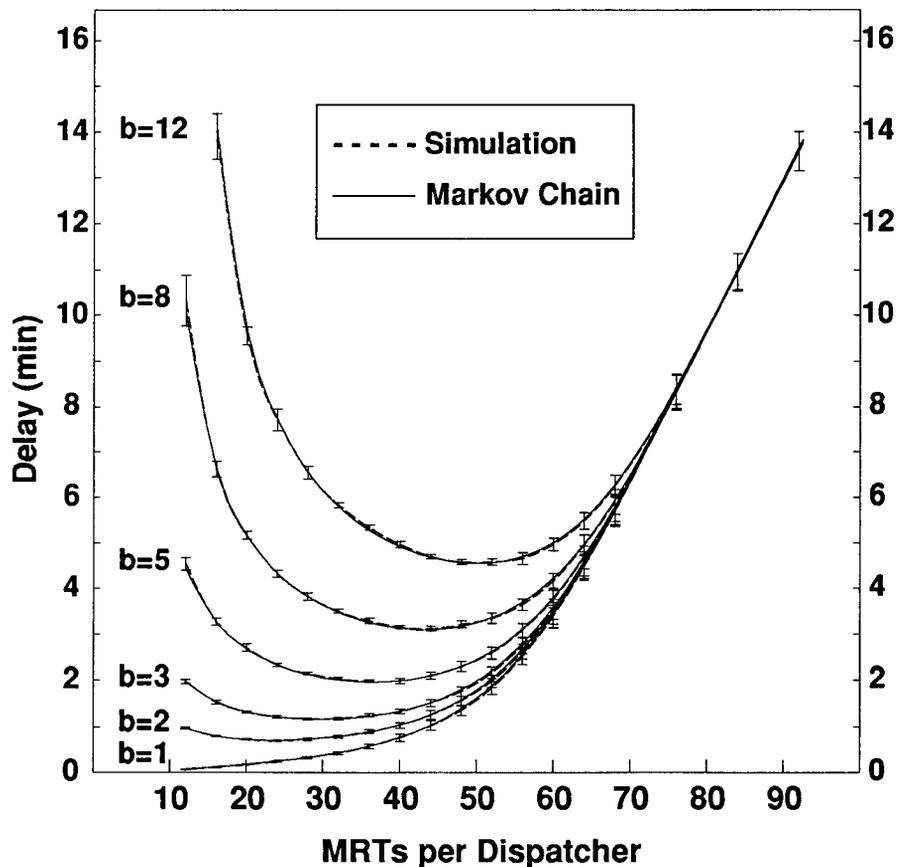
Figure 34 Comparison of OB Simulation Model Results with Markov Chain Model

A.3 Integrated Network Model

The integrated network model consists of both dispatch radio and telephone traffic sources. The radio traffic is made up of a collection of either CB or OB MRS subnets which

are constructed identically to the models described above. Hence no further verification for MRS models is given. Mobile telephone traffic, on the other hand, is modeled as Erlang-B source, and verification is presented here.

Figures 35 and 36 compare theoretical results of telephone traffic with simulation model results in the absence of signalling overhead. The theoretical model used is Erlang-B. The Erlang-B model is an infinite population model where the rate remains constant for the duration of the model's lifetime and the rate is directly proportional to the population. Simulation model MTS traffic is varied in simulation runs by varying the traffic arrival rate, and therefore, the independent variable in the comparisons is the MTT population. Figures 35 and 36 show population versus blocking probability respectively and channel utilization factor. Ninety-five percent confidence intervals are included for the simulation models, however, the simulation model and theoretical model curves are virtually superimposed, reflecting the accuracy of the simulation model implementation of the Erlang-B system.

The probability of blocking for the Erlang-B M/m/n n-server loss model is well known. It is given by

