ANALYSIS AND SIMULATION OF COMPUTER CONTROLLED CELLULAR MOBILE RADIO SYSTEMS

by

SAAD HAJ BAKRY

A thesis submitted to the
University of Aston in Birmingham
for the degree of
Doctor of Philosophy

April 1980

ANALYSIS AND SIMULATION OF COMPUTER CONTROLLED CELLULAR MOBILE RADIO SYSTEMS

by

SAAD HAJ BAKRY

A thesis submitted to the
University of Aston in Birmingham
for the degree of
Doctor of Philosophy

April 1980

ANALYSIS AND SIMULATION OF COMPUTER CONTROLLED

CELLULAR MOBILE RADIO SYSTEMS

by

Saad Haj Bakry

A thesis submitted to the
University of Aston in Birmingham
for the degree of
Doctor of Philosophy 1980

Summary

Cellular mobile radio systems will be of increasing importance in the future. This thesis describes research work concerned with the teletraffic capacity and the computer control requirements of such systems. The work involves theoretical analysis and experimental investigations using digital computer simulation.

New formulas are derived for the congestion in single-cell systems in which there are both land-to-mobile and mobile-to-mobile calls and in which mobile-to-mobile calls go via the base station. Two approaches are used, the first yields modified forms of the familiar Erlang and Engset formulas, while the second gives more complicated but more accurate formulas. The results of computer simulations to establish the accuracy of the formulas are described. New teletraffic formulas are also derived for the congestion in multi-cell systems. Fixed, dynamic and hybrid channel assignments are considered. The formulas agree with previously published simulation results.

Simulation programs are described for the evaluation of the speech traffic of mobiles and for the investigation of a possible computer network for the control of the speech traffic. The programs were developed according to the structured programming approach leading to programs of modular construction. Two simulation methods are used for the speech traffic: the roulette method and the time-true method. The first is economical but has some restriction, while the second is expensive but gives comprehensive answers. The proposed control network operates at three hierarchical levels performing various control functions which include: the setting-up and clearing-down of calls, the hand-over of calls between cells and the address-changing of mobiles travelling between cities. The results demonstrate the feasibility of the control network and indicate that small mini-computers inter-connected via voice grade data channels would be capable of providing satisfactory control performance.

Key Words

MOBILE RADIO SYSTEMS, TELETRAFFIC ENGINEERING, COMPUTER NETWORKS, COMPUTER SIMULATION, STRUCTURED PROGRAMMING.

ACKNOWLEDGEMENTS

The author wishes to acknowledge with much gratitude and appreciation Dr. M. H. Ackroyd, for the invaluable guidance, helpful suggestions, encouragement and support throughout the duration of the research work presented in this thesis.

Special thanks are due to Mr. D. Bear for his useful comments on the teletraffic analysis part of the work and to Professor J. E. Flood for constant encouragement.

A debt of gratitude is owed to Mrs. C. Johns, for reading the manuscript and to Miss N. P. Freeman and Mrs. H. M. Turner for typing the thesis.

Finally, the author would like to thank all of his colleagues in the telecommunication group for providing the friendly and encouraging environment in which this research was performed.

LIST OF CONTENTS

	Page No.
TITLE PAGE	i
SUMMARY	ii
ACKNOWLEDGEMENTS	iii
LIST OF CONTENTS	iv
LIST OF FIGURES	vi
LIST OF TABLES	xi
CHAPTER ONE - INTRODUCTION	
1.1 LARGE SCALE MOBILE RADIO SYSTEMS	1
1.2 TELETRAFFIC AND CONTROL PROBLEMS	6
1.3 INVESTIGATION METHODS	10
1.4 THE PRESENT RESEARCH	11
CHAPTER TWO - TELETRAFFIC ANALYSIS	
2.1 TELETRAFFIC MODELS	15
2.2 SINGLE-CELL SYSTEMS	17
2.2.1 Simple Analysis	17
2.2.2 Alternative Approach	26
2.2.3 Comparison with Simulation Results	32
2.3 MULTI-CELL SYSTEMS	37
2.3.1 Fixed Channel Assignment	37
2.3.2 Dynamic Channel Assignment	37
2.3.3 Hybrid Channel Assignment	44
2.3.4 Comparison with Other Results	46
CHAPTER THREE - SIMULATION TOOLS	
3.1 STRUCTURED PROGRAMMING FOR SIMULATION	52
3.2 SIMULATION TOOLS FOR SPEECH TRAFFIC	55

		Page No.
3.2.1	Time-True Simulation	55
3.2.2	Roulette Simulation	66
3.2.3	Roulette Versus Time-True Simulations	74
3.3	SIMULATION TOOLS FOR COMPUTER NETWORKS	
3.3.1	Design of Computer Networks	80
3.3.2	Models for Computer Networks	81
3.3.3	Simultion of Computer Networks	85
CHAPTER FOUR - COMPUTER CONTROL		
4.1	CONTROL REQUIREMENTS	95
4.2	A PROPOSED CONTROL NETWORK	98
4.3	DATA TRAFFIC FOR CONTROL REQUIREMENTS	102
4.4	SIMULATION OF PROPOSED NETWORK	115
CHAPTER FIVE - CONCLUSIONS AND FUTURE RESEARCH		
5.1	CONCLUSIONS	146
5.2	FUTURE RESEARCH	150
APPENDICES		
APPENDIX I	COMPUTING TELETRAFFIC FORMULAS	153
APPENDIX II	SIMULATION PROGRAMS	174
REFERENCES		193

LIST OF FIGURES

FIGURE	CAPTION	Page No.
1.1	The lay-out of a cellular system with a buffer of one-cell wide belt.	4
1.2	The lay-out of a cellular system with a buffer of two-cell wide belt.	5
2.1	Theoretical and simulation results with infinite traffic source and 20 channels. Equal proportions of type 1, type 2 and type 3 traffic.	34
2.2	Theoretical and simulation results with 200 mobiles and 20 channels. Equal proportions of type 1, type 2 and type 3 traffic.	35
2.3	Comparison between theoretical predicted congestion and simulation results at moderate congestion values. Basic load = 5 Erlangs per cell; 10 channels per cell (fixed channel assignment); 8 fixed, 2 dynamic channels per cell (hybrid channel assignment).	48
2.4	Comparison between theoretical predicted congestion and simulation results at low congestion values. Basic load = 11.4 Erlangs per cell; 18 channel per cell (fixed channel assignment); 14 fixed, 4 dynamic channels per cell (hybrid channel assignment).	50

FIGURE	CAPTION	Page No.
3.1	Programming simulation in a top-down structured programming approach.	54
3.2	Levels of mobile radio system description for time-true simulation.	58
3.3	Levels of mobile radio system state for time-true simulation.	59
3.4	A mobile radio system description in Algol 68 for time-true simulation	61
3.5	An Algol 68 procedure for generating exponentially distributed random numbers.	63
3.6	Examples of four Algol 68 procedures at low levels for the time-true simulation of a mobile radio system.	64
3.7	An Algol 68 procedure at the highest level for the time-true simulation of a mobile-radio system.	65
3.8	Levels of mobile radio system description for roulette simulation.	71
3.9	Levels of mobile radio system state for roulette simulation.	71
3.10	A mobile radio system description in Coral 66 for roulette simulation.	73
3.11	A Coral 66 procedure for generating random numbers of a specified range using the hardware random number generator.	73

FIGURE	CAPTION	Page No.
3.12	Examples of two Coral 66 procedures at different levels for the roulette simulation of a mobile radio system.	75
3.13	Efficiency of time-true simulation versus traffic offered and congestion for a system with 200 mobiles and 20 channels.	78
3.14	Efficiency of roulette simulation versus traffic offered and congestion for a system with 20 channels.	79
3.15	Schematic diagram of a packet switching computer.	84
3.16	Levels of computer network description.	87
3.17	Levels of computer network state.	88
3.18	Computer network description in Algol 68.	90
3.19	Examples of four Algol 68 procedures operating at the lowest level for the simulation of a computer network.	92
3.20	An Algol 68 procedure operating at the highest level for the simulation of a computer network.	93
4.1	The structure of a controlling network for cellular mobile radio.	99
4.2	The structure of a possible connectivity between city nodes.	101
4.3	The structure of a 'cell network'.	117
4.4	The structure of a 'system unit network'.	118

FIGURE	CAPTION	Page No.
4.5	The structure of an 'inter-city network'.	120
4.6	Channel throughput versus channel traffic in a cell network.	124
4.7	Average packet delay versus channel traffic in a cell network.	125
4.8	Efficiency of transmission versus channel traffic in a cell network.	126
4.9	Maximum length of queues per system-unit node versus traffic rate in a system unit network.	128
4.10	Average packet delay versus traffic rate in a system-unit network.	129
4.11	Average node throughput versus traffic rate for two different routing strategies in an inter-city network.	133
4.12	Average channel usage versus traffic rate for two different routing strategies in an inter-city network.	134
4.13	Average packet delay versus traffic rate for two different routing strategies in an inter-city network.	135
4.14	Maximum length of queues per node versus traffic rate for two different traffic streams in an inter-city network.	136
4.15	Call set-up delay in a single-cell system.	140

FIGURE	CAPTION	Page No.
4.16	Call set-up delay in a multi-cell system.	141
4.17	Hand-over delay.	142
4.18	Address-changing delay.	143
I.1	The lay-out of a cellular system with a central cell and 5 circles of cells around it.	155
I.2	Computing $\frac{A^C}{C!} \Big/ \sum_{x=0}^{C} \frac{A^x}{x!}$	159
I.3	Computing $\frac{\frac{M!}{(M-C)!} \frac{a^C}{C!}}{\sum_{x=0}^{C} \frac{M!}{(M-x)!} \frac{a^x}{x!}}$	160
I.4	Computing $\sum_{x=0}^{C/2} \frac{(A_1+A_2)^{2x}}{(2x)!} \cdot \frac{(A_3/2)^{(\frac{C}{2}-x)}}{(\frac{C}{2}-x)!}$	161
I.5	Computing $\sum_{x=0}^{\frac{C}{2}-1} \frac{(A_1+A_2)^{(2x+1)}}{(2x+1)!} \cdot \frac{(A_3/2)^{(\frac{C}{2}-x-1)}}{(\frac{C}{2}-x-1)!}$	162
I.6	Computing $\sum_{x=0}^{C} \sum_{y=0}^{<(C-x)/2>} \frac{(A_1+A_2)^x}{x!} \cdot \frac{(A_3/2)^y}{y!}$	163
I.7	Computing $F(state) = \sum_{W=state}^{C} \frac{C!}{W!(C-W)!} V^{C-W}(1-V)^W$	164

LIST OF TABLES

TABLE	CAPTION	Page No.
3.1	The ranges of a random number generator for the roulette simulation of a mobile radio system with an infinite number of mobiles.	69
3.2	The ranges of a random number generator for the roulette simulation of a mobile radio system with a finite number of mobiles.	72
4.1	Rate of packet transmission per mobile in a single-cell mobile radio system.	106
4.2	Rate of packet transmission per mobile for the various basic control functions.	109
4.3	Rate of packet transmission per mobile for the hand-over functions.	114
4.4	Rate of packet transmission per mobile for the address-changing functions.	115
I.1	Traffic-congestion table for a single-cell system with an infinite number of mobiles and equal proportions of type 1, type 2 and type 3 traffic.	169
I.2	Traffic-congestion table for a single-cell system with 200 mobiles and equal proportions of type 1, type 2 and type 3 traffic.	170

TABLE	CAPTION	Page No.
I.3	Traffic-congestion table for a single-cell system with an infinite number of mobiles and different proportions of mobile-to-mobile calls.	171
I.4	Traffic-congestion table for a multi-cell system using fixed or dynamic channel assignment. The frequency reuse buffering area is 6 cells. The proportion of mobile-to-mobile calls within a single cell is 0.01.	172
I.5	Traffic-congestion table for a multi-cell system using fixed or hybrid channel assignment. The frequency reuse buffering area is 6 cells. The proportion of mobile-to-mobile calls within a single cell is 0.01.	173

CHAPTER ONE

INTRODUCTION

1. INTRODUCTION

1.1 LARGE SCALE MOBILE RADIO SYSTEMS

The first appearance of mobile radio came soon after the invention of radio at the turn of the century. Over the years, demands for the use of mobile radio have been increasing. This increase has been influenced by the better utilization of vehicles which mobile radio offers. In the past, technology has only been able to meet demands for mobile radio on a small scale giving priority to public service vehicles such as ambulances, police cars and fire appliances. With the present rapid advance of technology, mobile radio telecommunications on a larger scale are becoming feasible.

The objective of large scale mobile radio systems is to enable anyone on the move to communication instantly and effectively with anyone else. A user of such a system would be able to make and receive calls from his vehicle as if he were in his office or at home. The mobile system would be an extension of the normal telephone network. Such large scale systems are of great future importance in the international market. At present advanced nations are actively involved in this area of research and also several developing nations are interested in its practical application[1-7].

The long-standing problem in mobile radio engineering is the lack of frequency channels available for mobile use. Early solutions to this problem involved providing more frequency channels at higher ranges of the frequency spectrum[8], as well as the development of methods for reducing the bandwidth of these channels. Although these solutions increase the capacity of mobile radio systems, the increase is limited, such that large scale systems are impracticable. The cellular approach for the design of large scale mobile radio systems is being observed as a promising practical method. For this reason, the term *cellular* systems in mobile radio engineering is synonymous with the term *large scale* or *high capacity* systems[8-13].

The cellular approach is a management scheme for providing highly effective utilization of the frequency channels available for mobile use. The central concept of this scheme is that the area in which the mobile system is to operate is divided into small zones or cells, each with its own base station. Such cells would normally be a few miles wide[10]. Frequency channels used in one cell may then be used simultaneously in others. This is made possible by ensuring that there is sufficient distance between cells using a particular frequency channel simultaneously so that interference is insignificant.

The true shape of these cells is likely to be highly irregular, as it is determined by the geography of the region and by the man-made structure present. However, a convenient ideal shape is required for planning the general lay-out of the cellular systems and the distribution of the base stations. The three shapes which can cover a plane area with no gaps or overlaps, have been considered[1,9,14] These are the equilateral triangle, the square and the hexagon. Of these three shapes, the hexagon has the smallest number of adjacent base stations. In addition, the hexagon most nearly approaches the circular-shaped pattern that can be most easily produced by the base station antenna. For these reasons, the hexagon is considered as an ideal shape for cells.

It has been observed that a belt of one or two cells wide, between any pair of cells using a particular frequency channel simultaneously, is sufficient to avoid mutual interference[8-29]. Figure 1.1 shows the lay-out of a cellular system using hexagonal-shaped cells with a buffer belt of one cell wide. The frequency reuse buffering area for this system consists of six cells. However, in the case of a two-cell wide buffer belt shown in figure 1.2, the buffer area consists of eighteen cells.

The smallest area in which all the frequency channels allocated for mobile use can be used simultaneously may be called the *cellular system unit*. This unit consists of three cells in the case of figure 1.1 and of seven

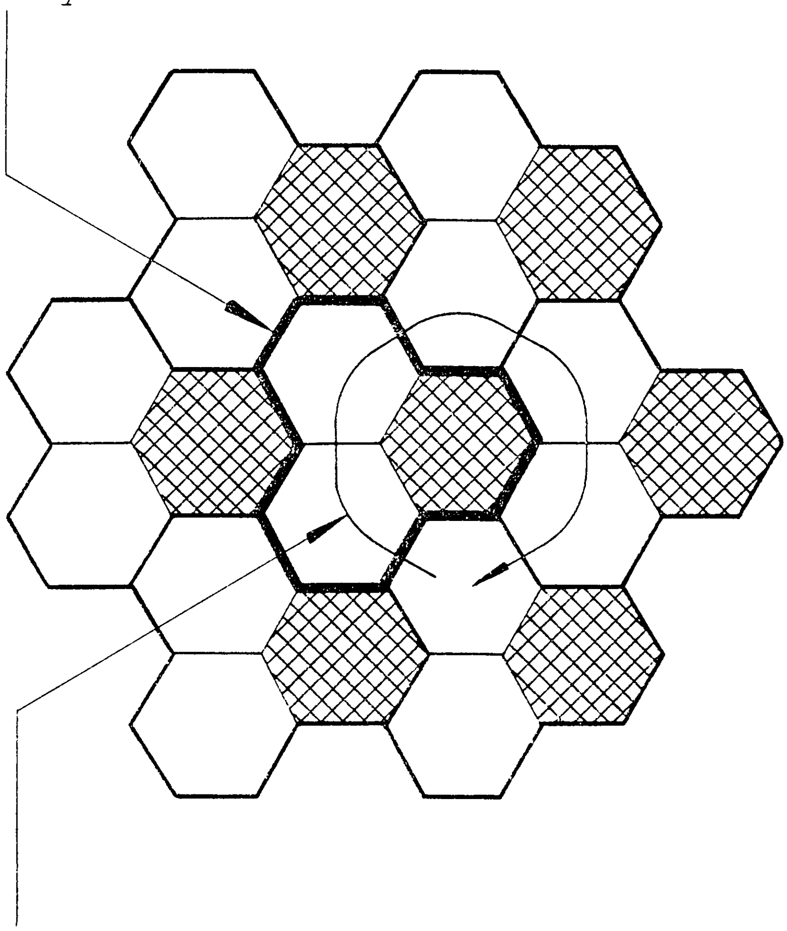

FIGURE (1.1)

THE LAY-OUT OF A CELLULAR SYSTEM WITH A BUFFER OF ONE-CELL WIDE BELT

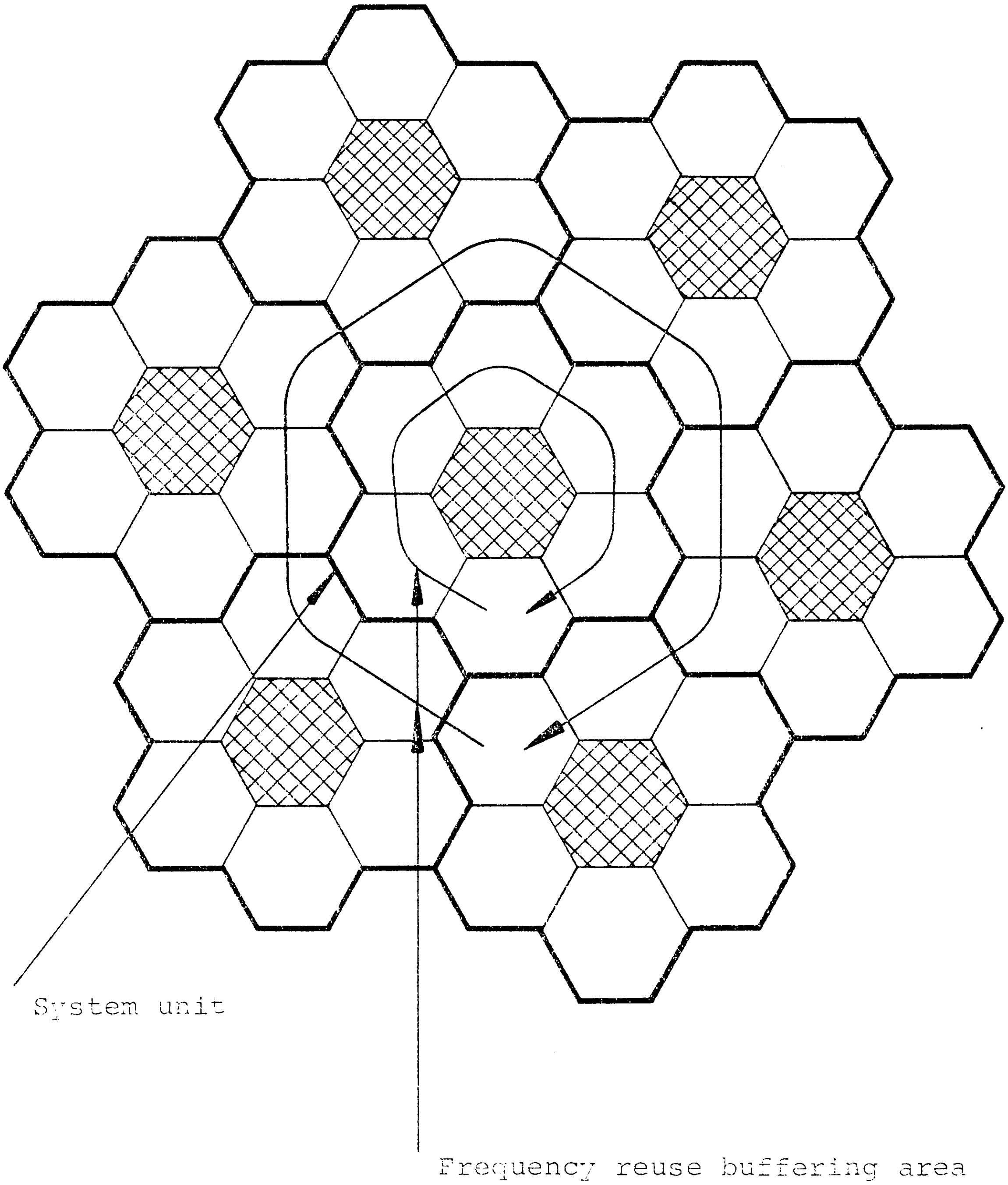

FIGURE (1.2)

THE LAY-OUT OF A CELLULAR SYSTEM WITH A BUFFER OF TWO-CELL WIDE BELT cells in the case of figure 1.2 .

Automatic control of cellular systems would be essential for providing instant and efficient mobile communication. Such control can be obtained by a set of interconnected computers. Thus, a cellular system would consist of a speech network for setting up voice paths and a controlling network for the management of the operations of the speech network. There are teletraffic problems associated with the speech network and control problems associated with the controlling network. The research reported here is concerned with the investigation of such problems.

1.2 TELETRAFFIC AND CONTROL PROBLEMS

Normal teletraffic theory is concerned with fixed station or land telephone systems. In such systems each call uses a single channel. In addition, each channel can only serve one call at a time. The teletraffic analysis of cellular mobile radio systems requires extensions of the normal theory. One reason for this is that in a single cell, in which calls go via the base station, there are two classes of call: mobile-to-mobile, and mobile-to-land. The first requires two channels per call, while the second needs only one. Another reason is that a single channel in a cellular system can serve many calls simultaneously, in different non-adjacent cells.

Cellular systems would be interconnected with land telephone networks. The population of the subscribers of a cellular system can be assumed to be very large, since it consists of all the mobile subscribers in all the cells, as well as the land network subscribers. However, a small number of mobile subscribers within a cell requires the assumption of a finite population. Lost call and delay systems may be considered.

The traffic capacity of a cellular system, depends on the method with which the system utilizes the frequency channels available for its use. The problem here is to ensure that frequency channels being used in certain cells are kept free within the necessary frequency reuse buffering areas. There are different schemes in which frequency channels can be assigned for use[14,22]. One such scheme is the *fixed channel assignment* in which frequency channels can only be used in designated cells. In this scheme there is a fixed relation between cells and the available frequency channels, as each cell would only use a fixed subset of these channels. Another scheme is *dynamic channel assignment*, in which all frequency channels available for the system can be used in any cell, provided that the frequency reuse buffering rules are maintained. In this scheme, there is no fixed relation between the cells and the channels that can be used within them, as channels are only temporarily assigned for use in cells for the duration of the call.

Each of the above two schemes has its problems. In a cellular system using the fixed channel assignment scheme, a call may be blocked in a particular cell, while there are free channels in the neighbouring cells. This problem does not exist with the dynamic channel assignment scheme. However, the dynamic scheme would need more equipment provision and would require access to and processing of large quantities of data. A combination of the fixed and dynamic schemes may be considered. Such a scheme may be called the *hybrid channel assignment scheme*[22].

The controlling computer network of a cellular system handles the speech communication traffic, with privacy assured by the absence of a human operator. This network should be able to deal with the complexity of the operations of such systems[23-29]. For example a mobile user needs to dial calls in the normal fashion and a called mobile needs to be located and a channel assigned to it. In addition, extra complexity is involved in the need to maintain calls in progress as mobiles move from one cell to another. Such a process would require the assignment of a channel to the call, via the base station of the new cell and the release of the channel used in the old cell. This is called the *hand-off* or the *hand-over* operation.

The controlling network should also have an appropriate mobile telephone numbering scheme. The problem here is that mobile subscribers may travel from one city to another changing their real addresses temporarily. The numbering scheme must ensure that the virtual addresses of such mobiles are kept fixed and associated with any changes in the real addresses.

The controlling network should perform efficiently the control operations needed. In addition, it should only use a minimum proportion of the frequency range available for the cellular system. The performance measures of such a network would include: network *reliability*, such as its capacity for dealing with errors and failures; network *responsiveness*, such as the time required for setting-up calls; and network *capacity* or *throughput*, that is the amount of traffic that can be handled during a specific time interval. Another important consideration is cost.

The design of the controlling network involves many different factors[30-53]. These include the choice of overall network structure, the communication protocols, the functions to be performed by individual processors and so on. There are generally too many different combinations for each possibility to be studied. It is therefore important for the designer to be able to investigate the various system configurations, which appear to be most relevant.

1.3 INVESTIGATION METHODS

Models of cellular mobile radio systems must be developed for the investigations of the different tele-traffic and control problems. Two approaches may be used for dealing with such models and for providing the best solutions to the problems involved: theoretical analysis and digital computer simulation. Queueing and normal tele-traffic theory techniques may be used for the theoretical analysis of some models[54-62]. Such models must be of limited complexity. However, digital computer simulation would be able to deal with models of more realistic complexity[63-70].

The simulation of a model involves the development of computer programs which reflect the appearance and show the effect of such a model. In doing this, simulation provides an insight into the behaviour of the model. Many different models may have to be considered. The reason for this is that the different possible solutions to the inter-related problems involved would produce too many combinations. To provide the system designer with the ability to simulate a wide range of models, it is useful for the simulation programs to be written in a modular form with well-defined interfaces. Any required model can be simulated by the use of standard modules with a minimum of programming effort being required to implement non-standard features.

The top-down approach to *structured programming* (71-76) is well-suited to the design of simulation programs. This approach involves refining the description of the model, step-by-step, from an initial brief description through progressively more detailed stages until, finally, an executable program results. This leads to better readability and assures the correctness of the programs. In addition, the approach leads naturally to programs having the required flexible modular construction. The resulting readability and flexibility of the programs is helpful when a new model has to be dealt with. Such a model can be simulated by the use of standard modules with a minimum of programming effort being required to implement non-standard features.

1.4 THE PRESENT RESEARCH

While cellular mobile radio systems are becoming of great future importance, demands for research into such systems are increasing. This research is concerned with the investigation of teletraffic and control aspects of such systems. Mathematical and computer simulation tools have been developed for these systems. Such tools provide both an insight into the behaviour of the systems, and also help in designing them.

In the next chapter, extensions of the normal tele-traffic theory are presented. These extensions deal with the requirements of the speech communication traffic of cellular systems. Theory is developed for several models. For single cell traffic, infinite and finite numbers of sources are considered using two different approaches. In addition, lost-call and delay systems are also considered. The new teletraffic formulas are compared with the normal teletraffic formulas. The theory is further extended to deal with cellular systems using three different channel assignment strategies: the fixed channel assignment, the dynamic channel assignment and the hybrid channel assignment. The validity of the theoretical results has been checked by simulation.

In chapter 3 simulation tools in the form of standard declarations, standard simulation procedures and standard timing rules are presented. These tools have been developed in accordance with the ideas of structured programming. They deal with both the speech network and the controlling network, and can be used for a wide range of system configurations.

As an example of the use of the speech network simulation tools, a system model has been considered. This model has been simulated on two computers: a DEC PDP-11/03 and an ICL 1904S. The simulation on the small machine was programmed in Coral 66[77-80] and used *roulette* simulation[56]. The roulette method is appropriate for small computers, but it has some restrictions. The simulation on the large computer was programmed in Algol 68[81-85], using *time true* simulation[63]. This is an expensive method but one which gives comprehensive answers.

The controlling network simulation tools consider the use of two transmission techniques. *Store-and-forward* transmission[42-45] is considered to be in use in the land links associated with the network and the *Aloha* packet switched technique[46-52] is considered to be in use in the radio links. These simulation tools have been programmed using Algol 68. This language enables new data structures to be defined and manipulated so that it is suitable for programming the simulation of complicated systems[66-81].

A possible and promising controlling network structure is presented in chapter 4. This network is a hierarchical computer network consisting of three main levels. The first level corresponds to the operations that can be dealt with at the single-cell or base-station level. The second level corresponds to the operations that can be dealt with at the system unit level. The third level corresponds to the operations that can be dealt with at the city level. Each of these levels consists of a network of interconnected processors. These networks are called the *cell network*, the *system unit network*, and the *inter-city network* respectively.

The controlling network simulation tools have been used for the simulation of the network structure chosen. Problems considered for investigation include the network capacity required to produce a satisfactory performance. Such problems have been developed step-by-step, advancing in complexity, starting with the basic control functions required within a single cell and moving to the more complex control problems, such as the hand-over function and the mobile address translation scheme. The simulation also considers other network variables such as routing strategies.

Finally, conclusions and suggestions for further work are outlined in chapter 5.

CHAPTER TWO

TELETRAFFIC ANALYSIS

2. TELETRAFFIC ANALYSIS

2.1 TELETRAFFIC MODELS

For the design of mobile radio systems, teletraffic formulas are desirable to enable the grade of service to be related to the traffic and the level of equipment provision. The present analysis gives a derivation of new teletraffic formulas for the time congestion in mobile radio systems. The analysis applies to a variety of mobile radio system configurations. These include:

1. Single-cell systems in which the only subscribers are the mobiles themselves.
2. Single-cell systems that are interconnected with the land telephone network so that land subscribers may call mobiles and vice versa.
3. Multi-cell systems which use fixed channel assignment so that each cell can be treated in isolation.
4. Multi-cell systems which use dynamic channel assignment so that any frequency channel available to the system may be used in any cell; subject to the frequency channel not already being in use in the frequency reuse buffering area of the cell.
5. Multi-cell systems which use hybrid channel assignment which is a combination of both fixed

and dynamic channel assignment.

It is assumed throughout the analysis that calls arise at random (pure chance traffic). Two cases are considered:

1. Very large population of mobiles or land subscribers, where the mean calling rate can be considered independent of the number of calls in progress.
2. Small population where the calling rate needs to be presented as proportional to the number of free mobiles. In this case the total number of mobiles, M, is assumed throughout the paper to be even.

Negative exponential distribution of holding times is assumed throughout, the mean holding time being assumed the same for all types of call.

It is assumed throughout that the number of frequency channels, C, is even. A call from a land subscriber to a mobile, or from a mobile to a land subscriber uses a single channel. Calls between mobiles within a cell are assumed to go via the base station and to require the simultaneous use of two channels *tromboning*.

Three types of call may be distinguished:

1. Land network originated calls for mobiles (type 1).
2. Mobile-originated calls for land network subscribers (type 2).
3. Mobile-originated calls for mobiles (type 3).

2.2 SINGLE-CELL SYSTEMS

2.2.1 Simple Analysis

The method of analysis presented in this section deals both with systems where the number of mobiles is, effectively, infinite and systems where the number of mobiles is finite. In each case, both lost calls (i.e. blocked calls are discarded) and delay (i.e. blocked calls are queued) operation is considered. The method is based on the following assumptions:

Assumption 1

$\alpha_i(x)$ is assumed equal to $\beta_i(x)$ where:

1. $\alpha_i(x)$ = the probability that the system leaves a state where x calls are in progress, due to the arrival of a call of type i, (i = 1,2,3).

2. $\beta_i(x)$ = the probability that the system arrives at a state where x calls are in progress, due to the termination of a call of type i.

Assumption 2

It is assumed that the fraction of channels engaged in calls of type i, when x channels are busy, is proportional to the traffic of type i carried and is independent of x. As an example, if one third of the traffic is of type 1, then it is assumed that, whenever 12 channels altogether are engaged, exactly 4 will be in use for calls of type 1.

Assumption 3

It is assumed that the difference between the traffic offered and the traffic carried may be neglected. That is, the congestion is assumed to be low.

Lost-Calls, Infinite Number of Mobiles

The probability of a state with x busy channels is denoted by S(x). From a state with x channels busy, the system can move, at a call arrival, to either a state with x+1 channels busy or to a state with x+2 channels busy. In order to calculate the blocking probability, it is appropriate to have S(x+1) and S(x+2) expressed in terms of S(x) as an intermediate step. The average number of calls in progress of type i is denoted by $[x_i | x]$, where i = 1,2,3. From assumption (1), above, the following equations can be derived.

$$S(x+1) = \frac{A_1+A_2}{[x_1+x_2\,|x+1]} S(x)$$

$$S(x+2) = \frac{A_3/2}{[x_3|\,x+2]} S(x)$$

where A_i is the offered traffic of call type i. The total traffic offered is denoted by A:

$$A = A_1 + A_2 + A_3$$

The following equations can be derived from assumptions (2) and (3).

$$[x_1+x_2\,|x+1]/(x+1) = (A_1+A_2)/A$$

$$[x_3|\,x+2]/(x+2) = A_3/(2A)$$

The foregoing equations may be combined to give the required equations for S(x+1) and S(x+2):

$$\begin{aligned} S(x+1) &= A\,S(x)/(x+1) \\ S(x+2) &= A\,S(x)/(x+2) \end{aligned} \tag{1}$$

In the usual way[55,56], equation (1) may be used to compute S(1), S(2),...,S(C) to within an unknown scale factor. The scale factor may then be determined by using

$$\sum_{x=0}^{C} S(x) = 1$$

The following formulas result for the congestion, S(C), for one and two-channel calls.

$$B_1 = B_2 = \frac{A^C/C!}{\sum\limits_{x=0}^{C} A^x/x!}$$

$$B_3 = \frac{(A/2)^{C/2}/(C/2)!}{\sum\limits_{x=0}^{C/2} (A/2)^x/x!} \tag{2}$$

These formulas are the same as Erlang's formula for lost call systems but, for the second one, with the traffic and the number of channels halved. It was not obvious at the outset that this familiar formula would emerge. If the congestion averaged over all types of call is required, these results can be averaged with weights in proportion to the traffic of each sort, giving

$$B = (A_1B_1 + A_2B_2 + A_3B_3)/(A_1+A_2+A_3) \tag{3}$$

The results above show that the effect of tromboning can be taken into account while Erlang's formula, or existing tables of its values, are used. Rather than use the formula directly, it is necessary to average together two values obtained from it, one for single channel calls and one for two-channel calls.

Lost-Calls, Finite Number of Mobiles

Congestion formulas for the case where the number of mobiles is finite may be derived in a way similar to that above. Assumption (1) leads to the equations

$$S(x+1) = \frac{(a_1+a_2)(M-x)}{[x_1+x_2|x+1]} S(x)$$

$$S(x+2) = \frac{a_3(M-x)}{2[x_3|x+2]} S(x)$$

where a_i is the offered traffic per free mobile, of call type i (i = 1,2,3). The total traffic per free mobile is denoted by $a(=a_1+a_2+a_3)$. By working on the usual lines, the following formulas result.

$$B_1 = B_2 = \frac{M!\ a^C/[(M-C)!\ C!]}{\sum_{x=0}^{C} M!\ a^x/[(M-x)!\ x!]} \tag{4}$$

$$= B_3 = \frac{(M/2)!\ a^{C/2}/\{[(M-C)/2]!(C/2)!\}}{\sum_{x=0}^{C/2} (M/2)!\ a^x/[(M/2-x)!\ x!]}$$

The congestion averaged over calls of all types, is given by

$$B = (a_1B_1 + a_2B_2 + a_3B_3)/(a_1 + a_2 + a_3) \tag{5}$$

The formulas above are the same as Engset's formula but, for the second one, with the number of mobiles and the number of channels halved. As with the previous case, it was not obvious at the outset that a familiar formula would emerge.

Delay System, Infinite Number of Mobiles

In lost call systems, the maximum number of concurrent demands for the use of a frequency channel that can be met is C, the number of frequency channels. In a delay system, where calls that cannot be dealt with immediately are queued, the maximum number of demands that may be coped with simultaneously is equal to C plus the size of the buffer in which demands awaiting a channel are queued. It is assumed throughout this chapter that the buffer is of unlimited size, so that, x, the number of demands being treated by the system, may range from zero to infinity.

For values of x which result in no calls being queued, the equal transition probability equations are the same as for the corresponding lost-call system. For values of x sufficiently large to result in calls being delayed, the following equal transition probability equations may be derived from assumption (1).

$$S(x+1) = (A_1+A_2)\ S(x)/\left[(C_1+C_2)|C\right]$$

$$S(x+2) = \tfrac{1}{2}\ A_3\ S(x)/\left[C_3|C\right]$$

The congestion, i.e. the probability of delay, is $\sum_{x=C}^{\infty} S(x)$ From the equal transition probability equations the following formulas may be derived by making use of assumptions (2) and (3).

$$B_1 = B_2 = \frac{A^C/[(C-1)!(C-A)]}{A^C/[(C-1)!(C-A)] + \sum_{x=0}^{C-1} A^x/x!} \qquad (6)$$

$$B_3 = \frac{(A/2)^{C/2} C/[(C/2)!(C-A)]}{(A/C)^{C/2} C/[(C/2)!(C-A)] + \sum_{x=0}^{C/2-1} (A/2)^x/x!}$$

The congestion, averaged over all types of call can be calculated by averaging the individual congestions, via the use of equation (3).

Delay Systems, Finite Number of Mobiles

The equations developed for lost-call systems with a finite number of mobiles are applicable for $x<C$. The number of demands awaiting service (i.e. waiting in the queue), for calls of type 1, may range zero to infinity. The number of demands awaiting service for calls of type 2 and 3 ranges from zero to M-C. A delayed call of type 2 or type 3 occupies a mobile, whereas a delayed call of type 1 occupies none. From assumption (1) the following equations may be derived, for $x \geq C$.

$$S(x+1) = (M-C)\ a_1\ S(x)/[C_1|C]$$

$$S(x+1) = (M-C-w_2)\ a_2\ S(x)/[C_2|C] \quad (7)$$

$$S(x+2) = \tfrac{1}{2}(M-C-w_3)\ a_3\ S(x)/[C_3|C]$$

where w_i denotes the number of delayed calls of type i. From these equal transition probability equations, and by the use of assumptions (2) and (3), C_1, C_2 and C_3 may be eliminated. By summation of the resulting expressions from x=0 to infinity, in the case of calls of type 1 or to M-C, in the case of calls of types 2 and 3, the required congestion formulas are obtained for each class of call.

$$B_1 = \frac{M!a^C/\{(M-C)!(C-1)![C-(M-C)a]\}}{M!a^C/\{(M-C)!(C-1)![C-(M-C)a]\} + \sum_{x=0}^{C-1}(M!a^x)/[(M-x)!x!]}$$

$$B_2 = \frac{M!a^C(\sum_{w_2=0}^{M-C}[a(M-C-w_2)/C]^{w_2})/[(M-C)!C!]}{M!a^C(\sum_{w_2=0}^{M-C}[a(M-C-w_2)/C]^{w_2})/[(M-C)!C!] + \sum_{x=0}^{C-1}(M!a^x)/[(M-x)!x!]}$$

$$(8)$$

$$B_3 = \frac{(M/2)!a^{C/2}(\sum_{w_3=0}^{M-C}[a(M-C-w_3)/C]^{w_3})/\{[(M-C)/2]!(C/2)!\}}{(M/2)!a^{C/2}(\sum_{w_3=0}^{M-C}[a(M-C-w_3)/C]^{w_3})/\{[(M-C)/2]!(C/2)!\} + \sum_{x=0}^{C/2-1}[(M/2)!a^x]/[(M/2-x)!x!]}$$

The congestion, averaged over calls of all types, can be obtained by combining these equations via equation (5).

As in the case with an infinite number of mobiles, the formulas above (for lost-call systems) are the familiar Engset formulas but, for two-channel calls, with the number of channels and the number of mobiles halved. Here, too, it was not obvious in advance that this result would emerge. The result shows that the Engset formula, or tables of its values can be applied in cases where appreciable tromboning occurs and the number of mobiles is small.

The formulas derived above express the congestion in terms of a, the offered traffic per free mobile. It is often useful, instead, to be able to express the congestion in terms of A, the average offered traffic. The relation between a and A may be derived as follows. Using the arguments given by D. Bear[56], the following equation may be obtained

$$a = A/[M-A(1-B_{call})] \qquad (9)$$

where B_{call} represents the call congestion averaged over calls of all types. On the assumption that the call congestion is approximately equal to the time congestion, averaged over calls of all types, equation (9) can be used to enable the congestion to be plotted as a function of A. If the congestion is small, A can be given explicitly in terms of a:

$$A = Ma/(1+a)$$

2.2.2 Alternative Approach

This section presents an alternative approach to the calculation of the congestion in lost call systems. The method is based on the use of different assumptions from those of the previous section and gives greater accuracy. Pure chance traffic with either a finite or an infinite number of mobiles are considered and the distribution of holding times is assumed to be negative exponential. In lost-calls systems, there is not distinction between calls of type 1 and of type 2 so far as congestion calculations are concerned; such calls are termed *one-channel* calls in the remainder of this section. Calls of type 3 are referred to as *two-channel* calls.

The approach, which is common to the various cases to be considered, can be outlined as follows:

1. The states in which blocking occurs are identified. A state, for present purposes, is a pair of values, x and y, which are, respectively, the number of one- and two-channel calls in progress.

2. The blocking probability is expressed in terms of $P(x)$ and $Q(y)$, the general discrete probability density functions of x and y.

3. For each case considered, a particular expression is produced for $P(x)$ and $Q(y)$. When substituted for $P(x)$ and $Q(y)$ in the blocking probability expression, the required blocking probability formula results.

Let x and y represent, respectively, the number of one-channel and two-channel calls in progress. If there were an infinite number of frequency channels and an infinite number of mobiles, x and y would be statistically independent. In this case, if $P(x)$ and $Q(y)$ represent the discrete probability density functions of x and y, $T(x,y)$, the joint probability of having x one-channel calls and y two-channel calls in progress, is given by

$$T(x,y) = P(x)\ Q(y)$$

It is assumed that in cases where the number of frequency channels or the number of mobiles (or both) is finite, the joint probability may be given by:

$$T(x,y) = \begin{cases} z\ P(x)\ Q(y), & \text{for } x,y \text{ in the set } D \\ 0, & \text{for } x,y \text{ not in the set } D. \end{cases}$$

The set, D represents the set of physically possible states i.e. the set of pairs of x,y values which can occur. The set D is defined by the condition

$$0 \leqslant x + 2y \leqslant C$$

The term, z, is a scale factor to keep the total probability as unity

$$\sum_{x=0}^{C} \sum_{y=0}^{\langle (C-x)/2 \rangle} z\, P(x)\; Q(y) = 1 \qquad (10)$$

where $\langle\cdot\rangle$ represents the operation of taking the integral part.

Arriving one-channel calls are blocked when no free channels are available. This occurs when the system is in a state for which $x + 2y = C$. The blocking probability for one channel calls is thus given by

$$B_1 = \sum_{x=0}^{C/2} z\, P(2x)\; Q(C/2-x) \qquad (11)$$

Arriving two-channel calls are blocked when one or no free channels are available. This occurs when the system is in states for which $x+2y = C-1$, or when it is in states for which $x+2y = C$. The blocking probability for two channel calls is given by

$$B_3 = B_1 + \sum_{x=0}^{C/2-1} z\, P(2x+1)\; Q(C/2-x-1) \qquad (12)$$

It now remains to obtain specific forms for P(x) and Q(y), in the cases where the number of mobiles is infinite and finite. Substitution of these forms in (10), (11) and (12) will yield the required congestion formulas.

Infinite Number of Mobiles

It is assumed that the state probabilities for a system having an infinite number of mobiles is adequately approximated by a Poisson distribution i.e. P(x) has the same form that it would have with an unlimited number of frequency channels:

$$P(x) = \exp[-(A_1+A_2)]\,(A_1+A_2)^x/x!$$

Similarly, Q(y) is assumed to have the form

$$Q(y) = \exp[-A_3/2]\,(A_3/2)^y/y!$$

Substitution of these expressions into (10), (11) and (12) yields the following equations for the congestion with an infinite number of mobiles.

$$B_1 = B_2 = \frac{\sum_{x=0}^{C/2} [(A_1+A_2)^{2x}(A_3/2)^{(C/2-x)}]/[(2x)!(C/2-x)!]}{\sum_{x=0}^{C} \sum_{y=0}^{<(C-x)/2>} [(A_1+A_2)^x (A_3/2)^y]/(x!y!)} \tag{13}$$

$$B_3 = B_1 + \frac{\sum_{x=0}^{C/2-1} [(A_1+A_2)^{(2x+1)}(A_3/2)^{(C/2-x-1)}]/[(2x+1)!(C/2-x-1)!]}{\sum_{x=0}^{C} \sum_{y=0}^{<(C-x)/2>} [(A_1+A_2)^x (A_3/2)^y]/(x!y!)}$$

The congestion, averaged over one and two channel calls may be calculated by the use of equation (3).

It may be of interest to note that a similar case to the problem of infinite number of mobiles is presented by D. Bear[56] (p. 46). The case is for a fixed-station local exchange that may have to connect two of its subscribers using two of its trunks. The exchange is assumed to have a large number of subscribers, so that the traffic stream is Poissonian. The formulas presented for the case are of more complicated nature than the formulas derived here for the problem of infinite number of mobiles. Further refinements and substitutions have been required for computing the local exchange formulas. Although the analytical method presented by Bear[56] is different than the method used here, the two methods produce the same results.