$$p_B = \frac{\rho^n}{n! \sum_{k=0}^{n} \frac{\rho^k}{k!}} \tag{A.41}$$

where $\rho = \lambda/\mu$ and λ is the mean call arrival rate, μ is the mean call service rate per channel, and n is the number of channels available [20,50,59]. The mean channel utilization, ρ, may be calculated directly form λ and μ. Once again, the accuracy of the simulation models can be seen by their close adherence to the presented theoretical results.
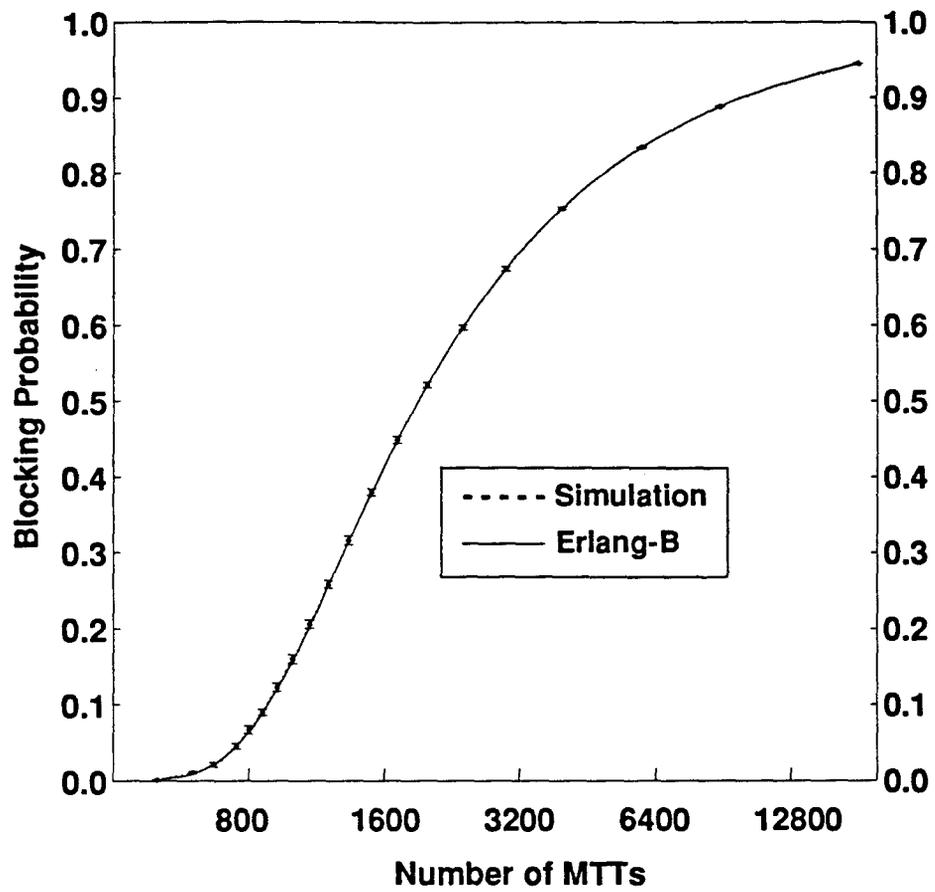
Figure 35 MTS Population versus Blocking Probability
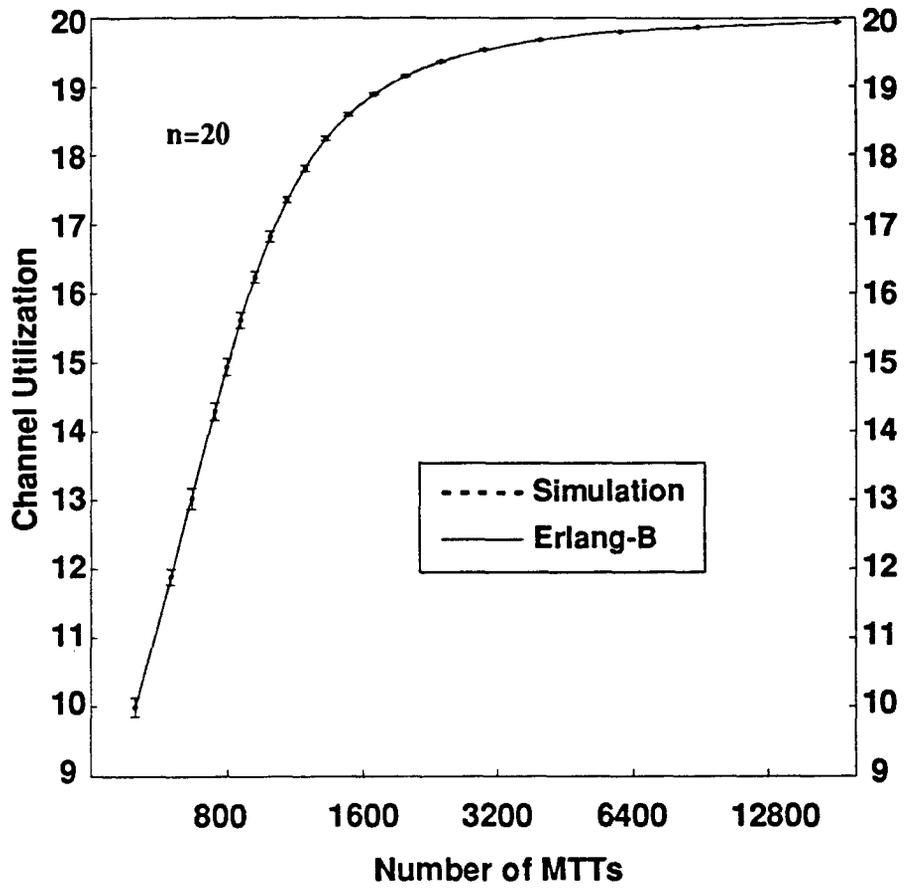
Figure 36 MTS Population versus Utilization Factor