Finite Number of Mobiles

The case is now considered where there are M mobiles (M even) and the traffic of type i offered by each free mobile, is a_i ($i = 1,2,3$). It is now assumed that the probability that the system goes from state x to state x+1 by a one channel call arrival is equal to the probability that it goes from state x+1 to state x by a call termination. That is,

$$(x+1)\ P(x+1) = (M-2y-x)(a_1+a_2)P(x)$$

By the recursive use of this equation, P(x) can be expressed in terms of P(O):

$$P(x) = \frac{(M-2y)!\,(a_1+a_2)^x}{(M-2y-x)!\,x!}\,P(O)$$

In a parallel way, the probability distribution for two channel calls is obtained:

$$Q(y) = \frac{[(M-x)/2]!\ a_3^{\,y}}{[(M-x)/2-y]!\,y!}\,Q(O)$$

Substitution of these expressions into (10), (11) and (12) results, after simplification and cancellation of P(O) and Q(O) in the required congestion formulas:

$$B_1=B_2=\frac{\displaystyle\sum_{x=0}^{C/2}\frac{(M+2x-c)!\,(a_1+a_2)^{2x}(M/2-x)!\ a_3^{\,(C/2-x)}}{(M-C)!\,(2x)!\,[(M-C)/2]!\,(C/2-x)!}}{\displaystyle\sum_{x=0}^{C}\ \sum_{y=0}^{<(C-x)/2>}\frac{(M-2y)!\,(a_1+a_2)^x[(M-x)/2]!\ a_3^{\,y}}{(M-2y-x)!\,x!\,[(M-x)/2-y]!\ y!}} \tag{14}$$

$$B_3=B_1+\frac{\displaystyle\sum_{x=0}^{C/2-1}\frac{(M+2x+2-C)!\,(a_1+a_2)^{(2x+1)}[(M-2x-1)/2]!\ a_3^{\,(C/2-x-1)}}{(M+1-C)!\,(2x+1)!\,[(M+1-C)/2]!\,(C/2-x-1)!}}{\displaystyle\sum_{x=0}^{C}\ \sum_{y=0}^{<(C-x)/2>}\frac{(M-2y)!\,(a_1+a_2)^x[(M-x)/2]!\ a_3^{\,y}}{(M-2y-x)!\,x!\,[(M-x)/2-y]!\ y!}}$$

As before, the congestion, averaged for one and two-channel calls may be calculated by the use of equation (5).

It is of some interest to note that, in the case of zero mobile-to-mobile traffic, the familiar Erlang and Engset formulas may be derived by starting with the same assumptions used in this section and by using similar methods. This is a satisfying result in view of the difference between the method of the present section and the previous one.

2.2.3 Comparison with Simulation Results

The accuracy of the formulas given in the preceding sections has been evaluated by comparison with results obtained from computer simulation of the models. To provide confidence in the simulation results, two separate simulations were used. One program used the time-true or event-by-event method[63,66]. The other program used the roulette method[56,57,69] of traffic simulation. These simulation program are described in the next chapter. The use of two separate simulation methods, programmed in different languages (Algol 68 and Coral 66), run on different computers (ICL 1904S and PDP-11/03) and using different pseudo-random number generators, was intended to provide a check on the validity of the simulations. Outside the statistical variation to be expected, no discrepancies between the simulations were observed.

The results of comparing the congestion estimated from the simulations with that predicted by the formulas derived in section 2.2.1 may be summarised as follows. The predicted congestion, averaged over calls of all types, agrees with the simulation results, for values of traffic for which the average congestion is not more than 0.9. This good agreement for the average congestion results from partial cancellation of the error in the predicted congestion for one-channel calls with that for two-channel calls. The congestion for one-channel calls is overestimated and the congestion for two-channel calls is underestimated. Agreement to within ten percent between the predicted values and the simulation results, for one and two-channel calls, only existed for values of traffic for which the congestion is less than 0.1. As an illustration of the results, figure 2.1, provides a comparison between the congestion predicted by equation (2) and the simulation results for the case of 20 channels and equal proportions of traffic of each type. The vertical lines about each point of the simulation indicate a 90 percent confidence interval[86]. Figure 2.2 shows the results predicted by equation (4), in a similar case to that of figure 2.1, but with a total of 200 mobiles.

The difference between the discrepancies for the two types of traffic may be explained as follows. The analysis employed the assumption that the number of calls in progress

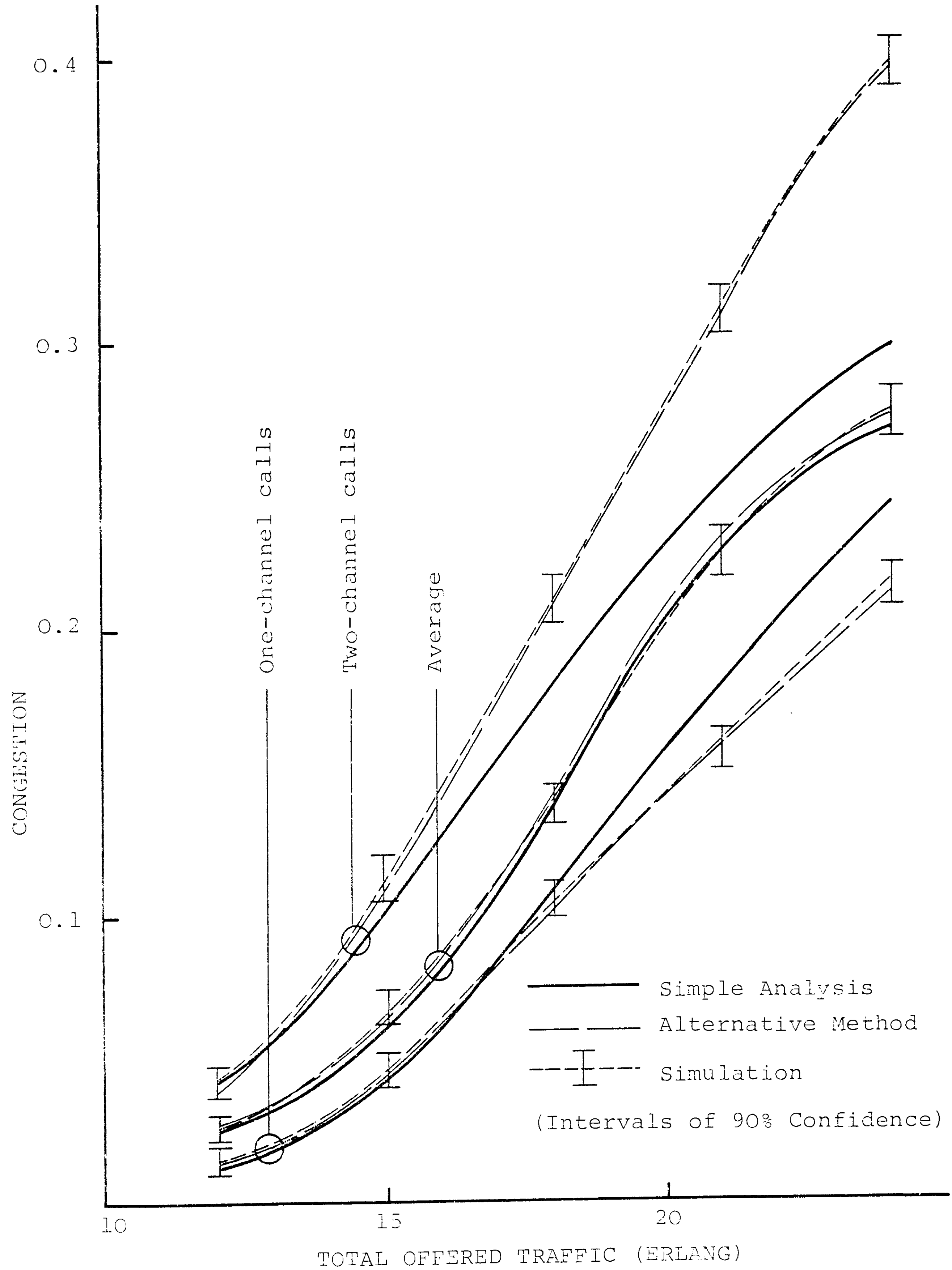

FIGURE (2.1)

THEORETICAL AND SIMULATION RESULTS WITH INFINITE TRAFFIC SOURCE AND 20 CHANNELS. EQUAL PROPORTIONS OF TYPE 1, TYPE 2 AND TYPE 3 TRAFFIC.

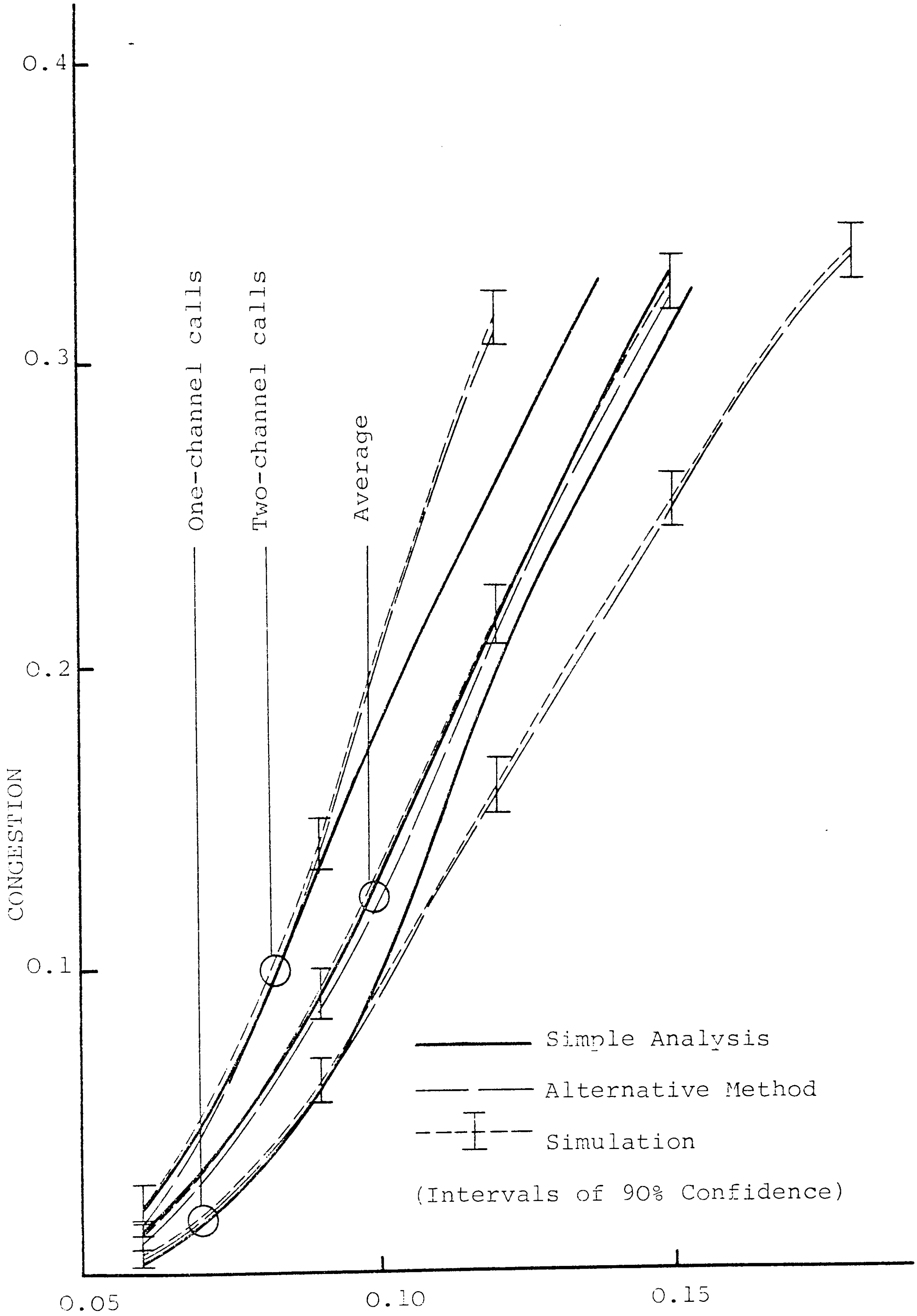

FIGURE (2.2)

THEORETICAL AND SIMULATION RESULTS WITH 200 MOBILES AND 20 CHANNELS. EQUAL PROPORTIONS OF TYPE 1, TYPE 2 AND TYPE 3 TRAFFIC of type i is proportional to the offered traffic of that type e.g.

$$(x_1 + x_2)/x_3 = (A_1+A_2)/(A_3/2)$$

In reality, the average number of calls of type i is proportional to the carried traffic of that type e.g.

$$(x_1 + x_2)/x_3 = [(A_1+A_2)(1-B_1)]/[(A_3/2)(1-B_3)]$$

The difference is accounted for by the fact that there will be a bigger relative difference between the offered and carried traffic for two-channel calls than there is for one-channel calls. This is because two-channel calls are blocked both when all channels are occupied and when all but one are occupied, whereas one-channel calls are only blocked when all channels are occupied.

The formulas derived in section 2.2.2 showed agreement to within ten percent of the simulation results for values of traffic for which the congestion does not exceed 0.4. This applied for both one-channel and two-channel calls, as well as for the average congestion. (See figures 2.1 and 2.2).

In the case where the number of mobiles is finite, the congestion, in reality, will be zero if the number of mobiles does not exceed the number of channels $M \leq C$. Equation (14) predicts nonzero but very small values of congestion under this condition. For example, in the case where M=C=20 and with an offered traffic of 1E per free mobile (equal proportions of each type of call), the predicted congestion about 10^{-6}.

2.3 MULTI-CELL SYSTEMS

Three types of multi-cell mobile radio systems are considered: systems with fixed channel assignment, systems with dynamic channel assignment and hybrid systems with a combination of fixed and dynamic channel assignment.

2.3.1 Fixed Channel Assignment

With systems using fixed channel assignment, a subset of the frequency channels available to the overall system are allocated to each cell. Each cell can, for the purposes of teletraffic analysis, be considered in isolation, that is, as if it were a single-cell system.

2.3.2 Dynamic Channel Assignment

In a multi-cell system with dynamic channel assignment, any frequency available to the system may be used in any cell, subject to the frequency not already being in use in the cell or in the buffer zone around it. The analysis presented in the following is based on the approach described in section 2.2.2 with some modifications. As before, P(x) represents the probability of having x one-channel calls in progress in a system with an infinite number of mobiles and of channels. $Q(y)$ represents the probability of having y two-channel calls under these circumstances. If the number of channels is finite, there is a possibility that insufficient channels will be available for use within the cell to support x one-channel calls and y two-channel calls. The probability that there *are* sufficient channels is denoted by $F(x+2y)$. For systems with a finite number of channels the probabilities of having x one-channel calls, y two-channel calls and of having x+2y or more channels not in use in the buffering area (and therefore useable in the cell considered) are inter-dependent. It is assumed that, $T(x,y)$, the probability of having x one-channel calls and y two-channel calls is, approximately, given by

$$T(x,y) = \begin{cases} zF(x+2y)P(x)Q(y), & \text{for } 0 \leq x+2y \leq C \\ 0 & \text{otherwise} \end{cases} \tag{15}$$

z is a scale factor to give unit total probability, that is

$$\sum_{x=0}^{C} \sum_{y=0}^{<(C-x)/2>} zF(x+2y)\, P(x)Q(y) = 1 \tag{16}$$

The blocking probability for one-channels calls, B_1, is the probability that all channels not in use in the buffer area are in use in the cell. This probability is approximated by

$$B_1 = B_2 = \sum_{x=0}^{C} \sum_{y=0}^{<(C-x)/2>} zf(x+2y)\ P(x)\ Q(y) \qquad (17)$$

where f(x+2y) is the probability that exactly x+2y channels are not in use in the buffering area. The probability of having just one channel available, which precludes a mobile-to-mobile call from being set up, is given by

$$\sum_{x=0}^{C-1} \sum_{y=0}^{<(C-x-1)/2>} zf(x+2y+1)\ P(x)\ Q(y) \qquad (18)$$

The approximate congestion for two-channel calls is given by the sum of (17) and (18):

$$B_3 = B_1 + \sum_{x=0}^{C-1} \sum_{y=0}^{<(C-x-1)/2>} zf(x+2y+1)\ P(x)\ Q(y) \qquad (19)$$

Expressions for f(x) and F(x) are required. Under the assumption that the state of a channel is independent of the state of other channels, the probability that, out of

C channels, x of them are free is given by the binomial distribution:

$$f(x) = \frac{C!}{x!(C-x)!} V^{C-x} (1-V)^x \tag{20}$$

where V is the channel occupancy. From this, the probability of having x or more channels available for use in a cell is given by

$$F(x) = \sum_{W=x}^{W=C} \frac{C!}{W!(C-W)!} V^{C-W} (1-V)^W \tag{21}$$

To use the foregoing formulas it is necessary to know V, the channel occupancy. This is given by

$$V = 1 - (1 - \frac{K}{CN})^g \tag{22}$$

where K is the traffic carried by the system and N is the number of cells in the system. g is the average number of cells per frequency reuse buffering area. Recall that frequency reuse buffering areas on the periphery of the system have fewer cells than other areas. Thus g, averaged over the system, may differ from the number of cells in a specific frequency reuse buffering area. For systems with very many cells, g approaches 6, for buffering with single belts of cells and 18, for buffering with double belts. A method for computing g for systems with a limited number of cells is presented in Appendix 1. As before, C is the number of frequency channels available to the system. In cases where the blocking probability is low, the offered traffic per cell, A, is approximately equal to the traffic carried, K/N, and V may be approximated by

$$V \simeq 1 - (1 - \frac{A}{C})^{g} \qquad (23)$$

The channel occupancy can also be expressed, approximately, in terms of the traffic offered per free mobile. Following the arguments given by D. Bear[56] (p.44), the following approximate expression can be derived under the assumptions that the congestion is low:

$$V \simeq 1 - (1 - \frac{Ma}{(1+a)C})^{g} \qquad (24)$$

where a is the offered traffic per free mobile and M is the number of mobiles per cell.

Infinite Number of Mobiles

It is assumed that the probability of having x calls in progress, for a system with, effectively, an infinite number of mobiles is adequately approximated by a Poisson distribution;

$$P(x) = \exp[-(A_1+A_2)](A_1+A_2)^{x}/x!$$

Similarly, Q(y) is assumed to have the form

$$Q(y) = \exp(-A_3/2)(A_3/2)^y/y!$$

where A_i is the traffic of type i offered per cell (i=1,2,3).

By substitution of these expressions in equations (16), (17) and (19), formulas are obtained for the congestion in a system with dynamic channel assignment and an infinite number of mobiles:

$$B_1 = B_2 = \frac{\sum_{x=0}^{C} \sum_{y=0}^{\langle (C-x)/2 \rangle} \left[(f(x+2y)(A_1+A_2)^x (A_3/2)^y \right] / x!y!}{\sum_{x=0}^{C} \sum_{y=0}^{\langle (C-x)/2 \rangle} \left[F(x+2y)(A_1+A_2)^x (A_{3/2})^y \right] / x!y!} \tag{25}$$

$$B_3 = B_1 + \frac{\sum_{x=0}^{C-1} \sum_{y=0}^{\langle (C-x-1)/2 \rangle} \left[f(x+2y+1)(A_1+A_2)^x (A_3/2)^y \right] / x!y!}{\sum_{x=0}^{C} \sum_{y=0}^{\langle (C-x)/2 \rangle} \left[F(x+2y)(A_1+A_2)^x (A_3/2)^y \right] / x!y!}$$

These results may be combined by the use of equation (3), if the congestion averaged over all sorts of call is required.

It may be of interest to note that L. Schiff[14] has given some teletraffic formulas for multi-cell systems with dynamic channel assignment and Poisson traffic. These formulas are based on complicated factors and the use of these formulas presents some practical difficulties.

Finite Number of Mobiles

For systems with a finite number of mobiles M (assumed even), the traffic of type i offered per free mobile is a_i (i=1,2,3). As before, it is assumed that the probability that the system is changed from having x single-channel calls in progress to having x+1, by a call arrival is equal to the probability that it goes from x+1 to x by a call termination. That is,

$$(x+1)\ P(x+1) = (M-2y-x)(a_1+a_2)\ P(x)$$

By repeated application of this equation, P(x) can be expressed in terms of P(O):

$$P(x) = \frac{(M-2y)\ \ (a_1+a_2)^x}{(M-2y-x)!x!}\ P(O)$$

In a similar fashion, the probability of having y two-channel calls is obtained:

$$Q(y) = \frac{[(M-x)/2]!\,a_3^y}{[(M-x)/2-y]!}\ Q(O)$$

By substitution of these expressions into equations (16), (17) and (19), the required formulas for the congestion in systems using dynamic channel assignment and having a finite number of mobiles are:

$$B_1=B_2=\frac{\sum_{x=0}^{C}\sum_{y=0}^{<(C-x)/2>} f(x+2y)\,\frac{(M-2y)!\,(a_1+a_2)^x[(M-x)/2]!\;a_3^y}{(M-2y-x)!\,x!\,[(M-x)/2-y]!\,y!}}{\sum_{x=0}^{C}\sum_{y=0}^{<(C-x)/2>} F(x+2y)\,\frac{(M-2y)!\,(a_1+a_2)^x[(M-x)/2]!\;a_3^y}{(M-2y-x)!\,x!\,[(M-x)/2-y]!\,y!}} \tag{26}$$

$$B_3=B_1+\frac{\sum_{x=0}^{C-1}\sum_{y=0}^{<(C-x-1)/2>} f(x+2y+1)\,\frac{(M-2y)!\,(a_1+a_2)^x[(M-x)/2]!\;a_3^y}{(M-2y-x)!\,x!\,[(M-x)/2-y]!\,y!}}{\sum_{x=0}^{C}\sum_{y=0}^{<(C-x)/2>} F(x+2y)\,\frac{(M-2y)!\,(a_1+a_2)^x[(M-x)/2]!\;a_3^y}{(M-2y-x)!\,x!\,[(M-x)/2-y]!\,y!}}$$

The results may be combined by the use of equation (5), if the congestion averaged over all sorts of call is required.

2.3.3 Hybrid Channel Assignment

The formulas that have been derived may be combined to produce formulas which apply to systems using hybrid channel assignment. In a hybrid channel assignment system, the channels are divided into two groups. C_f of the channels are allocated to cells, individually for use as fixed channels and C_d are allocated to the entire system, for dynamic assignment. In each cell, traffic is offered first to the fixed channels. Traffic that is not accommodated by the fixed channels is then offered to the dynamically assigned channels. In a system with a considerable number of cells, congestion in the dynamic channels can be considered as pure chance traffic and independent of the congestion of the fixed channels in an individual cell. The overall congestion of the hybrid system, may therefore be approximated as the product of the fixed channel congestion and the dynamic channel congestion. The congestion for calls of type i is now denoted by B_{f_i}, B_{d_i} and B_{h_i} in the fixed, dynamic and hybrid channel assignment cases, respectively, and where i=1,2,3. The traffic offered to the dynamic channels is the overflow from the fixed channels and has the value $(NA_1 B_{f_1} + NA_2 B_{f_2} + NA_3 B_{f_3})$. The congestion for calls of type i is thus given by

$$B_{h_i}(C, NA_1, NA_2, NA_3) = b_i \; B_{d_i}(C_d, NA_1 b_1, NA_2 b_2, NA_3 b_3) \tag{27}$$

where

$$b_i = B_{f_i}(C_f, NA_1, NA_2, NA_3), \qquad i=1,2,3$$

which is given by the analysis of single-cell systems. These results can be used straightforwardly to obtain the congestion, averaged over calls of all types.

Computer programs have been developed using Coral 66 for computing the teletraffic formulas derived here. The programs are based on the algorithms shown in the form of flowcharts in Appendix I. As an example the listing of one computer program is presented together with traffic-congestion tables.

2.3.4 Comparison With Other Results

This section provides a comparison between the theoretical results obtained in previous sections and results, published elsewhere, obtained by simulation.

Fixed Channel Assignment

In multi-cell systems with fixed channel assignment, each cell can be considered as if it were a single-cell system. The results obtained for single-cell systems in Section 2.2.3 are therefore applicable to the case of multi-cell systems with fixed channel assignment. These results showed close agreement with results obtained by simulation.

Dynamic and Hybrid Channel Assignment

A number of authors have described the investigation of teletraffic in mobile radio systems by the use of simulation(15,22). A recent paper, by Kahwa and Georganas(22) describes the simulation of a large system using hybrid channel assignment. They provide a comparison between their results and the results presented by Cox and Reudink in an earlier paper[19].

The system considered by Kahwa and Georganas consisted of 40 hexagonal cells of equal size. A frequency reuse buffering area of one belt of cells was used. In an example considered by Kahwa and Georganas, for comparison with the results of Cox and Reudink, the system is allocated 30 channels in two ways:

1. Fixed channel assignment, with 10 channels per cell.

2. Hybrid channel assignment, with 24 channels allocated on a fixed basis (8 per cell) and 6 channels allocated for dynamic assignment - an average of 2 dynamic channels per cell.

Figure 2.3 shows the predicted congestion, obtained from equation (13), for the fixed channel assignment case and from equation (27), for the hybrid case. Also shown in this graph are the congestion values obtained by Kahwa and Georganas from their simulations of hybrid assignment systems (taken from their figure 4.2). Their graph is plotted with the traffic expressed as percentage increase over 5 erlangs per cell; figure 2.3 is also plotted this way.

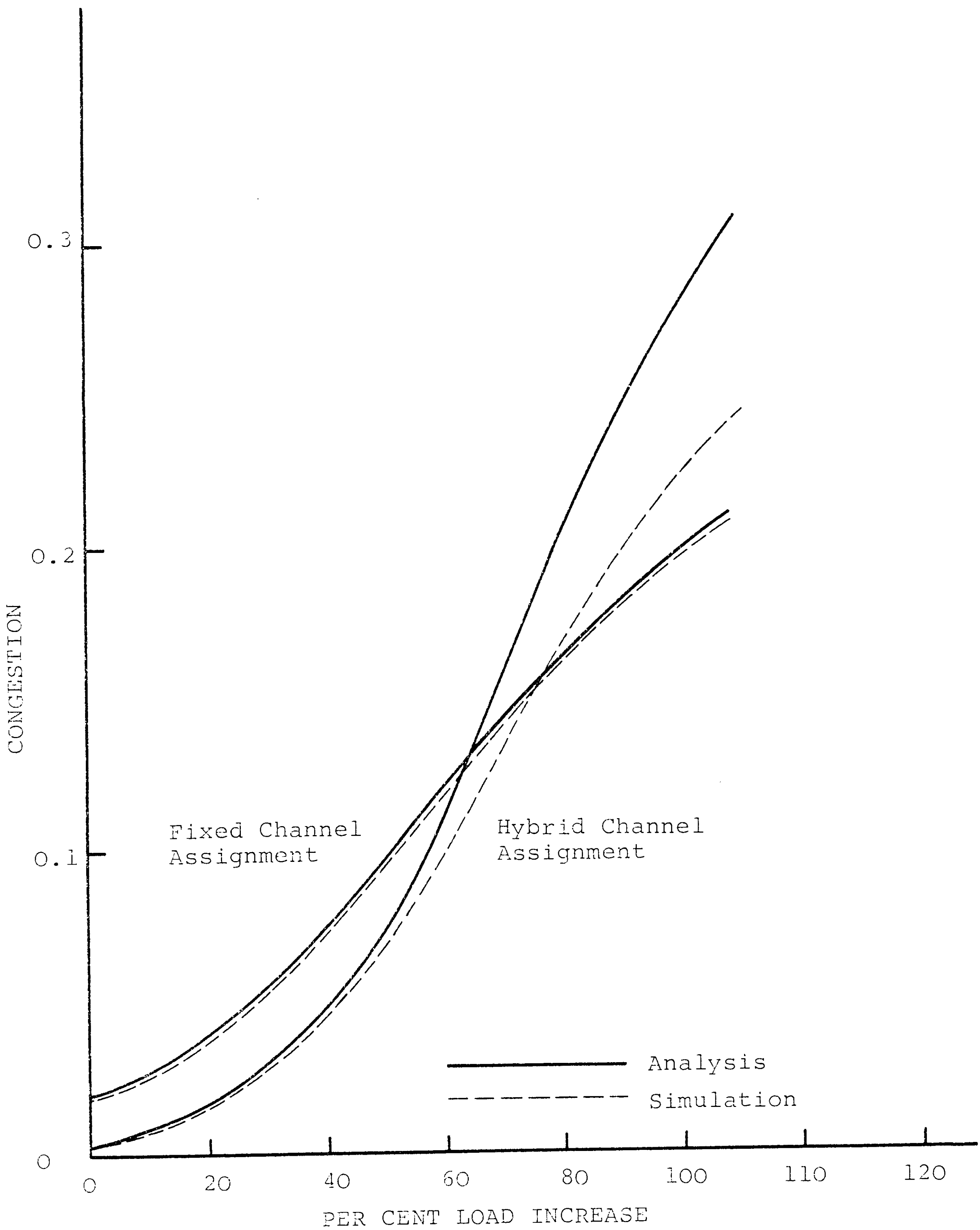

FIGURE (2.3)

COMPARISON BETWEEN THEORETICAL PREDICTED CONGESTION AND SIMULATION RESULTS AT MODERATE CONGESTION VALUES. BASIC LOAD=5 ERLANGS PER CELL; 10 CHANNELS PER CELL (FIXED CHANNEL ASSIGNMENT); 8 FIXED, 2 DYNAMIC CHANNELS PER CELL (HYBRID CHANNEL ASSIGNMENT).

For hybrid channel assignment, the theoretical results agree with the simulation of Kahwa and Georganas for traffic values giving congestion of 0.1 or less. At higher values of traffic, the formulas appear to predict too high a congestion. Some departure at high traffic levels is to be expected because one of the original assumptions was that the congestion is low.

The single-cell roulette simulation mentioned in section 2.2.3 and described in the next chapter was used to provide the results shown in figure 2.3 for the case of fixed-channel assignment. The agreement is close for the whole range considered; that is, up to a congestion value of 0.2.

Kahwa and Georganas also consider cases where lower congestion values are obtained. Two systems, each with 54 channels are considered:

1. Fixed assignment with 18 channels per cell.

2. Hybrid assignment with 14 fixed channels and an average of 4 dynamic channels per cell.

Figure 2.4 provides a comparison between results from equations (13) and (27) and the results given in figure 4.3 in[22], in the hybrid channel assignment case, and with results from roulette simulation for the fixed

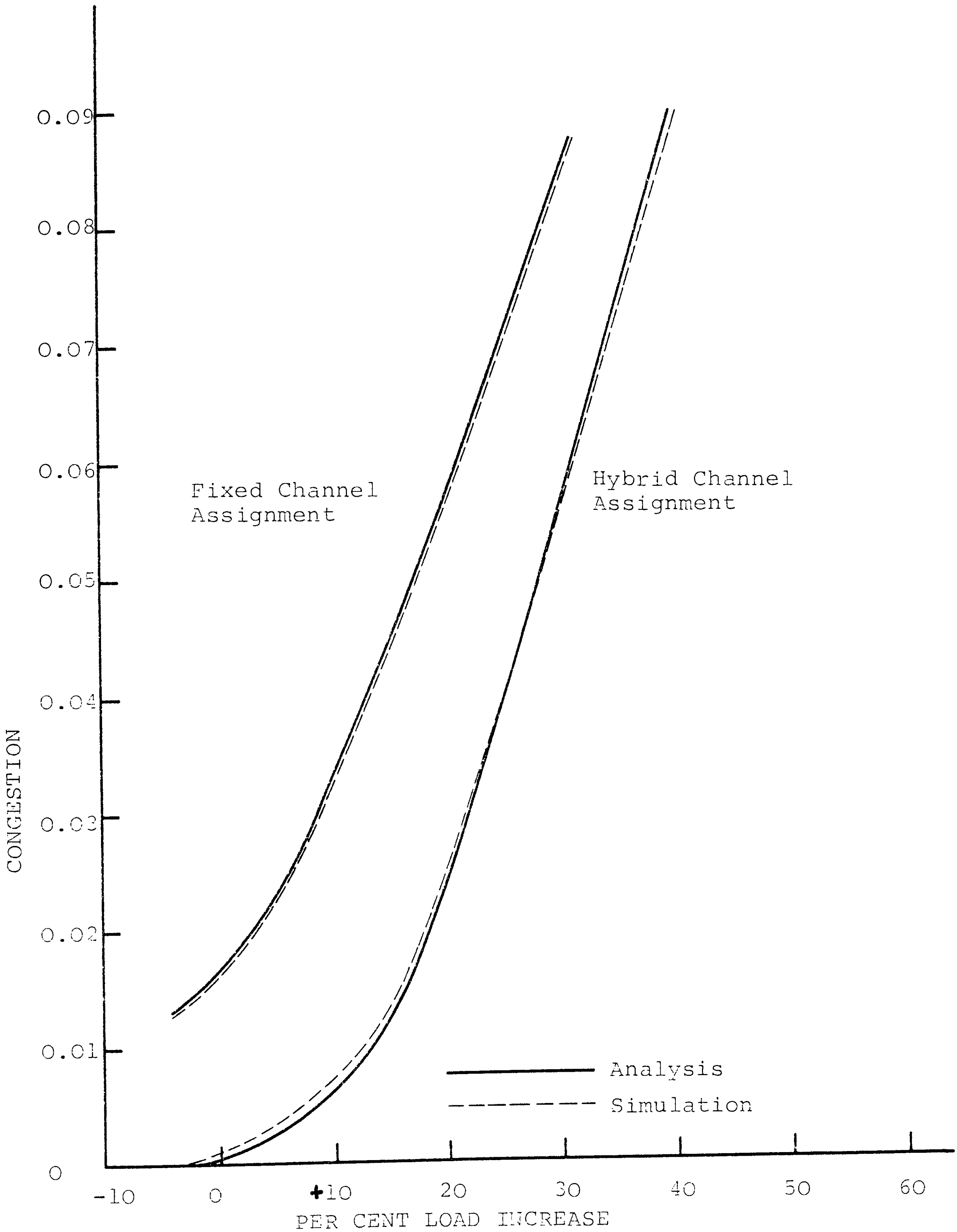

FIGURE (2.4)

COMPARISON BETWEEN THEORETICAL PREDICTED CONGESTION AND SIMULATION RESULTS AT LOW CONGESTION VALUES. BASIC LOAD=11.4 ERLANGS PER CELL; 18 CHANNEL PER CELL (FIXED CHANNEL ASSIGNMENT); 14 FIXED, 4 DYNAMIC CHANNELS PER CELL (HYBRID CHANNEL ASSIGNMENT).

channel assignment case. The traffic is expressed as a percentage increase over a base load of 11.4 erlangs. There is no significant disagreement over the range of traffic considered.

CHAPTER THREE

SIMULATION TOOLS

3. SIMULATION TOOLS

3.1 STRUCTURED PROGRAMMING FOR SIMULATION

Simulation using digital computers is widely used when it is required to design or study systems which are too complicated for theoretical analysis to give the necessary answers [63-71]. The teletraffic and control aspects of large scale mobile radio systems are far too complicated to be analysed completely by purely theoretical methods. Computer simulation is therefore required for dealing with such systems.

To simulate a system, it is necessary to have a suitable model. Sufficient detail needs to be included to provide accurate simulation results; however, excessive detail in the model is undesirable, as this leads to needless computation. A system model has three main features [63-65].

1. A description of the system structure and the required performance measures.
2. A description of the system activities.
3. Timing rules for the execution of the different activities and the accumulation of the performance measurements.

Programming a system simulation requires the system model to be transformed into computer language so that the computer correctly implements the model. The top-down approach to structured programming[71-76] may be used for the design of the simulation programs. This program-making method involves refining the description of the model in steps until a compilable program results. The algorithm in figure 3.1, represents this step-wise refinement process.

Two main advantages may be observed in the refinement process:

1. As the refinement progresses, the programmer will be concerned with only one limited problem at any time. This helps in the production of correct programs by easing the programming task. In addition, jumping statements can be avoided and readable straightforward programs are produced.

2. The resulting programs will have a modular construction. This provides the system designer with the ability to simulate other system configurations related to the model considered. Any required configuration may be simulated by the use of standard modules, while non-standard features require a minimum of programming effort[71].

The refinement process may be applied to any computer language required, whether it is a high-level problem-oriented language or a low-level machine-oriented language. However, extra complexity is involved in leading the refinement to

```
BEGIN
      Comment
            1.  Write a description of the required
                system in a modular form.
            2.  This description must be correct,
                though not executable by a computer.
      Invariant Comment
            The description in its current form is
            correct.
      WHILE
            Condition
            The description is not entirely executable.
      DO (
            Process
            1.  Take a module which is not executable.
            2.  Make a correct refinement of that
                module in a modular form.
          )
END
```

FIGURE (3.1)

PROGRAMMING SIMULATION IN A TOP-DOWN STRUCTURED PROGRAMMING APPROACH a low-level language, especially in the case where the system model is of a complicated nature. Therefore, the use of high-level languages is preferable. Some of these languages are specially devised for simulation[65,66]. However, these are usually designed to suit certain systems and may be restricted by certain rules. The high-level language Algol 68 is particularly suitable for the simulation of complicated systems[71,81]. This language is appropriate for the top-down approach, because of its provision for the definition and manipulation of multilevel data formats. The main disadvantage of Algol 68 is that it cannot be run on small computers. One suitable computer language that can be used for simulation on small machines is Coral 66[77-80].

Programs have been developed by the writer for the simulation of both speech traffic and computer networks using the top-down structured programming method, together with the computer languages Algol 68 and Coral 66. These simulation programs enable the designer of large scale mobile radio systems to investigate various teletraffic and control problems. The remainder of this chapter deals with the development of these simulation programs.

3.2 SIMULATION TOOLS FOR SPEECH TRAFFIC

3.2.1 Time-True Simulation

System Model

In time-true simulation, the state of the model duplicates, instant by instant, the state of the system being simulated[63]. A basic problem is that, in the real system, events may occur simultaneously, whereas the simulation can only deal with one event at a time. One solution is to define a time base which is advanced in increments. After the time base has been advanced, all the activities due to occur within the next time increment are simulated. The time base is then advanced and the process is repeated. In this section, the application of time-true simulation to a mobile radio system of a general structure is described.

For the purpose of simulation, mobile radio systems can be considered to consist of mobiles and channels. The mobiles originate and receive three types of call as described in chapter 2. These are network-to-mobile, mobile-to-network and mobile-to-mobile calls. Network-to-mobile and mobile-to-network calls each require one channel, whereas a mobile-to-mobile call requires two. Each type of call loads the system with two streams:

1. The call arrivals stream, which may have a certain statistical pattern. This stream can also be described in terms of the distribution of call inter-arrival time.
2. The call terminations stream, which similarly may have a particular statistical pattern and can be described in terms of the distribution of call

holding time or call service time.

A mobile radio system may be described in a top-down approach as shown in figure 3.2. A mobile in such a system may be structured in the following manner:

1. Mobile identity such as a telephone number.
2. Traffic interfaces to signal the arrival and the termination of calls.
3. Mobile state to indicate whether the mobile is free or busy.
4. Performance measurements to enumerate successful calls, mobile active time, blocked calls due to all channels being busy, and blocked calls due to the mobile being busy.

Similarly, a channel may be structured in the following manner:

1. Channel identity to facilitate monitoring various channel assignments.
2. Channel state.
3. Performance measurement to enumerate channel usage.

The basic system activity is the handling of calls for each mobile. This activity involves dealing with some other activities at lower levels such as the following activities, which may demonstrate the state of the system at different levels, as shown in figure 3.3:

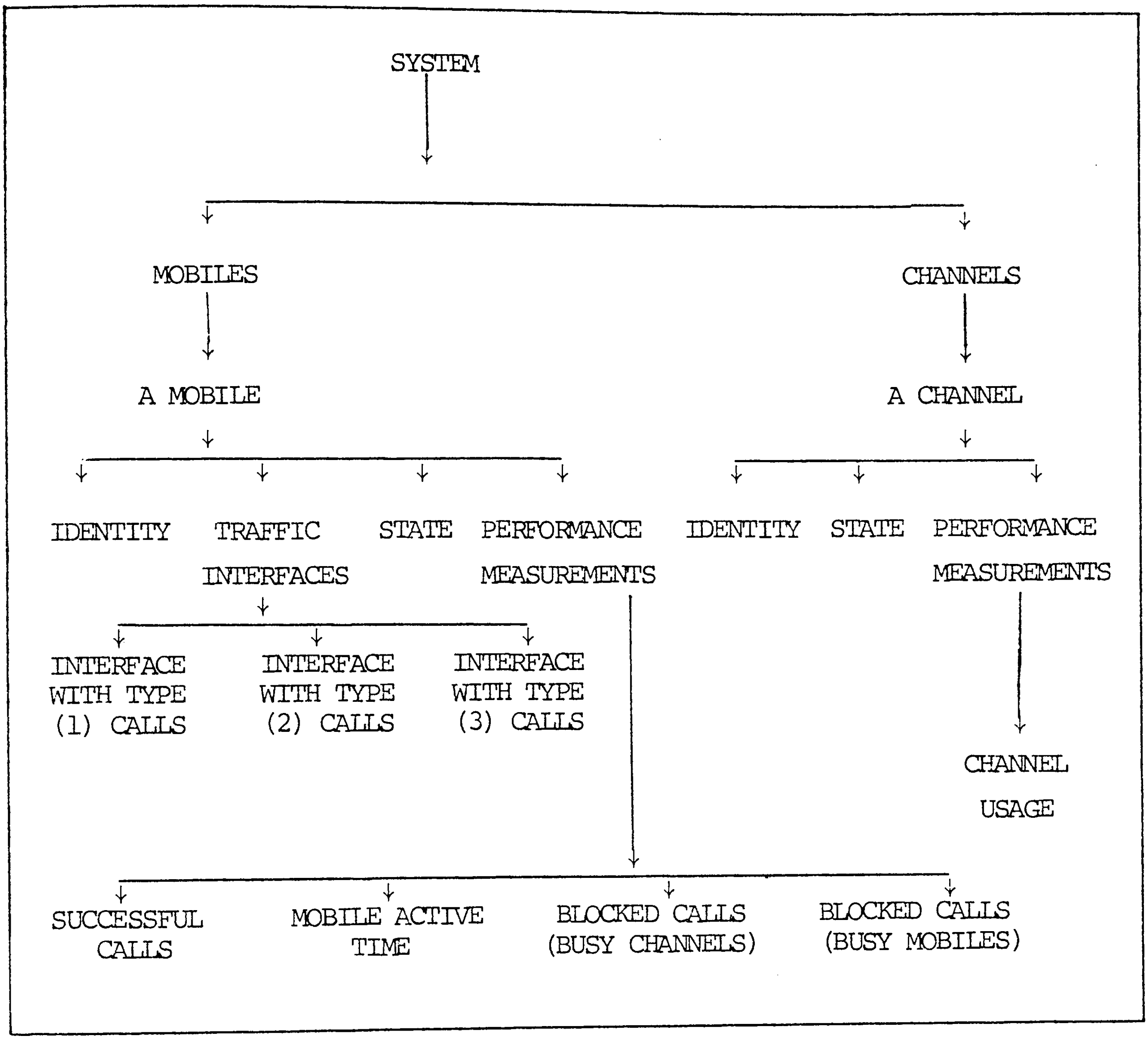

FIGURE (3.2)

LEVELS OF MOBILE RADIO SYSTEM DESCRIPTION FOR TIME-TRUE SIMULATION

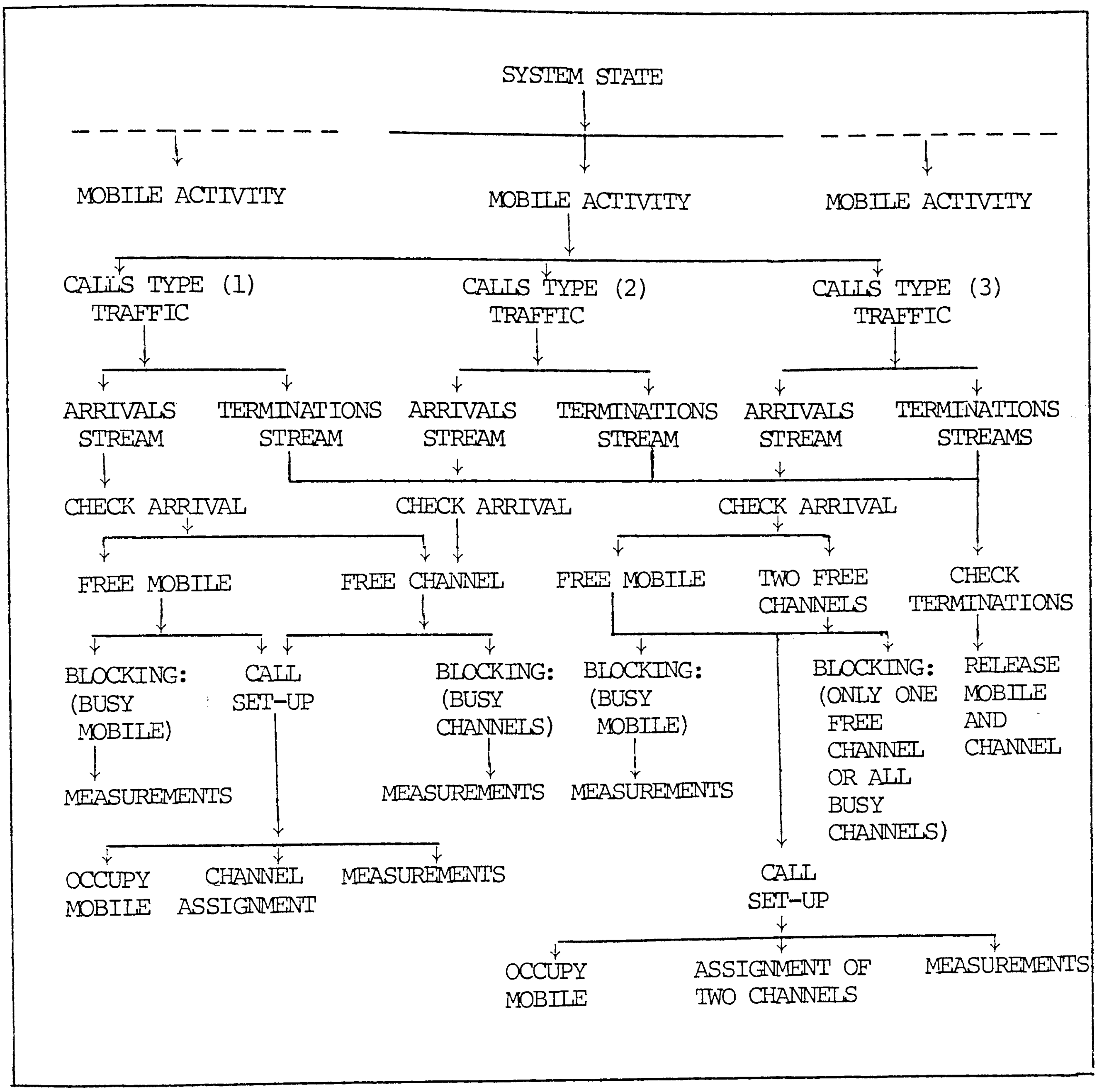

FIGURE (3.3)

LEVELS OF MOBILE RADIO SYSTEM STATE FOR TIME-TRUE SIMULATION

1. The application of both the call arrivals stream and the call terminations stream for each type of call.

2. The setting-up of calls by checking the state of the mobile and searching for free channels.

3. The termination of calls by releasing the concerned mobiles and channels.

4. The accumulation of performance measurements.

The basic time unit for the execution of the different activities for all mobiles is considered to be one second of true time.