B. Simulation Modelling and Data Analysis

All simulation models are written in the SIMSCRIPT II.5[1] programming language, and executed on Sun SPARC II[2] Workstations under the UNIX[3] operating system. The SIMSCRIPT language is chosen over a general purpose programming language for its built in queuing and scheduling control structures, its statistics gathering capabilities, and its nearly self-documenting source code. Built in features help keep programming models small and thereby reduce coding and debugging time, and enhance program reliability. In further consideration of SIMSCRIPT, it provides a greater degree of flexibility than most other comparable simulation languages and lends itself well to large complicated models [57].

A simulation model consisting of only MRS traffic which is made up of 52 subnets with 44 MRTs per dispatcher each and using a closed batch size of 8, requires over 7 megabytes of computer memory and 3.4 hours of CPU time. In theory, larger model could be run with the given computer resources, but experience has shown that larger models rarely survive to run through to normal completion. In addition, time constraints dictate that much larger models are not practical given the number of test runs that may be made to achieve any degree of statistical reliability over the large number of network configurations tested.

---

[1] SIMSCRIPT is a registered trademark of CACI

[2] SPARC is a registered trademark of Sun Microsystems

[3] UNIX is a registered trademark of AT&T

B.1 Notes on Methodology

In varying the traffic levels for an MRS subnet model, the MRT population is varied on an MRT per dispatcher basis; increments of 4 to 8 MRTs per dispatcher are used. This means that larger increases in absolute population are made when there are a greater number of dispatchers. However, the number of MRTs per dispatcher has been shown to be the more relevant measure of subnet load than absolute population.

When several subnets are combined under the integrated network strategy, the subnet configuration is fixed with 44 MRTs per dispatcher, so each subnet is under a fixed traffic load. In order to make adjustments to MRS traffic within a combined network the number of subnets is varied. When there are only a few subnets, adding or removing one subnet constitutes a fairly large proportional change in overall MRS traffic. When there are many subnets varying the number of subnets makes more gradual proportional changes. While it is desirable to be able to make small proportional changes in order to fine tune network performance, this is not always possible in practical model implementation.

When varying the number of subnets, one must keep in mind that there are practical limits to the number of subnets that may be accommodated by a given number of channels. A model with 35 available channels and enough subnets present to maintain full channel utilization reaches is sufficiently large that it reached the upper limit of given computer resource functionality. Experience has shown that larger models will rarely run though to normal completion.

Traffic variations for the MTS segment are simply made by adjusting the rate at which new calls enter the system. Since the rate is specified by a single real number, very fine traffic adjustments may be made. In addition, the simple manner in which MTS calls are generated

does not pose a computational constraint on the system being modelled; the number of MRS subnets is the limiting factor.

B.2 Confidence Intervals

The results of Appendix A show a strong correlation between the given analytical models and computer simulation results. However, underlying those results is the assumption that the output from several simulation runs is representative of the expected output of all possible runs. One common method used to establish an estimate of the expected behavior is to make several test runs and then calculate a *confidence interval* about the mean value of a particular output parameter. This is the approach taken above.