Programming

The computer language Algol 68 has been used in programming the simulation. The compiler used was the Royal Radar Establishment Algol 68 compiler for International Computer Limited (ICL) 1900 series[83,84]. The programs were developed using an ICL 1904S machine[85].

Figure 3.4 shows the code which describes the system as presented in figure 3.2 .

Algol 68 provides for one level to be referred to, from a higher level by its *field selection facility*[81], as in mobile OF system. Field selection from one level to a non-adjacent lower level may become complicated. As an example, the following code would be required for the selection of the field '$time_1$' in the traffic interface of

```
COMMENT a system of m mobiles
        and n channels    COMMENT
INT m, n ;  Read ((m, n)) ;

MODE INTERFACE = STRUCT (INT time1, time2, time3, release
                         caller) ;
MODE MEASUREMENT = STRUCT (INT s1,s2,s3,b1,b2,b3,c1,c2,c3,
                           REAL us1, us2, us3) ;

MODE MOBILE = STRUCT (INT idn, chn,
                      INTERFACE interface,
                      MEASUREMENT measurement,
                      BOOL  state) ;

MODE CHANNEL = STRUCT (INT idn, mbℓ,
                       REAL usg,
                       BOOL  state) ;

MODE SYSTEM = STRUCT ([1:m] MOBILE mobile,
                      [1:n] CHANNEL channel) ;
SYSTEM system ;
```

FIGURE (3.4)

A MOBILE RADIO SYSTEM DESCRIPTION IN ALGOL 68
FOR TIME-TRUE SIMULATION

one of the mobiles in the system:

```
time1  OF (interface OF ((mobile OF system) [1]));
```

In such cases, it has been convenient to write procedures to perform such selections.

Before a simulation can start, fields within data structures such as identity of mobiles, state of mobiles and state of channels, need to be specified. In addition, performance measures must be set to a suitable initial state, usually starting with a zero value. Various procedures have been written to perform these initialisations.

Procedures for the simulation of different traffic streams have also been developed. Figure 3.5, shows an example of an Algol 68 procedure for generating exponentially-distributed random numbers of a specified mean value. The procedure is based on Knuth(87) and uses the standard Algol 68 procedure *random* which generates uniformly distributed random numbers, in the range from 0 to 1.

Procedures have been written for the simulation of the activities shown in figure 3.3, together with procedures for obtaining and reporting performance measurements. Figure 3.6 shows examples of four procedures to perform the following operations:

1. Check the arrival of a network to mobile call.
2. Check a free channel.

3. Check the blocking of all channels.

4. Accumulate channel usage.

```
PROC expran = (INT mean) INT :
  (
     REAL exp ;
      exp : = - ℓn (random) ;
     ENTIER (mean * exp + 1 )
  )  ;
```

FIGURE (3.5)

AN ALGOL 68 PROCEDURE FOR GENERATING EXPONENTIALLY DISTRIBUTED RANDOM NUMBERS

These examples illustrate how the programmer need only be concerned with a limited problem at a time.

The foregoing procedures operate at low levels. Figure 3.7 shows a procedure operating at the highest level which simulates the activity of the whole system. This procedure calls lower level procedures, which in turn call procedures from still lower levels, finishing with calls to procedures at the lowest level.

The time-true simulation described here can give a full picture of the behaviour of the system. Each mobile in the system may have its own traffic pattern which can

```
COMMENT check the arrival of a network-to mobile call  COMMENT
PROC tstgl = (MOBILE mbl, INT clock) BOOL :
    (IF time1 OF (interface OF mbl) = clock
    THEN TRUE ELSE FALSE FI);


COMMENT  check a free channel  COMMENT
PROC tstch = (CHANNEL chn) BOOL:

    (IF state of chn = TRUE
    THEN TRUE ELSE FALSE FI);



COMMENT check the blocking of channels COMMENT
PROCtstbl = ([ ] CHANNEL chn) BOOL:

    (INT k: = 0 ;
    FOR i TO UPB chn
    WHILE tstch (chn[i]) = FALSE
    DO k: = i ;
    IF k = UPB chn
    THEN TRUE ELSE FALSE FI ) ;
COMMENT accumulate channel usage  COMMENT
PROC chusg = (MOBILE mbl, REF CHANNEL Chn, INT clock) VOID

    (usg OF chn PLUS
    (release OF (interface OF mbl) - clock) ) ;
```

FIGURE (3.6)

EXAMPLES OF FOUR ALGOL 68 PROCEDURES AT LOW LEVELS FOR THE TIME-TRUE SIMULATION OF A MOBILE RADIO SYSTEM

```
COMMENT the function of a mobile-radio system COMMENT
PROC sysfn = (REF SYSTEM s,
              REAL rate1, rate2, rate3,
              INT hold1, hold2, hold3, clock) VOID:
(
  COMMENT dealing with call terminations COMMENT
  terminations (mobile OF s, channel OF s,
               rate1, rate2, rate3,
               hold1, hold2, hold3, clock) ;

  COMMENT dealing with call arrivals COMMENT
  arrivals (mobile OF s, channel OF s,
            rate1, rate2, rate3,
            hold1, hold2, hold3, clock)
) ;
```

FIGURE (3.7)

AN ALGOL 68 PROCEDURE AT THE HIGHEST LEVEL FOR THE TIME-TRUE SIMULATION OF A MOBILE RADIO SYSTEM be of any statistical distribution. In addition, performance measurements may also be enumerated for each individual mobile.

3.2.2 Roulette Simulation

System Model

In roulette simulation, a true-time scale is not kept. Instead, the probabilities of the various arrival and termination events are known and a pseudo-random number generator is used, in effect, to spin a roulette wheel, to decide which is the next event to occur[56,57]. The method is normally regarded as applying only for systems where the inter-call arrival times and the holding times both have negative exponential distribution. In this section, roulette simulation is applied to the simulation of a mobile radio system.

In a mobile radio system, the possible events during a small time interval dt are as follows:

event (1) : Network-to-mobile call termination.
event (2) : Mobile-to-network call termination.
event (3) : Mobile-to-mobile call termination.
event (4) : Network-to-mobile call arrival.
event (5) : Mobile-to-network call arrival.
event (6) : Mobile-to-mobile call arrival.
event (7) : No call arrival or termination.

These events are mutually exclusive. If $Q(j)$ is the probability of event (j), the following equation is satisfied[54-56]:

$$\sum_{j=1}^{7} Q(j) = 1$$

Consider a system with an infinite number of mobiles and consider A_1, A_2, A_3 are the traffic offered by the network-to-mobile, mobile-to-network and mobile-to-mobile calls respectively. If x_1, x_2, x_3 are the numbers of calls in progress for each of the three types of call, the probabilities for each event may be given as follows, assuming that the average holding time for all calls is the unit of time[56].

$$Q(1) = x_1 \, dt$$

$$Q(2) = x_2 \, dt$$

$$Q(3) = x_3 \, dt$$

$$Q(4) = A_1 \, dt$$

$$Q(5) = A_2 \, dt$$

$$Q(6) = (A_3/2) \, dt$$

$$Q(7) = 1 - \sum_{j=1}^{6} Q(j)$$

The random number generated by the conventional roulette method considers for each event a range of numbers proportional to the probability of the event[57]. A range of numbers for the event no call-arrival or termination is included. Ackroyd's roulette[69] discards this range.

This increases the efficiency of the roulette simulation especially at low levels of traffic. An evaluation of this increase is presented in section 3.2.3. The principle of Ackroyd's roulette is applied here. The probability of all events with either a call-arrival or a call-termination may be given as follows:

$$\sum_{j=1}^{6} Q(j) = (x_1 + x_2 + x_3 + A_1 + A_1 + \frac{A_3}{2})\,dt$$

If L is a random number generated within the following range:

$$0 < L \leqslant (x_1 + x_2 + x_3 + A_1 + A_2 + \frac{A_3}{2})$$

The ranges in which each of the following events occur may be given as follows:

event (1): $0 < \ell_1 \leqslant x_1$

event (2): $x_1 < \ell_2 \leqslant (x_1 + x_2)$

event (3): $(x_1 + x_2) < \ell_3 \leqslant (x_1 + x_2 + x_3)$

event (4): $(x_1 + x_2 + x_3) < \ell_4 \leqslant (x_1 + x_2 + x_3 + A_1)$

event (5): $(x_1 + x_2 + x_3 + A_1) < \ell_5 \leqslant (x_1 + x_2 + x_3 + A_1 + A_2)$

event (6): $(x_1 + x_2 + x_3 + A_1 + A_2) < \ell_6 \leqslant (x_1 + x_2 + x_3 + A_1 + A_2 + \frac{A_3}{2})$

Table 3.1, summarizes the above results by showing the probability of each event and the range of numbers that signals its occurrence.

The roulette method may also be applied to a mobile radio system with a finite number of M mobiles. If a_1, a_2, a_3

Event	Probability	Range
Network-mobile call termination	$x_1.dt$	$\ell_1 > 0$ $\ell_1 \leqslant x_1$
Mobile-network call termination	$x_2.dt$	$\ell_2 > x_1$ $\ell_2 \leqslant x_1+x_2$
Mobile-mobile call termination	$x_3.dt$	$\ell_3 > x_1+x_2$ $\ell_3 \leqslant x_1+x_2+x_3$
Network-mobile call arrival	$A_1.dt$	$\ell_4 > x_1+x_2+x_3$ $\ell_4 \leqslant x_1+x_2+x_3+A_1$
Mobile-network call arrival	$A_2.dt$	$\ell_5 > x_1+x_2+x_3+A_1$ $\ell_5 \leqslant x_1+x_2+x_3+A_1+A_2$
Mobile-mobile call arrival	$(\frac{A_3}{2}).dt$	$\ell_6 > x_1+x_2+x_3+A_1+A_2$ $\ell_6 \leqslant x_1+x_2+x_3+A_1+A_2+\frac{A_3}{2}$

TABLE (3.1)

THE RANGES OF A RANDOM NUMBER GENERATOR FOR THE ROULETTE SIMULATION OF A MOBILE RADIO SYSTEM WITH AN INFINITE NUMBER OF MOBILES are the traffic offered by each of the three types of call per free mobile and if x is the number of busy mobiles, Table 3.2 summarizes the results obtained for this system.

Figure 3.8 shows a top-down description of the system model considered by the roulette method. It may be observed that the performance measurements and the traffic interfaces here are at the system level, while they were considered at the mobile level in the time-true method.

Figure 3.9 illustrates the state of the system at different levels.

Programming

The computer language Coral 66 has been used in programming the simulation. The compiler used was the HPAC compiler(80) and the programs were developed on a 28K DEC PDP-11/03 machine. A pseudo-random number generator using a 24 bit feedback shift-register was used, interfaced via a DRV-11 parallel interface. The facility for addressing *absolute* addresses in Coral 66(77-79) was convenient in using this generator.

Figure 3.10 shows the code which describes the system as presented in figure 3.8. This code illustrates the way in which the system is interfaced with the hardware random number generator. Figure 3.11 shows a Coral 66 procedure for generating random numbers within a specified range. This is useful, since the range required for the roulette

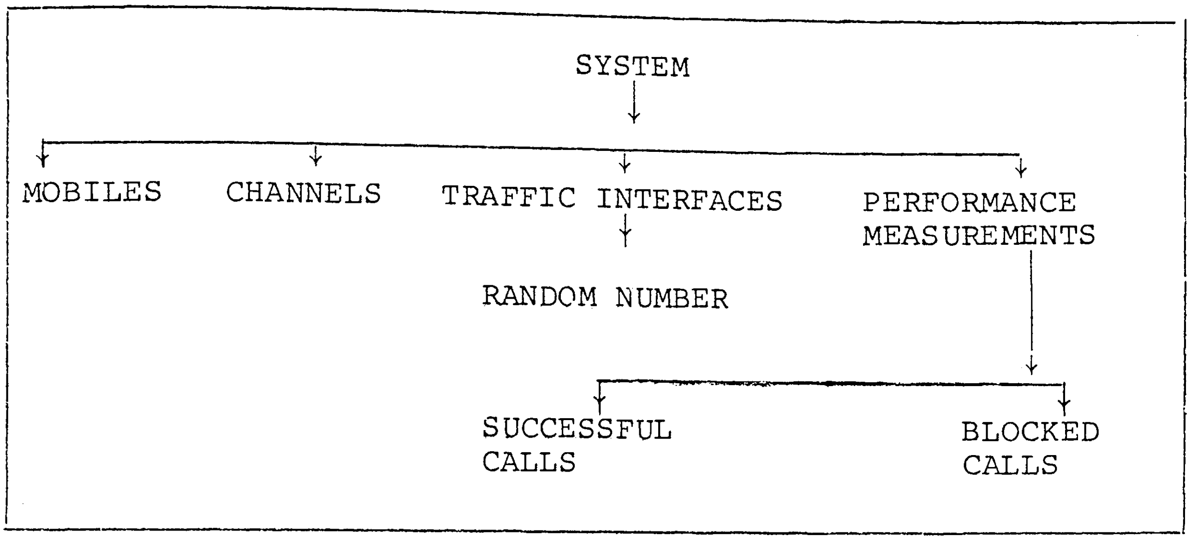

FIGURE (3.8)

LEVELS OF MOBILE RADIO SYSTEM DESCRIPTION FOR ROULETTE SIMULATION

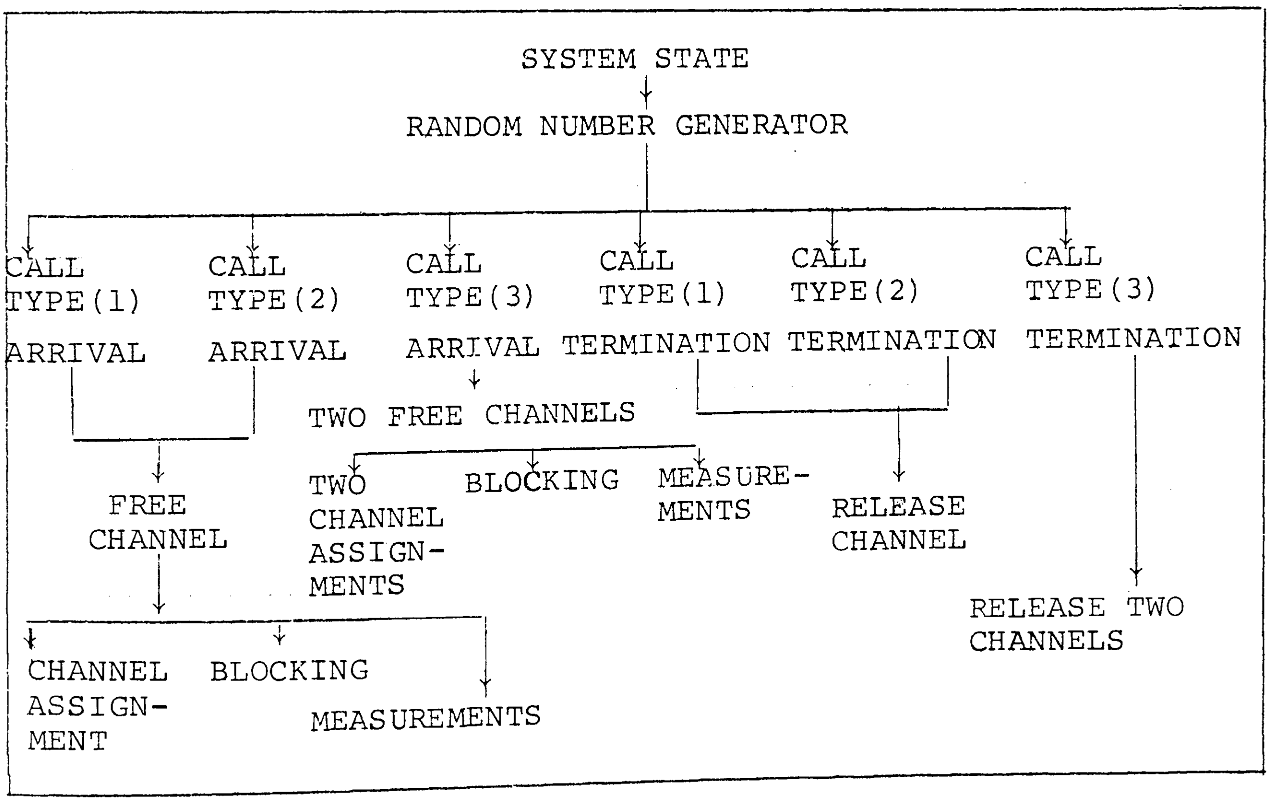

FIGURE (3.9)

LEVELS OF MOBILE RADIO SYSTEM STATE FOR ROULETTE SIMULATION

Event	Probability	Range
Network mobile call termination	$x_1.dt$	$\ell_1 > 0$ $\ell_1 \leqslant x_1$
Mobile-network call termination	$x_2.dt$	$\ell_2 > x_1$ $\ell_2 \leqslant x_1+x_2$
Mobile-mobile call termination	$x_3.dt$	$\ell_3 > x_1+x_2$ $\ell_3 \leqslant x_1+x_2+x_3$
Network-mobile call arrival	$(M-x).a_1.dt$	$\ell_4 > x_1+x_2+x_3$ $\ell_4 \leqslant x_1+x_2+x_3+(M-x).a_1$
Mobile-network call arrival	$(M-x).a_2.dt$	$\ell_5 > x_1+x_2+x_3+(M-x).a_1$ $\ell_5 \leqslant x_1+x_2+x_3+(M-x)(a_1+a_2)$
Mobile-mobile call arrival	$(M-x)(\frac{a_3}{2}).dt$	$\ell_6 > x_1+x_2+x_3+(M-x)(a_1+a_2)$ $\ell_6 \leqslant x_1+x_2+x_3+(M-x)(a_1+a_2+\frac{a_3}{2})$

TABLE (3.2)

THE RANGES OF A RANDOM NUMBER GENERATOR FOR THE ROULETTE SIMULATION OF A MOBILE RADIO SYSTEM WITH A FINITE NUMBER OF MOBILES

```
ABSOLUTE (INTEGER traffic/OCTAL (167774) ) ;
BEGIN
  INTEGER random ;
  COMMENT m mobiles and n channels,
          the value of n can be up to 100 ;
  INTEGER m, n ;
  INTEGER ARRAY channels [1: 100]
  COMMENT performance measurements ;
  INTEGER sc1, sc2, sc3, bℓ1, bℓ2, bℓ3;
```

FIGURE (3.10)

A MOBILE RADIO SYSTEM DESCRIPTION IN CORAL 66 FOR ROULETTE SIMULATION

```
INTEGER PROCEDURE rangn (INTEGER max) ;
BEGIN
  INTEGER i, j;
  i : = traffic ;
  FOR j : = 1, j+1
  WHILE i < = 0 OR i > max
  DO i : = traffic ;
  ANSWER i ;
END ;
```

FIGURE (3.11)

A CORAL 66 PROCEDURE FOR GENERATING RANDOM NUMBERS OF A SPECIFIED RANGE USING THE HARDWARE RANDOM NUMBER GENERATOR random numbers changes as the number of calls in progress changes.

Procedures have been written for the simulation of the activities shown in figure 3.9, together with procedures for the initialisation of performance measures ,clearing channels and reporting results. Figure 3.12 shows examples of two procedures at different levels for performing the following tasks:

1. The allocation of a free channel.
2. The arrival of a network-to-mobile call.

The roulette method has used a much simpler model than the time-true model. The method is useful for the insight it gives into the behaviour of systems of restricted form. A comparison between this method and the time-true method is presented in the next section. As an example of the simulation of the speech traffic of a mobile radio system with a finite number of mobiles, a Coral 66 program based on the roulette method is presented in Appendix II.

3.2.3 Roulette Versus Time-True Simulations

The roulette and the time-true simulations have been used to establish the accuracy of the teletraffic formulas derived in chapter 2. A mobile radio system of 200 mobiles and 20 channels has been considered as an example. In spite of differences of the two simulation methods, they produced the same results. The following compares the relative merits

```
COMMENT allocate a free channel ;

INTEGER PROCEDURE frchn (INTEGER ARRAY channel ;
                          INTEGER n) ;

    BEGIN
        INTEGER  i ;
        FOR i:=0, i+1
        WHILE channel |i| <> 0 AND i<=n DO;
        ANSWER i ;
    END ;


  COMMENT  a network-mobile call arrival;

  PROCEDURE   nmcall(INTEGER ARRAY channel ;
                     INTEGER m,n, x1,x2
                     x3, a1,constant,
                     ran, sc1, bl1 );


  BEGIN
         INTEGER max, min, x, i ;
         x:= x1+x2 + 2* x3 ;
         min:= (x1+x2+x3)* constant;
         max:= min + a1*constant ;
         IF ran > min AND ran < = max
         THEN BEGIN
                 i:= frchn ;(channel,n) ;
                 IF i < = n
                 THEN BEGIN
                        channel |i| :=1 ;
                        sc1:= sc1+1;
                        x1:= x1 + 1;
                      END

                ELSE
                      bl1:=bl1+1 ;

                END;
  END;
```

FIGURE (3.12)

EXAMPLES OF TWO CORAL 66 PROCEDURES AT DIFFERENT LEVELS FOR THE ROULETTE SIMULATION OF A MOBILE RADIO SYSTEM of the two approaches.

The Programming Task

The roulette method uses a much simpler model than the time-true method and so the programming is simpler. Thus Coral 66 on the PDP-11/03 is well matched to the task of programming the roulette simulation and Algol 68, with its more elaborate facilities, is well suited to the time-true method. The size of the programming task, in each case can be compared in terms of the number of lines of program text used : 200 for the roulette method, against 800 for the time-true.

Timing

The time-true simulation required 100 ms of computer time to simulate one second of true time, while the roulette simulation required only 10 ms per event. It seems surprising that the time-true simulation should run so sluggishly, especially when the much greater power of the 1904S is considered. The difference is due to two main factors:

1. The simplicity of the roulette method.
2. The use of a hardware random number generator on the PDP-11/03.

Efficiency

The time-true method can be inefficient, especially if events occur infrequently and yet a finely-divided time scale is used. Figure 3.13 shows how the efficiency, in terms of the proportion of useful time changes as the traffic level changes. The change of congestion is also illustrated.

The roulette method in its conventional form[57], is also inefficient at low traffic levels, since it considers a range of random numbers for the event of no-call arrival or termination. However, Ackroyd's roulette[69] is independent of traffic levels and has a 100 per cent efficiency in terms of the number of call-arrivals or terminations per roulette cycle. Figure 3.14 shows how the efficiency of the conventional roulette changes as the traffic level changes. This demonstrates how Ackroyd's concept increases the efficiency of the roulette method.

Cost

The PDP-11/03 costs about £10 per day to run including capital depreciation but excluding accommodation. Thus a run taking one minute costs a few pence. The cost on the 1904S for the 10 minutes to do the corresponding time-true simulation was estimated at about £12.

Insight

The roulette method is useful in giving insight into the behaviour of a system of restricted form. The time-true simulation allows much greater flexibility, so that models of unlimited generality can be simulated. For example, using the time-true method, it is possible to

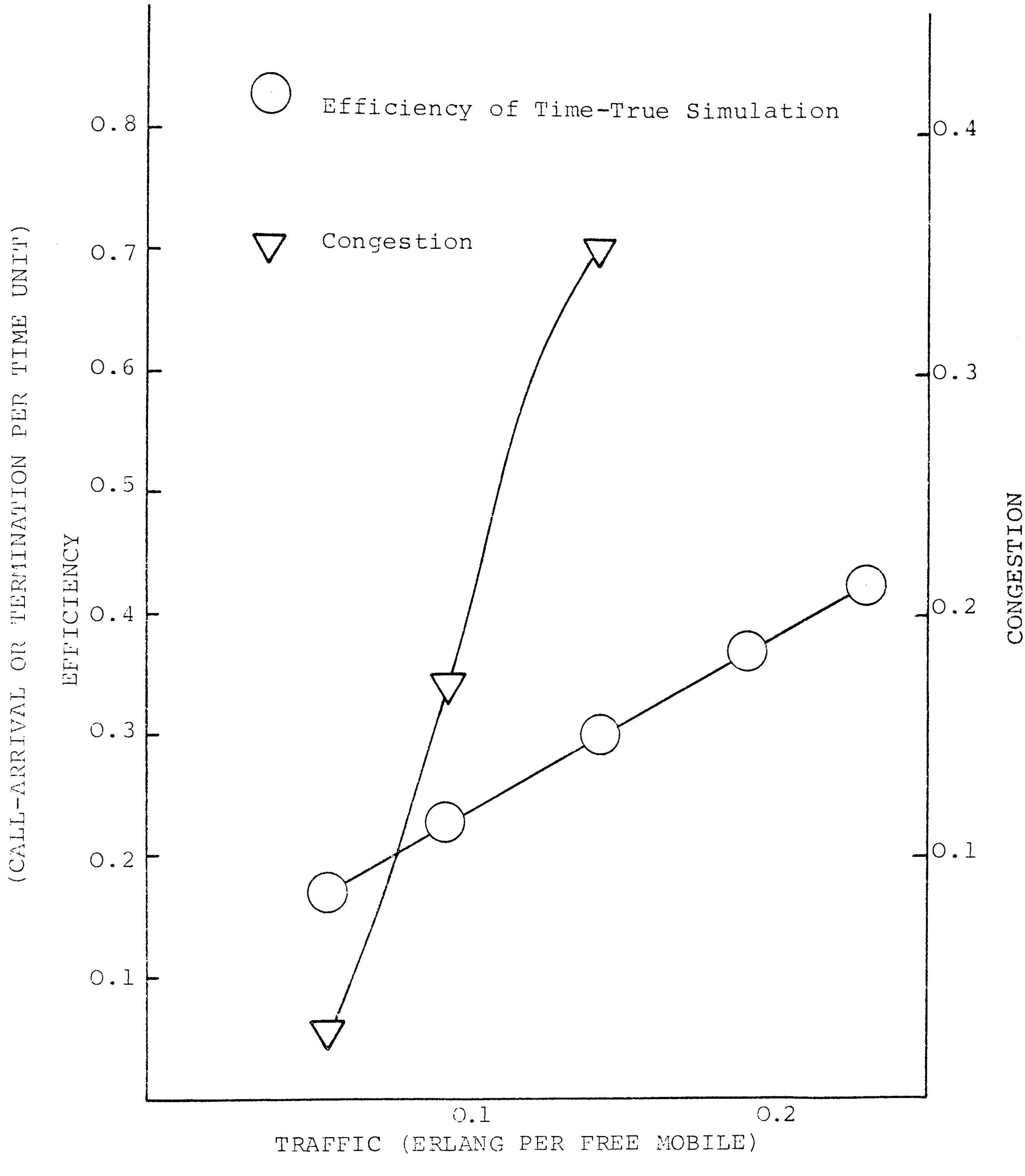

FIGURE (3.13)

EFFICIENCY OF TIME-TRUE SIMULATION VERSUS TRAFFIC OFFERED AND CONGESTION FOR A SYSTEM WITH 200 MOBILES AND 20 CHANNELS

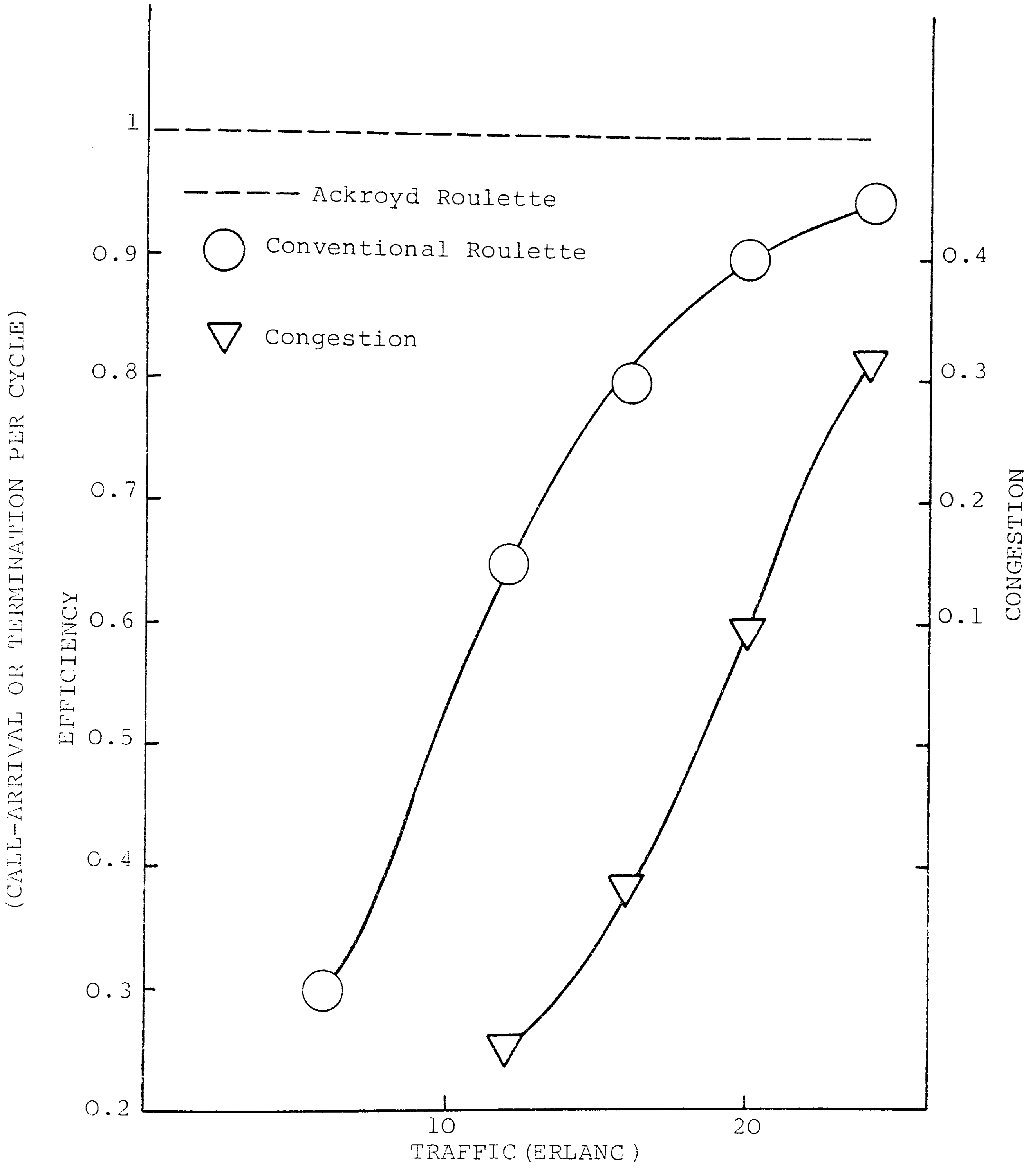

FIGURE (3.14)

EFFICIENCY OF ROULETTE SIMULATION VERSUS TRAFFIC OFFERED AND CONGESTION FOR A SYSTEM WITH 20 CHANNELS simulate the system with users whose individual patterns of use vary. In general, the roulette simulation is useful for quickly obtaining a crude understanding of the system behaviour. The time-true simulation can be extended to deal with more complicated systems than have been described here.

It may be concluded,therefore,that the time-true and the roulette method are complementary, rather than directly competitive.

3.3 SIMULATION TOOLS FOR COMPUTER NETWORKS

3.3.1 Design of Computer Networks

Future large-scale mobile radio systems are likely to use computer networks for the control of their operations. The controlling computers in such networks would communicate with each other via a land data network and with each mobile via a radio data link[25-29]. Such a structure represents a general form of computer communication networks[30-36].

The design variables of computer networks include the following[30,42,59]:

1. Network functions and design objectives.
2. Network topological structure.
3. Processing capacity and storage size.
4. Capacity of different channels.
5. Flow control policy.
6. Routing strategy.

In general, the design objectives of computer networks include the following:

1. Reliability : this may be evaluated in terms of the effectiveness of various network variables in dealing with communication errors and in keeping the network in operation when failures in different parts of the network occur.

2. Responsiveness : this may be measured in terms of the time required for a message to reach its destination.

3. Capacity : this may be evaluated in terms of maximum network throughput.

4. Cost : in many cases, network reliability, responsiveness and capacity have to be balanced within acceptable cost limits.

In the remainder of this chapter, models of general form of computer networks for the control of large scale mobile radio systems are described together with simulation tools for the investigations of such networks.

3.3.2 Models for Computer Networks

The general form of networks considered here consists of nodes interconnected by land and radio links, with each node having the form of a computer. Communication between nodes is assumed to operate on the basis of packet switching[30-33].

Data is conveyed by packets, each of which contains the following:

1. Control information, such as the destination address, the source address and the time of despatch.
2. The data to be sent.
3. Error detection and possibly error correction bits.

In the usual way[30], when a packet is received by a node, it is checked for errors. If correctly received it is acknowledged, and if addressed to another node, it is transmitted forward in an appropriate direction. Incorrectly received packets result in some suitable action, such as a request for retransmission.

For land links, it is of course, only possible for a single packet to be transmitted via a particular link at a given time[37-45]. With radio links, if a frequency is shared, it is possible for two or more packets accidentally to be transmitted simultaneously. The packets involved in such a *collision* will normally be received incorrectly. The Aloha technique[46-53], overcomes this problem by retransmission, after a random delay, of packets that have not been acknowledged. The simulations discussed here assume the use of the *slotted* Aloha technique[47], where the transmission of packets is synchronised to a common time frame.

Each node in the network would have its own central processor unit and memory, together with data communication interfaces. It is assumed that, for each transmission link, the node has two queues : one for arriving packets awaiting processing, and one for packets awaiting transmission. The maximum queue size may be specified. Figure 3.15 shows a node with three full-duplex ports.

Data traffic to be handled by the network is considered to originate at each node. Such traffic may have any statistical pattern[71], as the network model considered is independent of traffic patterns. This enables the system designer to investigate his model for any traffic stream required.

The driving stimulus of a central processor unit is the arrival of a complete packet in an input queue. It is assumed that a processor tests its input queues sequentially in a polling system[45]. As the processing required depends on the type of packet, different procedures may be needed for dealing with various packets.

In reality, the nodes of the network would operate concurrently. The simulation progress, being run on a single processor, can only deal with the nodes sequentially. A virtual machine approach[88] has been used to enable the parallel operation to be represented. To implement this, a basic time unit has been defined, taken to be the time

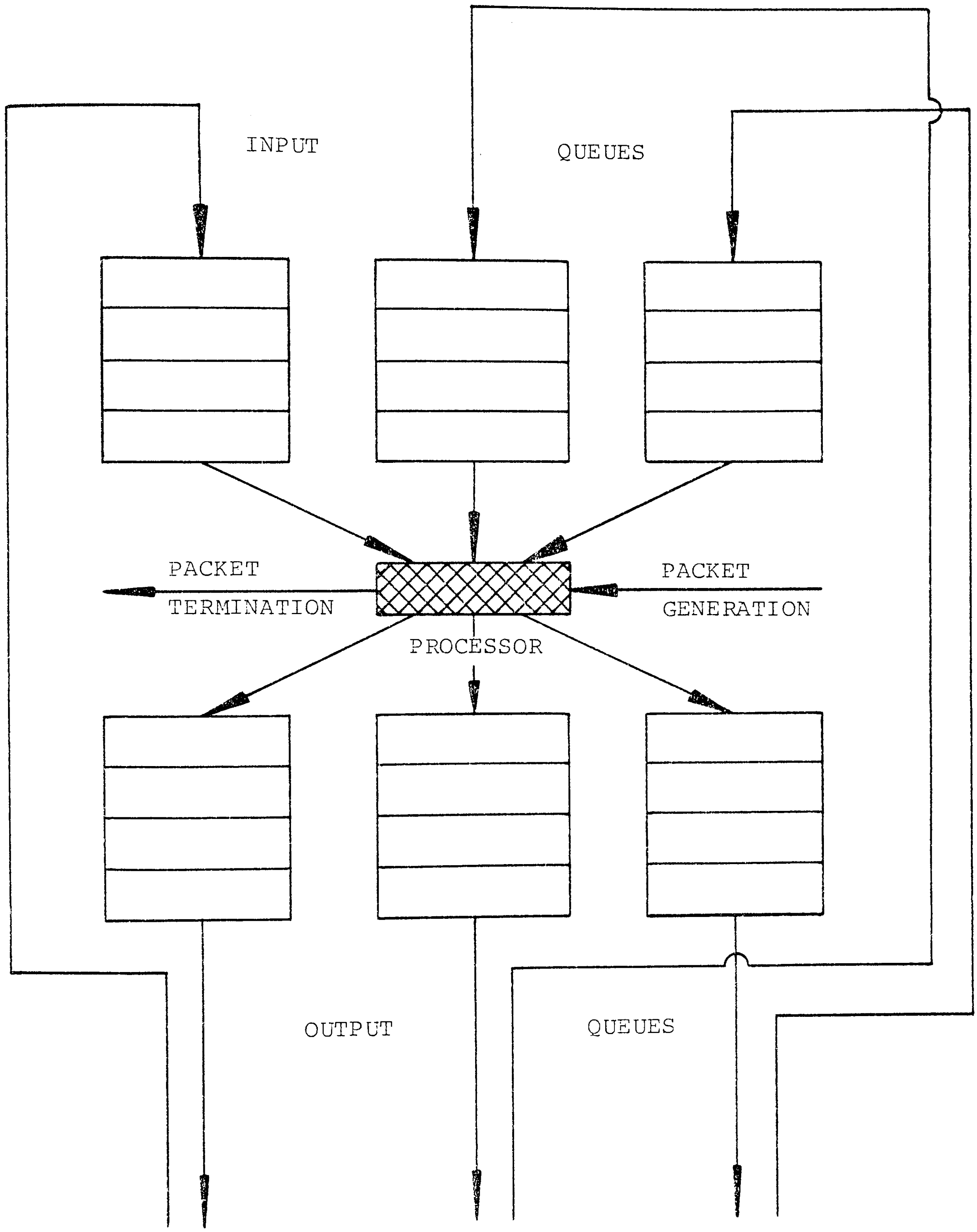

FIGURE (3.15)

SCHEMATIC DIAGRAM OF A PACKET SWITCHING COMPUTER for a packet to pass through the channel having the greatest capacity. All the operations of the simulation at one instant need to have been completed before a special clock is advanced, by the basic time unit, so that the simulation at the next instant can be carried out. This is the same principle used for the time-true simulation of the speech traffic of mobile radio systems. With this form of modelling, the timing of events can be observed, in the same way as on a real-life time-scale.

The form of network model described here enables various measures[34] to be evaluated during the simulation. These measures include channel utilization, node throughput, buffer storage size and responsiveness. For the control of mobile radio systems, responsiveness is the main design objective. For a required level of responsiveness, the value of the other measures can be predicted.

3.3.3 Simulation of Computer Networks

The top-down approach together with Algol 68 have been used for the development of simulation tools for programming the models described in the preceding section. The operation of a network simulation program can be divided into two aspects:

1. The description of the network structure and the required measures.
2. The description, as time progresses of the network

state and the accumulated performance measurements.

At the highest level of description, the network can be considered as a set of nodes, interconnected according to some specific topology. As mentioned earlier, each node is assumed to have buffer storage allocated to each link. While the location of each node may be specified symbolically, further top-down refinement is needed. In particular, the buffer storage size needs to be represented before the network can be represented completely in symbolic form. Each store can be represented as an array of pages, where each page provides accommodation for one packet. At the next level down, a packet is specified, for the purpose of simulation, by its source, destination, time of despatch and some other information. Figure 3.16 illustrates the various levels of the network description.

The network measurements required from the simulation correspond to any of the levels shown in figure 3.16. High level measurements, such as network capacity can be computed from measurements corresponding to lower levels.

Figure 3.17 shows how the network state may be viewed at different levels. There is correspondence between each of the levels of network state and each of the levels of network structure shown in figure 3.16.

At the highest level, the network state depends on the states of its nodes. The basic activities of each

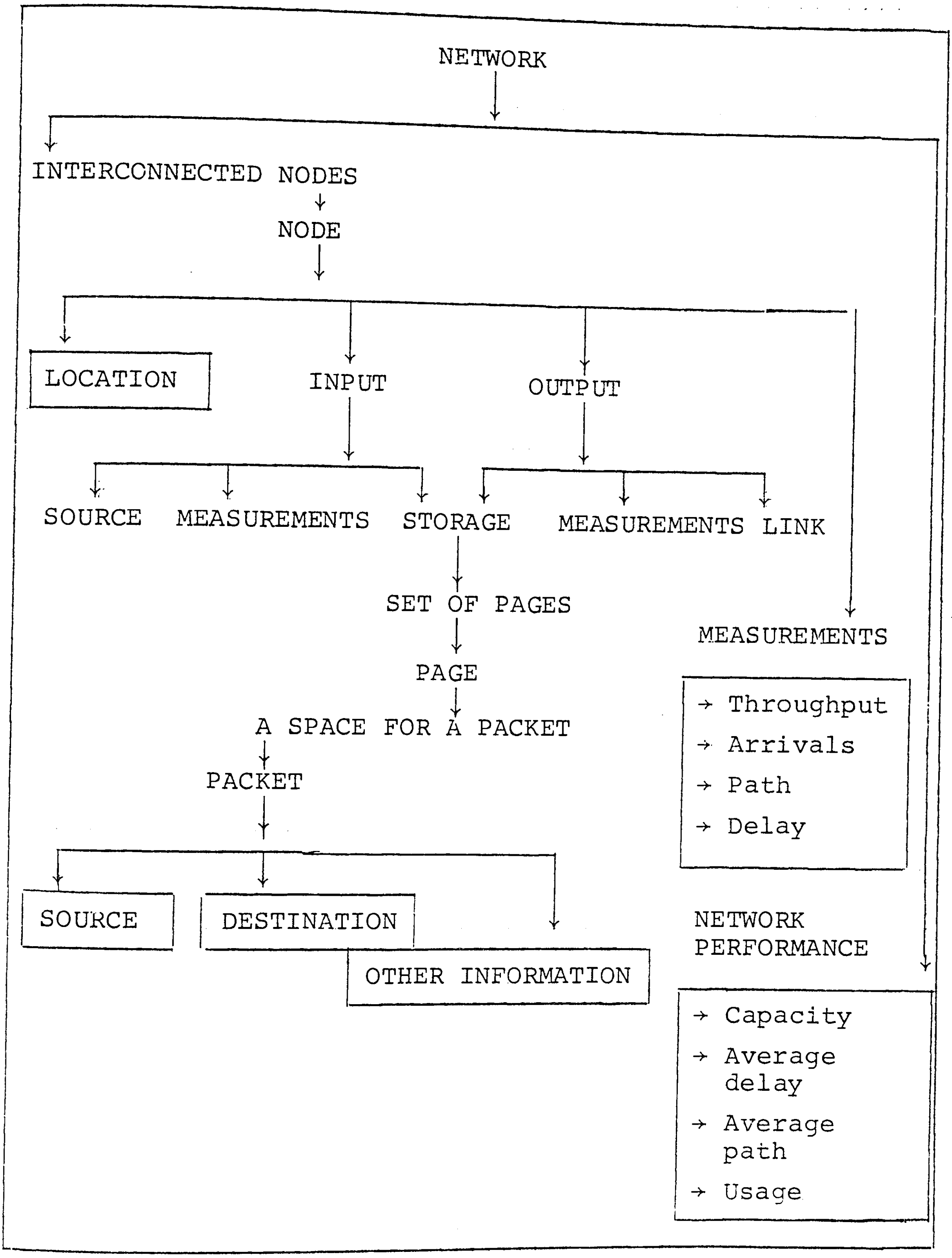

FIGURE (3.16)

LEVELS OF COMPUTER NETWORK DESCRIPTION

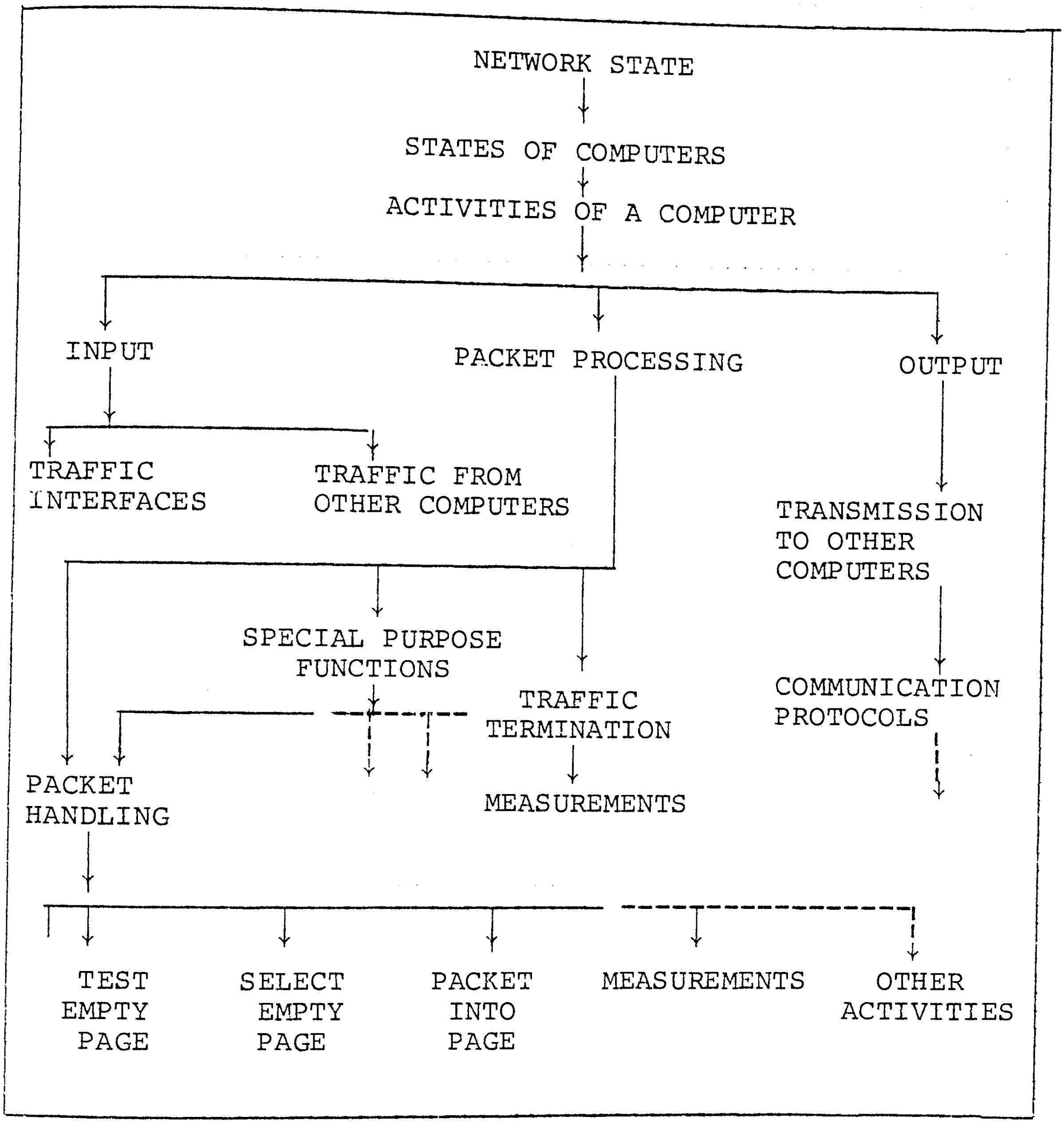

FIGURE (3.17)

LEVELS OF COMPUTER NETWORK STATE

node include the following:

1. Input and output activities, which include interfacing the node with the traffic sources and the transmission of packets to other nodes.