A confidence interval for a population parameter is an interval calculated from a sample statistic which estimates the population parameter with some degree of confidence. The confidence level is given as the probability that the population parameter falls within the calculated interval [60]. With regard to analyzing simulation model output, however, calculating a confidence interval for an output parameter may not provide the desired result. Computer simulations must often run for an initial period of time before statistics are collected in order to eliminate initialization bias. Several methods have been proposed to control transient effects [61–63]; however, one primary method has been applied to models used here. Successive identical test statistics from consecutive simulation blocks of execution are compared in order to determine how long it takes for transient effects to subside. This determination is done in part by comparisons among the test statistics and in part by comparison of the test statistics to the known analytic results given above. If transient effects are not negated, the confidence intervals that are calculated may contain initialization

bias. All calculations are based on experiments with a single subnet since each subnet in a group behaves similarly.

The degree of confidence may be increased by examining more samples of simulation output; however, the magnitude of the simulation model and the available computer resources constrain the number of test runs that may actually be performed. The number of runs that should be made was determined by selecting the size of the confidence interval. A reasonably small 95% confidence interval could be obtained for the simulation models. Figures 32 and 34 show 95% confidence intervals for single MRS subnet simulation results for delay. These two figures exemplify typical confidence intervals for all similar figures presented above. Ninety-five percent confidence intervals for MTS simulation models are shown in Figures 35 and 36. Only 90% confidence intervals could be achieved for the integrated network. A typical curve with confidence intervals is given below (Figure 37). Figure 37 is a modification of Figure 25 showing only the results of MRS modelling but including 95% confidence intervals for the number of MRTs in the network.
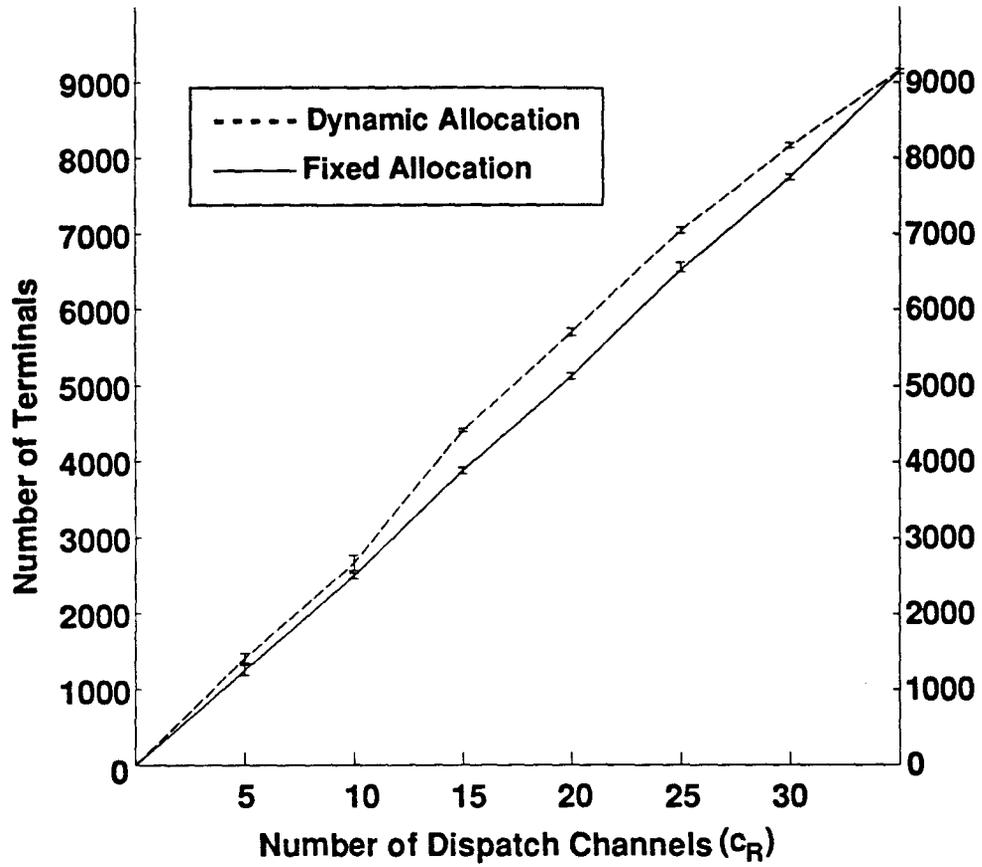
Figure 37 CB Low Performance Network with 95% Confidence Intervals

C. Source Code

The following source code is provided as an example of the software used in creating simulation models; it is the CB MRS subnet model employed above. The model is made up of several parts, each of which is maintained in a its own source file. Source files are compiled separately, then subsequently linked together to form a single executable version. Each of section that follows constitutes a single source file.

C.1 FC.sim

This file constitutes the simulation model preamble.

```
Preamble

    Normally mode is undefined

    Define secs to mean days

    Resources include
        DISPATCHER                      •


    Permanent entities
        Every QUEUE
            has an ARRIVAL.COUNT,
            has a COMPLETION.COUNT,
            has a BATCH.FORMATION.COUNTER,
            has a CURRENT.BATCH,
            may have a PENDING.TIMEOUT,
            owns a CALL.REQUEST.SET
```

```
                Define ARRIVAL.COUNT,
                    BATCH.FORMATION.COUNTER,
                    COMPLETION.COUNT
                    as integer variables
                Define PENDING.TIMEOUT,
                    CURRENT.BATCH
                    as pointer variables
                Define CALL.REQUEST.SET
                    as sets

    Events include
        RESTART,
        FINAL.REPORT


    Processes include
        CHECK.POINT
        Every MRT.CALL
            has a BEAM.ID
            has a CURRENT.MRT

        Every DISPATCH
            has a BEAM.ID,
            has a BTIME

        Every MRT
            has a BEAM.ID

        Define BEAM.ID
            as an integer variable

        Define BTIME
            as a double variable

        Define CURRENT.MRT
            as a pointer variable


    Temporary entities
        Every CALL.REQUEST
            has an INITIATION.TIME,
```

```
            has a HOLDING.TIME,
            has an CALL.ID,
            has a MY.BATCH,
            belongs to a CALL.REQUEST.SET

            Define HOLDING.TIME,
                INITIATION.TIME
                as double variables
            Define CALL.ID,
                MY.BATCH
                as pointer variables


        Every BATCH
            has a NEXT.CALL,
            has a SIZE,
            may have a CURRENT.DISPATCH

            Define NEXT.CALL,
                CURRENT.DISPATCH
                as pointer variables
            Define SIZE
                as an integer variable


    Define NUM.DISPATCHERS,
        MRT.CALL.IAT,
        NUM.SPOT.BEAMS,
        NUM.TIMEOUTS,
        NUM.QUEUED,
        BATCH.SIZE,
        NUM.MRTS.PER.DISP,
        REQUEST.TOTAL,
        CALL.ARRIVAL.STREAM,
        HOLD.TIME.STREAM
        as integer variables

    Define IAT,
        TWB,
        TWD,
        TWS,
```

```
        CHANNEL.SETUP.TIME,
        CHANNEL.TEARDOWN.TIME,
        END.OF.RUN,
        SERVICE.TIME,
        MEAN.HOLDING.TIME,
        RESTART.TIME,
        GLOBAL.DELAY,
        RUN.LENGTH,
        LAST.CALL.TIME,
        CHECK.IN.TIME,
        TIMEOUT.PERIOD
        as double variables

Tally MEAN.GLOBAL.DELAY as the mean,
        GLOBAL.DELAY.SD as the std.dev
        of GLOBAL.DELAY

Tally MEAN.IAT as the mean,
        IAT.SD as the std.dev
        of IAT

Tally MEAN.SERVICE.TIME as the mean,
        SERVICE.TIME.SD as the std.dev
        of SERVICE.TIME

Tally MEAN.TWB as the mean,
        TWB.STD.DEV as the std.dev
        of TWB

Tally MEAN.TWD as the mean,
        TWD.STD.DEV as the std.dev
        of TWD    •

Tally MEAN.TWS as the mean,
        TWS.STD.DEV as the std.dev
        of TWS


Define NUM.IATS to mean 61
Define NUM.DELAYS to mean 31
Define NUM.LENS to mean 26
```

```
        Define MRT.CALL.RATE to mean 0.001
        Define CNTL.MSG.SER.DELAY to mean 0.050
        Define PROPAGATION.DELAY to mean 0.250

        Define .ACTIVE to mean 0
        Define .ASLEEP to mean 1
        Define .WAITING to mean 2

        Define INI.FILE to mean Unit 1
        Define OUT.FILE to mean Unit 2
        Define STDOUT to mean Unit 6
        Define STDERR to mean Unit 98
End
```