2. Packet processing, which include various activities at different levels. These activities include dealing with traffic terminations, the handling of packets and accumulation of performance measurements.

Figure 3.18 shows the code which describes the network as presented in figure 3.16.

The Algol 68 field-selection facility is even more useful here than it was when describing the time-true simulation of mobile radio speech traffic as the description of the network here involves more levels. A page in an input buffer of one of the nodes may be selected as follows:

```
Page : = (buffer OF (input of (computer OF network)[1])[2])[3]
```

Procedures have been written to perform different complicated selections.

The Algol 68 AS instruction has been used to simulate the storage of a packet in a page as follows:

```
page : = packet AS PAGE ;
packet : = page AS PACKET ;
```

An empty page in a buffer is denoted by:

```
page : = ("X", "X", "X", O, O, O, O);
```

```
COMMENT  n : number of computers
         ℓ : number of inputs and outputs per computer
         j : length of input queue,
         k : length of output queue    COMMENT

INT n, ℓ, j, k ; Read ((n, ℓ, j, k));
MODE PACKET = STRUCT (CHAR source, destination, transit,
                       INT  class, counter, time, clock);
MODE PAGE = STRUCT (CHAR ch1, ch2, ch3, INT cn1,cn2,cn3,cn4);
MODE INPUT = STRUCT (CHAR source [1: j] PAGE buffer,
                      INT active, blocked, idle);
MODE OUTPUT = STRUCT (CHAR link, [1:k] PAGE buffer,
                       INT active, blocked, idle);
MODE MEASUREMENT = STRUCT (INT throughput, arrivals,
                             path, delay) ;
MODE COMPUTER = STRUCT (CHAR name, [1:ℓ] INPUT input,
                         [1:ℓ] OUTPUT output,MEASUREMENT m);
MODE PERFORMANCE = STRUCT (REAL average delay,
                            average throughput, efficiency,
                            utilization);
MODE NETWORK = STRUCT ([1:n] COMPUTER computer
                        PERFORMANCE per) ;
NETWORK net ;
```

FIGURE (3.18)

COMPUTER NETWORK DESCRIPTION IN ALGOL 68

In a similar way to the simulation of speech traffic, fields within data structures such as names of computers, sources of inputs and destination of outputs, need to be specified before the simulation can start. In addition, the network must be initialized by being set to a suitable initial state - usually with all buffers empty. Various procedures have been written to do this.

Procedures have been written for the simulation of the states shown in figure 3.17, together with procedures for obtaining and reporting performance measures. Figure 3.19 shows examples of four procedures operating at the lowest level to perform the following:

1. Test a page for being empty.
2. Select the first empty page in a buffer.
3. Store a packet in the first empty page.
4. Accumulate the delay of an arriving packet destined to the node at which it has arrive.

Figure 3.20 shows a procedure operating at the highest level which simulates the activities of the whole network. This procedure operates in a similar way to the highest level procedure written for the simulation of mobile radio speech traffic. It calls lower level procedures, which in turn call procedures from still lower levels, finishing with calls to procedures at the lowest level, as illustrated in figure 3.19.

```
COMMENT test a page for being empty  COMMENT
PROC tstep = (PAGE page) BOOL :
(IF ch1 OF page = "X" AND ch2 OF page = "X" AND
    ch3 OF page = "X" AND cn1 OF page = 0 AND
    cn2 OF page = 0   AND cn3 OF page = 0 AND
    cn4 OF page = 0
THEN TRUE ELSE FALSE FI) ;
COMMENT select the first empty page in a buffer  COMMENT
PROC empag = ([]PAGE buffer)INT :
(INT t : = UPB buffer + 1 ;
FOR  ℓ FROM UPB buffer BY - 1 TO LWB buffer
WHILE tstep (buffer[ℓ])
DO t : = ℓ;
t ) ;


COMMENT put a packet in the first empty page  COMMENT
PROC pkipg = (PACKET packet, REF [ ]PAGE buffer) VOID :
    (buffer [empag (buffer)] : = packet AS PAGE ) ;
COMMENT accumulate packet delay  COMMENT
PROC pkdly = (PACKET packet) INT :
    (clock OF packet - time OF packet) ;
```

FIGURE (3.19)

EXAMPLES OF FOUR ALGOL 68 PROCEDURES OPERATING AT THE LOWEST LEVEL FOR THE SIMULATION OF A COMPUTER NETWORK

```
COMMENT the network function  COMMENT

PROC netfun = (REF NETWORK net,
               INT clock, REAL rate) VOID.
    (COMMENT poll all computers COMMENT
     FOR ℓ TO Upb (COMPUTER OF net) DO
     (COMMENT computer function COMMENT
      input( ( computer OF net) [ℓ] ,
             Upb (computer OF net). clock, rate) ;
      processing ((computer OF net) [ℓ] , clock) ;
      output (  (computer OF net) [ℓ], net, clock)
     ) ;
```

FIGURE (3.20)

AN ALGOL 68 PROCEDURE OPERATING AT THE HIGHEST LEVEL FOR THE SIMULATION OF A COMPUTER NETWORK

The simulation tools described in this section have been applied to the simulation of a possible computer network for the control of large scale mobile radio systems. The network and the simulation results are presented in the next chapter.

CHAPTER FOUR

COMPUTER CONTROL

4. COMPUTER CONTROL

4.1 CONTROL REQUIREMENTS

In large scale mobile radio telephone systems, mobiles originate and receive three types of call as described in chapter 2. These are as follows:

1. Land-network originated calls for mobiles.

2. Mobile-originated calls for land network subscribers.

3. Mobile-originated calls for mobiles.

Automatic control for the handling of such calls would be essential in view of two factors:

1. The requirement for a mobile user to be able to dial calls in the normal fashion, with privacy assured by the absence of human operators.

2. The complexity of the functions involved in the handling of calls.

The control functions for the handling of calls[23-29] may be divided into two categories; basic functions and complementary functions. The basic functions would be required for both single-cell and multi-cell mobile radio systems. The complementary functions would be required for the handling of some of the calls in multi-cell systems.

The basic control functions include the following:

1. Call request. This function initiates the process of setting-up calls by informing the controlling system of the call originator and destination.

2. Paging. This function determines whether a mobile for which an incoming call is destined is available to receive that call.

3. Channel assignment. The role of this function is to locate an available voice channel and assign it to a mobile originating or receiving a call. For a mobile-to-mobile call, this function has to be performed twice. In the case where a free voice channel cannot be located, the call is considered to be blocked.

4. Ringing. This function indicates to a mobile that an incoming call has arrived and is ready to be received.

5. Call termination. The role of this function is to inform the controlling system that a call has terminated. As a result the voice channel or channels assigned to such a call would be freed.

The complementary control functions include the following:

1. Hand-over. In a multi-cell system, as an active mobile moves from one cell to another, it is necessary for the call to be transferred from the base station of the old cell to the base station of the new cell. The hand-over function involves the assignment of a channel to the call via the new base station and the release of the channel used by the old base station.

2. Address-changing. It is required that every mobile would have its own telephone number. As a mobile moves from one area or city to another, changing its real address, it is necessary to keep the telephone number which represents the virtual address of the mobile unchanged. The address-changing function involves informing the controlling system of the changes in the real address, so that calls addressed to the invariant virtual address can be routed to the mobile changeable real address.

The automatic control of a cellular mobile radio system must be able to perform the above control functions. Such control can be provided by a network of interconnected computers. The remainder of this chapter describes a possible structure for the control network and shows how such a structure may handle the control functions required.

Computer simulation of the proposed network for the investigation of its performance is also presented.

4.2 A PROPOSED CONTROL NETWORK

A computer network, for the implementation of the control functions required for the handling of calls in cellular systems is described here. As mentioned in chapter 3, there are usually many different options available to the designer of such networks. The choice of the overall network structure is one important factor[42]. Figure 4.1 shows a hierarchical structure of a possible control network for a region around a major city. The structure has three levels;

1. The cell level.
2. The cellular system unit level.
3. The city level.

At the cell level, each cell would have a base station or *cell node* consisting of automatically controlled equipment for communicating with mobiles by speech and data, a control processor and interfaces to data and speech links with other nodes. Each cell may use a number of radio channels for the speech transmission and has one full-duplex radio data channel, which by Aloha [46-53] packet switching, handles the whole data traffic for the cell. In addition, every mobile would have, a processor for data transmission

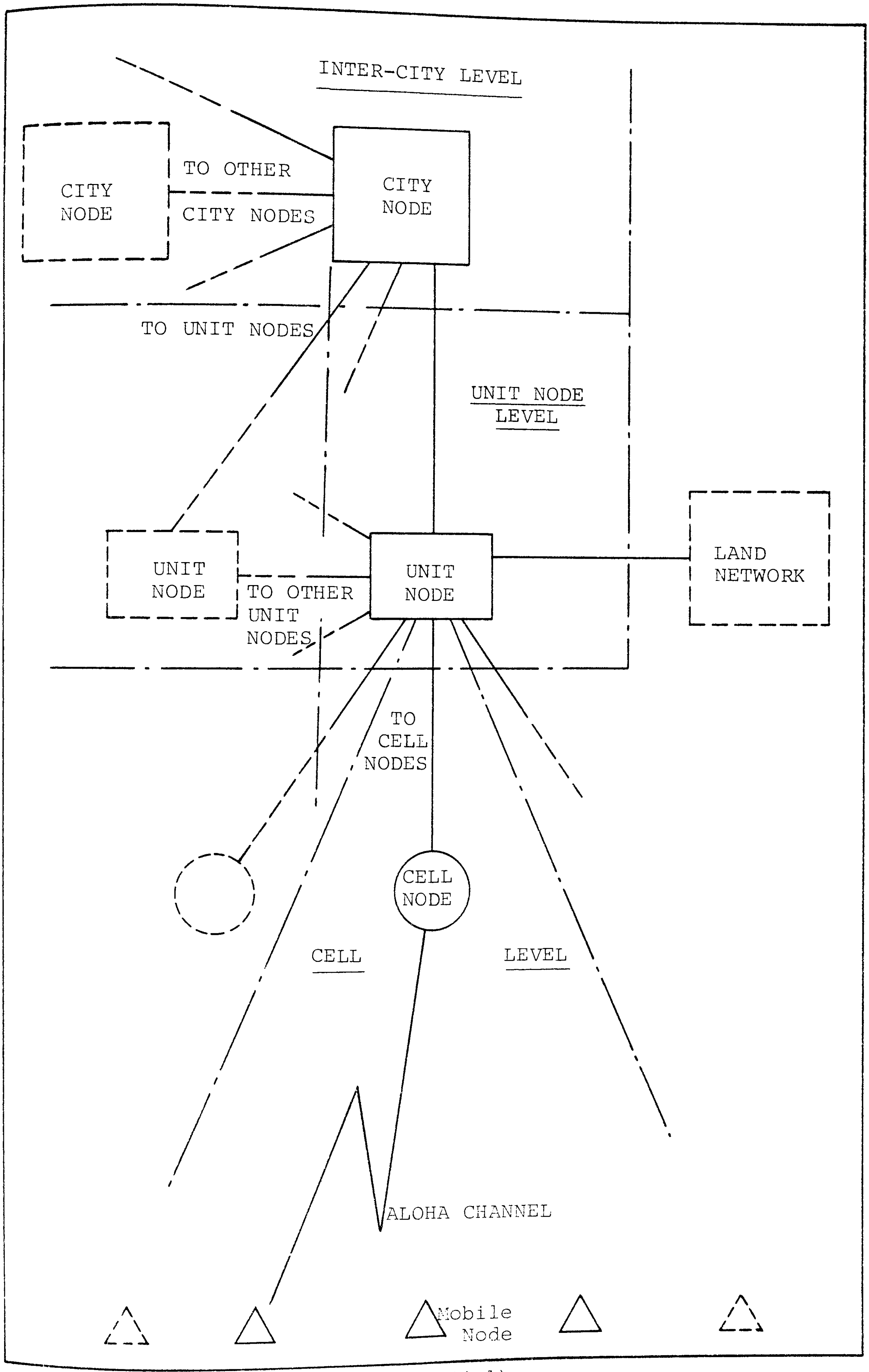

FIGURE (4.1)

THE STRUCTURE OF A CONTROLLING NETWORK FOR CELLULAR MOBILE RADIO incorporated with its speech communication equipment.

As mentioned in chapter 1, a cellular system unit is the smallest area in which all the frequency channels allocated for mobile use can be used simultaneously. A system unit consists of three cells, in the case where the frequency reuse buffering area is one belt of cells, and it consists of seven cells, if the buffering area is two belts of cells. The control network structure shown in figure 4.1 considers that the cellular system is divided into a number of system units, with each systems unit having a *unit node*. Each unit node would consist of telephone switching equipment and a control processor. A unit node would be connected to the following:

1. The cell nodes belonging to it.
2. The neighbouring unit nodes.
3. The land telephone network.
4. The *city node*.

The city node would incorporate a control processor connected by land data channels to other city nodes. These nodes may be of any number and widely spread over a large geographical area. In choosing the degree of connectivity between the various city nodes, reliability and cost of communications must be balanced. One topological structure that may balance the two factors is shown in figure 4.2.

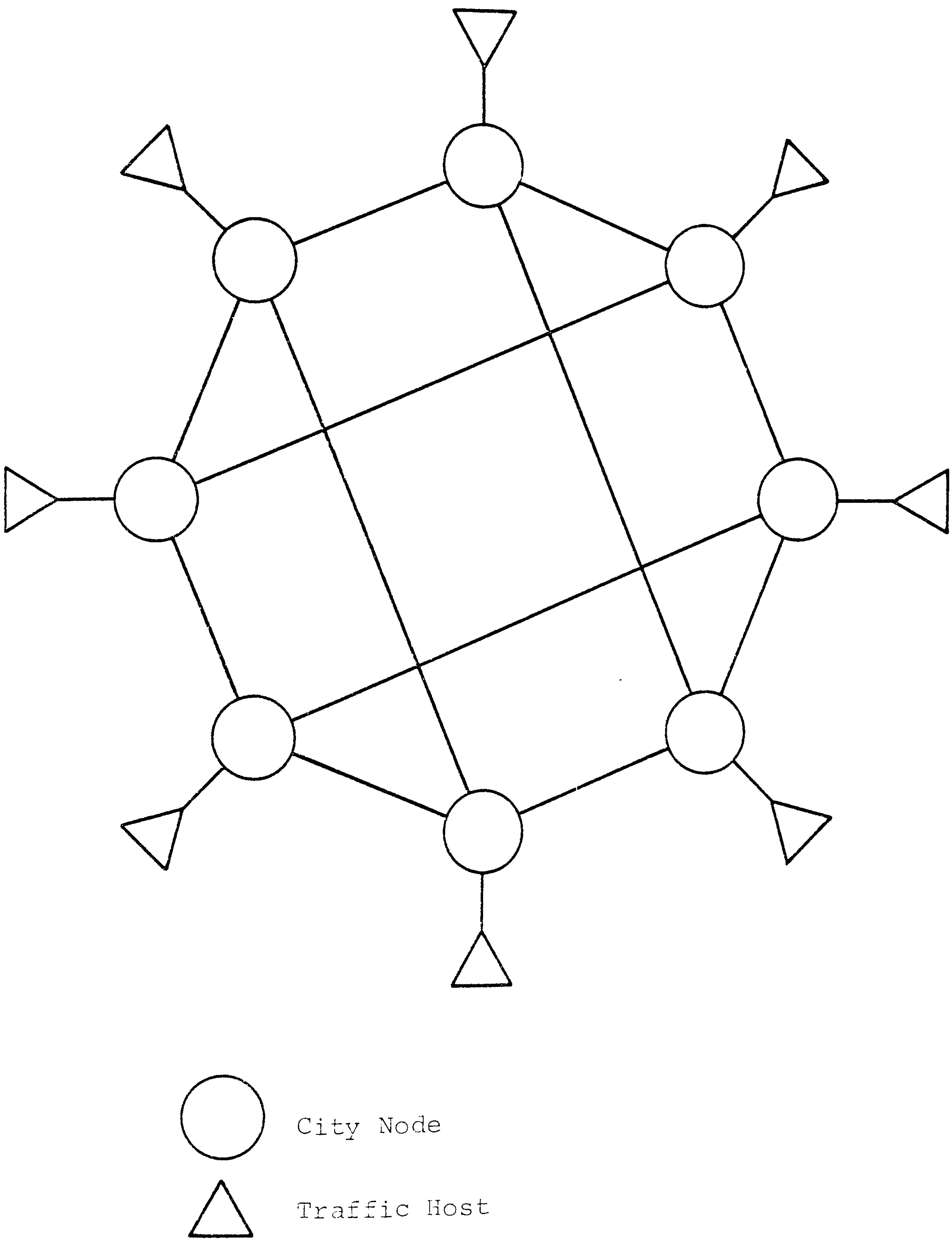

FIGURE (4.2)

THE STRUCTURE OF A POSSIBLE CONNECTIVITY
BETWEEN CITY NODES

In this structure, every node is linked with three other nodes, so that alternative routing is possible while the expensive full connectivity structure is avoided.

The hierarchical structure of the proposed network provides flexibility in constructing the network for different requirements. For single-cell systems, the network would consist of a cell node linked to the land telephone network with no unit nodes or city nodes needed. For a multi-cell system in which no inter-city communication is required, the network would consist of two levels only, the cell level and the unit level. New cells and system units may be built and incorporated with other systems. In addition, cities with new mobile radio systems may be linked to other cities via an inter-city network.

4.3 DATA TRAFFIC FOR CONTROL REQUIREMENTS

The basic control functions which deal with the setting-up and clearing-down of calls are one source of data traffic in the control network. Another source of traffic is the complementary control functions which deal with the handling of mobiles as they move between cells and travel between cities. The data traffic generated by the two sources depend on the method in which the control network implements the control functions. One such method is presented in this section. The method is described for three cases of complementary nature and increasing complexity, as follows:

1. Single-cell systems.

2. Multi-cell systems with all the mobiles remaining within their cells.

3. Multi-cell systems with free movement for all mobiles.

Hypothetical data traffic streams have been derived for each case using certain assumptions. As the cases considered are of complementary nature, the assumptions made have the same property. The resulting data streams have been used for the simulation of the proposed control network described in the previous section.

Single-Cell Systems

The operations that may be initiated for the implementation of the control functions required in single-cell systems, are described here as sequences of events for handling each of the three types of call. For a land-to-mobile call the sequence of events may be as follows:

1. The call is routed to the cell node through the land telephone network.

2. The cell node would then transmit a paging message over the radio data channel assigned to it, searching for the called mobile. The mobile, after recognising its page, has to acknowledge its availability.

3. The cell node selects a free radio voice channel and sends a channel assignment message to the mobile, which tunes its radio to the voice channel assigned to it.

4. The cell node sends a ringing message to the mobile, which generates an alerting signal to the user.

5. When the call is completed, the mobile sends a call termination message to the cell node, which frees the radio voice channel in use.

For a mobile-to-land call, the following sequence of events may apply:

1. The caller picks-up the phone in his car and dials the number of the land network subscriber he requires. The processor incorporated in the mobile stores the dialled number and when dialling is completed, sends a call request message to the cell node.

2. The cell node selects a free voice channel and informs the mobile of its channel assignment. The mobile tunes its radio to the voice channel

assigned to it.

3. The land network deals with routing the call to the land subscriber and activates its ringing device to call its attention.

4. When the call is completed, the mobile reports that to the cell node, which frees the radio voice channel in use.

A mobile-to-mobile call may be handled in a similar way to the above sequences of events. It may be observed that, such a call requires two free voice channels to be assigned, one for the calling party and another for the called party.