C.2 call.sim

This file constitutes a process of an individual MRT call.

```
Process MRT.CALL
     Given BEAM.ID and CURRENT.MRT
     Define BEAM.ID as an integer variable
     Define CURRENT.MRT as a pointer variable

     Define .DELAY as a double variable
     Define .CALL.REQ as a pointer variable

     Create a CALL.REQUEST called .CALL.REQ
     Let CALL.ID(.CALL.REQ) = MRT.CALL
     Let IAT = time.v - LAST.CALL.TIME
     Let LAST.CALL.TIME = time.v
     Let INITIATION.TIME(.CALL.REQ) = time.v
     File .CALL.REQ in CALL.REQUEST.SET(BEAM.ID)
     Add 1 to NUM.QUEUED


     Add 1 to ARRIVAL.COUNT(BEAM.ID)

     Add 1 to BATCH.FORMATION.COUNTER(BEAM.ID)
     If BATCH.FORMATION.COUNTER(BEAM.ID) = 1
         Create a BATCH called CURRENT.BATCH(BEAM.ID)
         NEXT.CALL(CURRENT.BATCH(BEAM.ID)) = .CALL.REQ
     Endif
     If BATCH.FORMATION.COUNTER(BEAM.ID) = BATCH.SIZE
         SIZE(CURRENT.BATCH(BEAM.ID)) = BATCH.SIZE
      ;  Activate a DISPATCH
             called CURRENT.DISPATCH(CURRENT.BATCH(BEAM.ID))
             giving CURRENT.BATCH(BEAM.ID) and time.v
             now
         BATCH.FORMATION.COUNTER(BEAM.ID) = 0
     Endif

     MY.BATCH(.CALL.REQ) = CURRENT.BATCH(BEAM.ID)

     Suspend
```

```
      .DELAY = time.v - INITIATION.TIME(.CALL.REQ)
      Remove .CALL.REQ from CALL.REQUEST.SET(BEAM.ID)
      Subtract 1 from NUM.QUEUED
      Let HOLDING.TIME(.CALL.REQ) =
          exponential.f(MEAN.HOLDING.TIME,HOLD.TIME.STREAM)
      Work HOLDING.TIME(.CALL.REQ) secs
      GLOBAL.DELAY = .DELAY
      Let SERVICE.TIME = HOLDING.TIME(.CALL.REQ)

      Reactivate the DISPATCH
          called CURRENT.DISPATCH(MY.BATCH(.CALL.REQ)) now
      Destroy the CALL.REQUEST called .CALL.REQ
      Add 1 to COMPLETION.COUNT(BEAM.ID)
      Reactivate the MRT called CURRENT.MRT now
End
```

C.3 checkin.sim

This file constitutes a process for giving positive feedback of program execution at specified constant simulated time intervals.

```
Process CHECK.POINT

    Define .CP as a pointer variable

    Use STDERR for output

    Print 1 line with
        time.v thus
CHECK POINT: time = **********
    Activate a CHECK.POINT called .CP in CHECK.IN.TIME secs
End
```

C.4 disp.sim

This file constitutes a process of an MRS subnet dispatcher.

```
Process DISPATCH
    Given CALL.BATCH and BTIME

    Define CALL.BATCH as a pointer variable
    Define BTIME as a double variable

    Define .TWD as a double variable
    Define .TWS as a double variable

    Define I as an integer variable

    Let .TWD = time.v
    Request 1 DISPATCHER(1)
    Let TWD = time.v - .TWD

    If CHANNEL.SETUP.TIME > 0.0
        Work CHANNEL.SETUP.TIME secs
    Endif

    Let .TWS = time.v
    For I = 1 to SIZE(CALL.BATCH) do
        Let TWS = time.v - .TWS
        Let TWB = BTIME -
            INITIATION.TIME(NEXT.CALL(CALL.BATCH))
        Reactivate the MRT.CALL
            called CALL.ID(NEXT.CALL(CALL.BATCH))
            now
        NEXT.CALL(CALL.BATCH) =
            S.CALL.REQUEST.SET(NEXT.CALL(CALL.BATCH))
        Suspend
    Loop

    If CHANNEL.TEARDOWN.TIME > 0.0
        Work CHANNEL.TEARDOWN.TIME secs
    Endif
```

```
    Relinquish 1 DISPATCHER(1)
    Destroy the BATCH called CALL.BATCH
End
```

C.5 init.sim

This file constitutes a subroutine for initializing simulation variables and entities.

```
Routine INITIALIZE
    Define I as an integer variable

    Create every QUEUE(NUM.SPOT.BEAMS)

    Create every DISPATCHER(1)
    Let U.DISPATCHER = NUM.DISPATCHERS

    For I = 1 to NUM.SPOT.BEAMS do
        BATCH.FORMATION.COUNTER = 0
    Loop

    Let NUM.QUEUED = 0
    Let LAST.CALL.TIME = 0.0
    Let MRT.CALL.IAT = 1.0 / MRT.CALL.RATE

    Let END.OF.RUN = RUN.LENGTH + RESTART.TIME

    If CHANNEL.SETUP.TIME = -1.0
        Let CHANNEL.SETUP.TIME =
            (BATCH.SIZE + 2) * CNTL.MSG.SER.DELAY +
            3 * PROPAGATION.DELAY
    Endif

    If CHANNEL.TEARDOWN.TIME = -1.0
        Let CHANNEL.TEARDOWN.TIME =
            CNTL.MSG.SER.DELAY + PROPAGATION.DELAY
    Endif
End
```

C.6 main.sim

This file constitutes the main procedure of the simulation model.

```
Main
    Define I and J as integer variables

    Define .NUM.MRTS.PER.BEAM as an integer variable

    Call GET.PARAMETERS
    Call INITIALIZE

    Activate a CHECK.POINT now
    Schedule a RESTART in RESTART.TIME secs
    Schedule a FINAL.REPORT in RESTART.TIME + RUN.LENGTH secs

    Let .NUM.MRTS.PER.BEAM =
        (NUM.DISPATCHERS * NUM.MRTS.PER.DISP) / NUM.SPOT.BEAMS
    For I = 1 to NUM.SPOT.BEAMS do
        For J = 1 to .NUM.MRTS.PER.BEAM do
            Activate a MRT giving I now
        Loop
    Loop
    Start simulation
End
```

C.7 mrt.sim

This file constitutes the process of a single MRT.

```
Process MRT given BEAM.ID

    Define BEAM.ID as an integer variable

        Here
        Wait exponential.f(MRT.CALL.IAT,CALL.ARRIVAL.STREAM)
            secs
        Activate a MRT.CALL giving BEAM.ID and MRT now
        Suspend
        Jump back
End
```

C.8 read.sim

This file constitutes a subroutine which reads and initialization file to obtain externally specified simulation parameters.

```
Routine GET.PARAMETERS

    Define .QUIT as an integer variable
    Define .FIELD as a real variable
    Define .NAME as a text variable

    Let CALL.ARRIVAL.STREAM = 1
    Let HOLD.TIME.STREAM = 3
    Let RESTART.TIME = 0.0
    Let BATCH.SIZE = 1
    Let CHANNEL.SETUP.TIME = -1.0
    Let CHANNEL.TEARDOWN.TIME = -1.0
    Let MEAN.HOLDING.TIME = 20.0

    Let NUM.MRTS.PER.DISP = -1
    Let NUM.DISPATCHERS = -1
    Let NUM.SPOT.BEAMS = -1
    Let TIMEOUT.PERIOD = -1.0
    Let RUN.LENGTH = -1.0
    Let CHECK.IN.TIME = -1.0

    Open INI.FILE for input, name = "SIM.ini"
    Use INI.FILE for input

    .QUIT = 0
    eof.v = 1
    Read .NAME
    While eof.v <> 2 do
        Read .FIELD

        If .NAME = "num.dispatchers"
            Let NUM.DISPATCHERS = int.f(.FIELD)
        Else if .NAME = "call.arrival.stream"
            Let CALL.ARRIVAL.STREAM = int.f(.FIELD)
        Else if .NAME = "hold.time.stream"
```

```
            Let HOLD.TIME.STREAM = int.f(.FIELD)
        Else if .NAME = "check.in.time"
            let CHECK.IN.TIME = .FIELD
        Else if .NAME = "restart.time"
            Let RESTART.TIME = .FIELD
        Else if .NAME = "num.spot.beams"
            Let NUM.SPOT.BEAMS = int.f(.FIELD)
        Else if .NAME = "mean.holding.time"
            Let MEAN.HOLDING.TIME = .FIELD
        Else if .NAME = "batch.size"
            Let BATCH.SIZE = int.f(.FIELD)
        Else if .NAME = "timeout.period"
            Let TIMEOUT.PERIOD = .FIELD
        Else if .NAME = "run.length"
            Let RUN.LENGTH = .FIELD
        Else if .NAME = "channel.setup.time"
            Let CHANNEL.SETUP.TIME = .FIELD
        Else if .NAME = "channel.teardown.time"
            Let CHANNEL.TEARDOWN.TIME = .FIELD
        Else if .NAME = "num.mrts.per.disp"
            Let NUM.MRTS.PER.DISP = .FIELD
        Else
            Use STDERR for output .
            Print 1 line with .NAME like this
Unknown parameter: ************************
            .QUIT = 1
        Endif endif endif endif endif endif endif endif endif
            endif endif endif endif

    Start new input record
    If eof.v <> 2
        Read .NAME
    Endif
    Loop

    If NUM.MRTS.PER.DISP = -1
        Print 1 line thus
        num.mrts.per.disp not specified
        .QUIT = 1
    Endif
    If NUM.DISPATCHERS = -1
```

```
            Print 1 line thus
            num.dispatchers not specified
            .QUIT = 1
        Endif
        If NUM.SPOT.BEAMS = -1
            Print 1 line thus
            num.spot.beams not specified
            .QUIT = 1
        Endif
        If TIMEOUT.PERIOD = -1.0
            Print 1 line thus
            timeout.period not specified
            .QUIT = 1
        Endif
        If RUN.LENGTH = -1.0
            Print 1 line thus
            run.length not specified
            .QUIT = 1
        Endif
        If CHECK.IN.TIME = -1.0
            Print 1 line thus
            check.in.time not specified
            .QUIT = 1
        Endif

        If .QUIT = 1
            Stop
        Endif

End
```

C.9 report.sim

This file constitutes and event to print a summary of simulation execution, including test statistics.

```
Event FINAL.REPORT

    Use STDOUT for output

    Print 14 lines with
        RESTART.TIME,
        CHECK.IN.TIME,
        NUM.DISPATCHERS,
        NUM.SPOT.BEAMS,
        CHANNEL.SETUP.TIME,
        CHANNEL.TEARDOWN.TIME,
        MEAN.HOLDING.TIME,
        BATCH.SIZE,
        MRT.CALL.RATE,
        NUM.MRTS.PER.DISP,
        CNTL.MSG.SER.DELAY * 1000,
        PROPAGATION.DELAY * 1000,
        RUN.LENGTH like this
Length of time before resetting statistical
variables = ****.***
Length of time between checkpoints = **********.*
Number of dispatchers = ***
Number of spot beams =  ***
Length of time to set up channel      = **.**** secs
Length of time to tear down a channel = **.**** secs
Mean holding time for an MRT call = ***.** secs
BATCH SIZE = ***
MRT call rate  = **.***** calls per second
Number of MRTs per disp = *****
Contol message serializing delay = ****.**
Propagation.delay = ****.**
Length of run = **************.* secs

print 5 lines with MEAN.SERVICE.TIME,
        MEAN.GLOBAL.DELAY,
```

```
          MEAN.TWB, MEAN.TWD, MEAN.TWS thus
MEAN SERVICE TIME: ****.***          MEAN DELAY: *****.***
TWB: ****.***      TWD: ****.***     TWS: ****.***



     Stop
End
```

C.10 reset.sim

This file constitutes an event which resets variables used for gathering statistics. The event is scheduled to occur when the model reaches a state of equilibrium.

```
Event RESTART
    Define I as an integer variable

    Reset totals of
        IAT,
        SERVICE.TIME,
        GLOBAL.DELAY



    For I = 1 to NUM.SPOT.BEAMS do
        Let ARRIVAL.COUNT(I) = 0
        Let COMPLETION.COUNT(I) = 0
    Loop

    NUM.TIMEOUTS = 0
End
```

C.11 timeout.sim

This file constitutes an event which may be used to schedule a time-out for batch formation. It has not been used in the above analysis.

```
Event TIMEOUT
    Given BEAM.ID and PART.BATCH
    Define BEAM.ID as an integer variable
    Define PART.BATCH as a pointer variable

    Add 1 to NUM.TIMEOUTS
    SIZE(PART.BATCH) = BATCH.FORMATION.COUNTER(BEAM.ID)
    BATCH.FORMATION.COUNTER(BEAM.ID) = 0
    Activate a DISPATCH
        called CURRENT.DISPATCH(PART.BATCH)
        giving PART.BATCH
        now
End
```