The events described above may generate different data traffic streams. One hypothetical data model has been derived under the following assumptions:

1. Each event generates a message at its source and transmits it to its destination.

2. All messages take the form of equal size packets.

3. All mobiles originate and receive equal calling rates. This applies to each of the three types of call.

Table 4.1 shows the data traffic streams resulting from the above assumptions, expressed in terms of the rate of packet transmission per mobile for the various basic control functions. The terms r_1, r_2 and r_3 represent the arrival rates of land-to-mobile, mobile-to-land and mobile-to-mobile calls respectively.

Function / Transmission	Call Request	Paging	Channel Assignment	Ringing	Termination	Total
Cell Node to Mobile	-	(r_1+r_3)	$(r_1+r_2+2r_3)$	(r_1+r_3)	-	$(3r_1+r_2+4r_3)$
Mobile to Cell Node	(r_2+r_3)	-	-	-	$(r_1+r_2+2r_3)$	$(r_1+2r_2+3r_3)$

Table 4.1

RATE OF PACKET TRANSMISSION PER MOBILE IN A SINGLE-CELL MOBILE RADIO SYSTEM

Multi-Cell Systems with all Mobiles Remaining in their Cells

In a multi-cell system in which no mobile leaves its cell, the control requirements would involve the basic control functions only, as no hand-over or address-changing would be required. The handling of calls in such systems differs from the handling of calls in single-cell systems. The points of differences are presented in the following.

A cell node here would not be linked to the land telephone network. Instead, it would be linked to a system unit node, which would be linked to the land telephone network. A system unit node would supervise the operations of its cell nodes, in such a way that a cell node would operate on instruction messages from its system unit node.

Calls in multi-cell systems may be originated by mobiles and addressed to other mobiles which may be situated in different system units. At the cell level, such a call may be considered as a mobile originated call for a network subscriber in one cell, and as a network originated call for a mobile in the other cell. At the system unit level, a call of this type may be routed from one unit node to another via the land telephone network, as if it is a normal call between two fixed telephone stations.

System unit nodes determine the assignment of radio voice channels to mobiles, as these nodes maintain tables indicating the locations of mobiles and the state of the frequency channels. If the multi-cell system uses the fixed-channel assignment scheme, each unit node would be able, by using the information in its own tables, to select a free voice channel for assignment. However, if the dynamic or the hybrid channel assignment scheme is used, a unit node would have to acquire information from its neighbouring unit nodes before selecting the voice channel to be used.

In order to derive data traffic streams for multi-cell systems with all mobiles remaining within their cells, the following assumptions have been considered:

1. The data traffic derived for single-cell systems apply here at the cell level.

2. All messages passing through the network at all levels take the form of equal size packets.

3. As unit nodes supervise the operations of cell nodes, packets transmitted by mobiles to cell nodes are passed to the unit nodes concerned. In addition, packets sent by cell nodes to mobiles are originated by unit nodes and transmitted to cell nodes.

4. All cells have the same number of mobiles and generate equal calling rates.

5. The system may use different channel assignment schemes.

Using the above assumptions, table 4.2 results, showing the rates of packet transmission per mobile through different parts of the control network for the various control functions. The terms used in the table include r_1, r_2 and r_3 which as before represent the calling rates of the three types of call per mobile. Other terms in use include q and d which represent respectively the number of cells in a system unit and a factor for expressing the effect of the channel assignment in use on data transmission between unit nodes. If the fixed channel assignment is used the factor d would be zero, as no messages between unit nodes on the state of the voice channels would be required. For other assignment schemes d would depend on the amount of information that has to be interchanged between system units for performing the assignment.

Function / Transmission	Call Request	Paging	Channel Assignment	Ringing	Termination	Total
Unit Node to Unit Node	-	-	$(2.d.q)$ $(r_1+r_2+2r_3)$	-	-	$(2.d.q)$ $(r_1+r_2+2r_3)$
Unit Node to Cell Node	-	(r_1+r_3)	$(r_1+r_2+2r_3)$	(r_1+r_3)	-	$(3r_1+r_2+4r_3)$
Cell Node to Unit Node	(r_2+r_3)	-	-	-	$(r_1+r_2+2r_3)$	$(r_1+2r_2+3r_3)$
Cell Node to Mobile	-	(r_1+r_3)	$(r_1+r_2+2r_3)$	(r_1+r_3)	-	$(3r_1+r_2+4r_3)$
Mobile to Cell Node	(r_2+r_3)	-	-	-	$(r_1+r_2+2r_3)$	$(r_1+2r_2+3r_3)$

Table 4.2

RATE OF PACKET TRANSMISSION PER MOBILE FOR THE VARIOUS BASIC CONTROL FUNCTIONS

Multi-Cell Systems with Free Movement for all Mobiles

In multi-cell systems in which mobiles are free to move between cells and travel between cities, the control requirements would involve the hand-over and the address-changing functions, together with the basic control functions.

Two types of hand-over function would be needed: a low-level hand-over transfers a call between two cells, each under the control of the same unit node and a high-level hand-over transfers a call from one cell to another, each controlled by a different unit node. The sequence of events for performing a low-level hand-over function may be as follows:

1. The radio signal strength of each active mobile would be monitored by the cell node, so that it is possible to detect when such a mobile has passed between cells.

2. When such a transition is detected, the cell node reports that to its unit node.

3. The unit node selects the new voice channel to be assigned to the mobile via the new cell node. A message is then transmitted to both the new and the old cell nodes informing them of the new channel assignment.

4. The new cell node sends a channel reassignment message to the mobile, while the old cell node releases the old voice channel assignment.

The handling of a high-level hand-over function differs from the handling of a low-level one, in view of the following:

1. The old unit node would have to inform the new unit node of the hand-over.

2. The new unit node would have then to perform the new channel assignment while the old unit node would be dealing with the release of the old assignment.

Two types of address-changing function would be needed: a low-level address-changing transfers the address of a mobile between cells in a given city and a high-level address-changing transfers the address of a mobile between different cities. The sequence of events for performing a low-level address-changing function may be as follows:

1. A mobile may transmit its identification number at intervals. This number would be picked up at the nearest cell node.

2. If the mobile is a newcomer to the cell, the

cell node would transmit a message to its unit node informing it of the mobile's arrival. The unit node would then update its location table by adding the identification of the new mobile linked with its corresponding cell.

3. If the previous location of the mobile was in a cell belonging to the same system unit, the unit node would omit the previous address of the mobile from its location table. In the case where the new address of the mobile belongs to another system unit, the new unit node would inform the old unit node of the new address. The old unit node would then be able to direct calls coming to the mobile through the land network to the new unit node.

The handling of a high-level address-changing function differs from the handling of a low-level one, as the high-level function requires the following extra actions:

1. When a unit node receives the identification of a mobile arriving from a different city, it transmits a message to the city node informing it of the newcomer.

2. The city node would send a message through the inter-city network to the old city node, which

would in turn inform the old unit-node of the mobile's new address.

The data traffic streams derived for the case of multi-cell systems with all mobiles remaining within their cells are considered to be applicable here for the setting-up and clearing-down of calls. For the handling of the hand-over and address-changing functions, data traffic streams have been derived under the following assumptions.

1. All the assumptions used for deriving data streams for the basic control functions are considered to be true here for the complementary control functions.

2. All mobiles transmit their identification number at equal rate.

3. The proportion of mobiles leaving a given cell in a given direction is equal to the proportion of mobiles arriving at the cell in the opposite direction.

4. All cities have equal number of system units.

The data traffic streams resulted from the above assumptions are presented in table 4.3 for the hand-over functions and in table 4.4 for the address-changing functions. The terms used in table 4.2 are also in use in the present tables together with the following new terms:

ℓ_1 = the proportion of mobiles leaving a given cell to other cells within a system unit.

ℓ_2 = the proportion of mobiles leaving a given cell to other cells in other systems units.

ℓ = the sum of ℓ_1 and ℓ_2.

e = the proportion of mobiles leaving a given cell to cells in other cities.

r = the rate at which a mobile transmits its identificaiton number.

n = the number of system units in a city.

Function / Transmission	Low-level Hand-over	High-level Hand-over	Total
Unit Node to Unit Node	$(2.d.q.\ell_1)$ $(r_1+r_2+2r_3)$	$(2.d.q.\ell_2+2.q.\ell_2)$ $(r_1+r_2+2r_3)$	$(2.d.q.\ell+2.q.\ell_2)$ $(r_1+r_2+2r_3)$
Unit Node to Cell Node	$(2.\ell_1)$ $(r_1+r_2+2r_3)$	$(2.\ell_2)$ $(r_1+r_2+2r_3)$	$(2.\ell)$ $(r_1+r_2+2r_3)$
Cell Node to Unit Node	ℓ_1 $(r_1+r_2+2r_3)$	ℓ_2 $(r_1+r_2+2r_3)$	ℓ $(r_1+r_2+2r_3)$
Cell Node to Mobiles	ℓ_1 $(r_1+r_2+2r_3)$	ℓ_2 $(r_1+r_2+2r_3)$	ℓ $(r_1+r_2+2r_3)$

Table 4.3

RATE OF PACKET TRANSMISSION PER MOBILE FOR THE HAND-OVER FUNCTIONS

Transmission \ Function	Low-level Address-changing	High-level Address-changing	Total
Network to City	-	(n.q.e.r)	(n.q.e.r)
City to Network	-	(n.q.e.r)	(n.q.e.r)
City to Unit	-	(q.e.r)	(q.e.r)
Unit to City	-	(q.e.r)	(q.e.r)
Unit to Unit	$(2.q.\ell_2.r)$	-	$(2.q.\ell_2.r)$
Cell to Unit	$(\ell.r)$	(e.r)	
Mobile to Cell	r		r

Table 4.4

RATE OF PACKET TRANSMISSION PER MOBILE FOR

THE ADDRESS-CHANGING FUNCTIONS

4.4 SIMULATION OF PROPOSED NETWORK

Programs for the simulation of the proposed control network have been developed using the simulation tools presented in chapter 3. In the following, the simulation programs are described and simulation results are presented.

The Simulation Programs

The simulation has been carried out in three parts. Each part corresponds to one of the three hierarchical levels of the network as described in the following.

The first part matches with the cell level. It deals with the radio data network between a cell node and the mobiles in its vicinity termed the *cell network* in the following. The network may have any given number of mobile nodes transmitting packets to the cell node over a shared radio channel. A second radio channel is used by the cell node to transmit packets and acknowledgements back to the mobile nodes. The transmission of packets over the shared radio channel is based on the slotted Aloha technique[46,47]. The network has in addition to the traffic streams over the shared radio channel, a traffic interface with the rest of the controlling network as shown in figure 4.3.

The second part of the simulation matches with the system unit level. It deals with the data network between a unit node and the traffic hosts linked to it as illustrated in figure 4.4. This network is referred to as the *system-unit network*. The structure of the network is a *star structure*. The network may have any given number of input-output ports. 10 ports are required in the case where the frequency reuse buffering area is one cell wide. More ports are required for larger buffers.

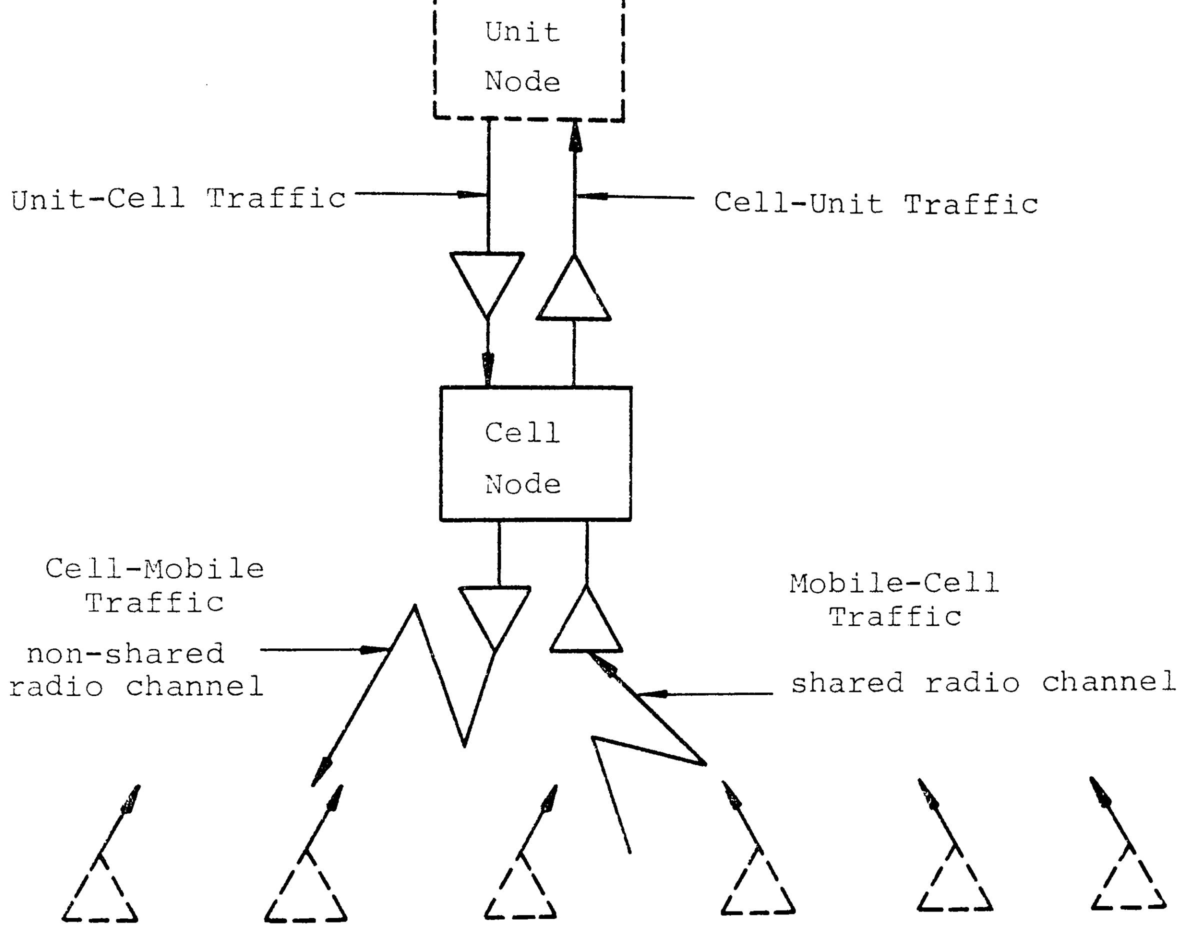

FIGURE (4.3)

THE STRUCTURE OF A 'CELL NETWORK'

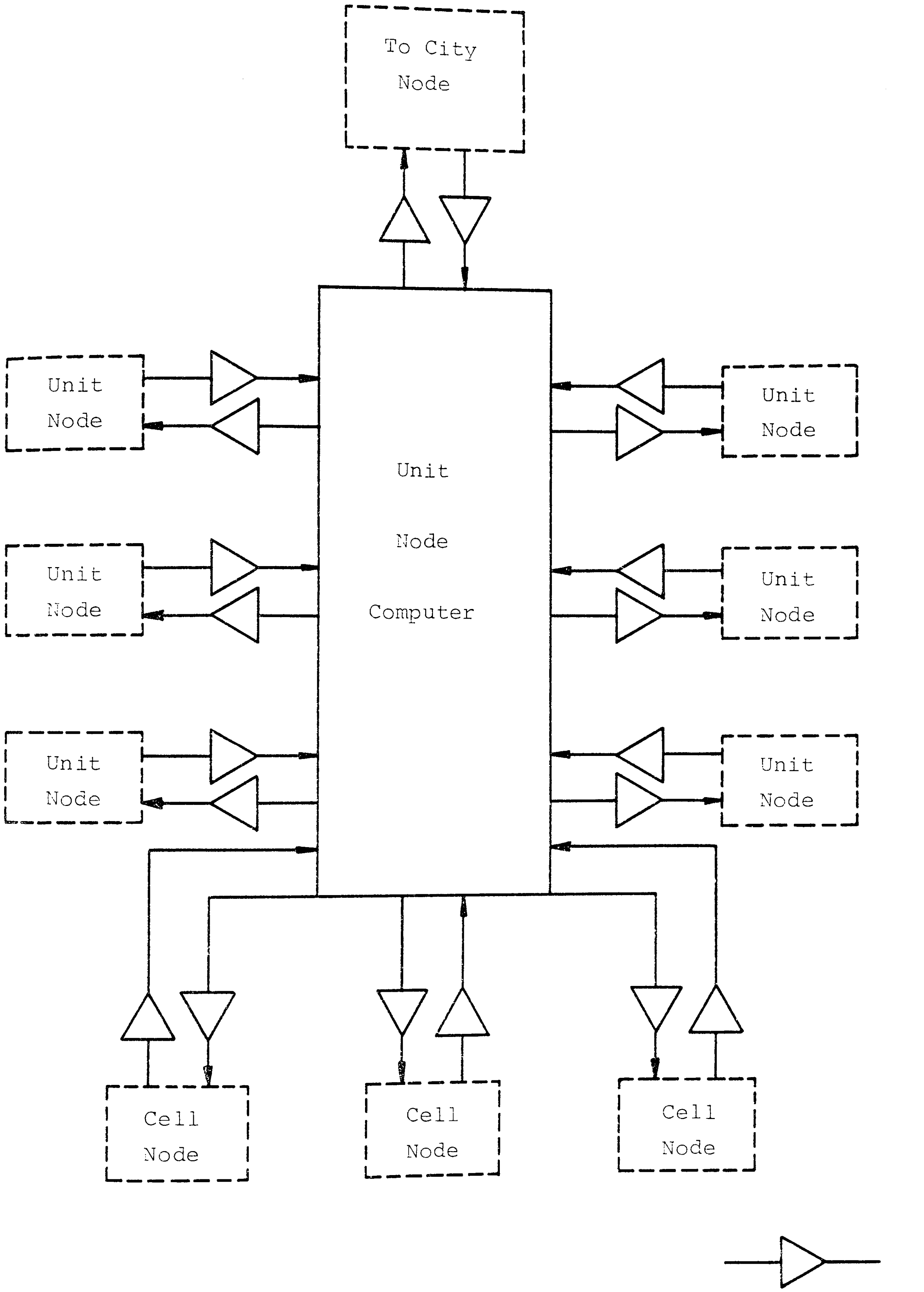

FIGURE (4.4)

THE STRUCTURE OF A 'SYSTEM UNIT NETWORK'

The third part of the simulation matches with the city level. It deals with the data network between cities termed the *inter-city network*. The network may have any given even number (greater than two) of city nodes with each node connected to three other city nodes as suggested in section 4.2. The traffic hosts in the inter-city network are the unit nodes linked to each city node via a packet concentrator as shown in figure 4.5.

All nodes are considered to have the same basic structure shown in figure 3.15, with each traffic interface having an input and an output buffer store for the queuing of packets. The sizes of the buffers are made sufficiently large that a buffer never fills completely during a simulation.

In a similar way to the models considered by the simulation tools presented in chapter 3, the central processor of each node tests its input buffers sequentially and processes the waiting packets. All packets are assumed to acquire equal treatment in terms of priority and processing. In addition, it is considered that an acknowledgement is generated on the processing of each task required by a packet. An acknowledgement is assumed to have the same form and size as a packet.

The simulation programs are of modular construction. The standard modules described in chapter 3 are used by each of the three parts of the simulation. In addition,

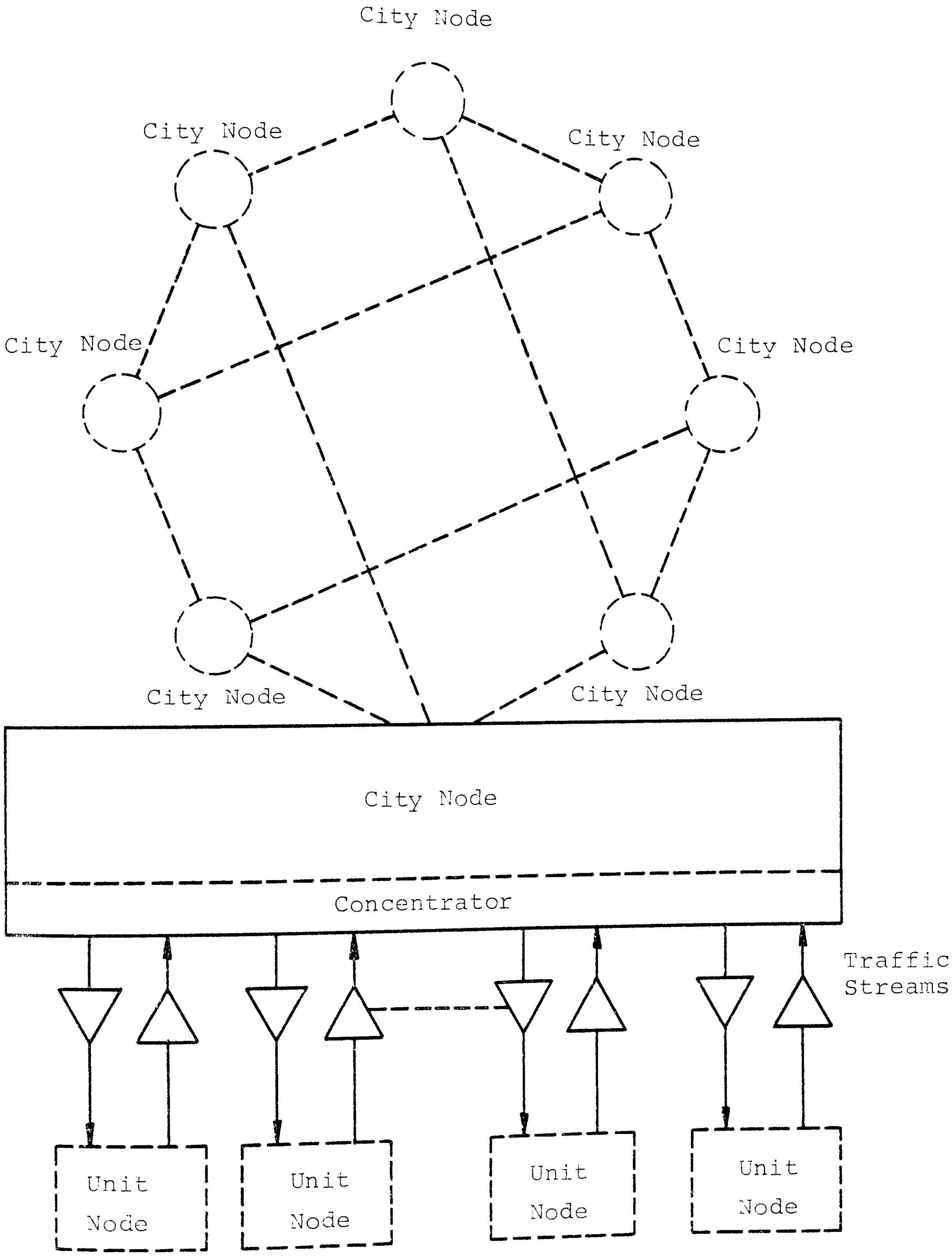

FIGURE (4.5)

THE STRUCTURE OF AN 'INTER-CITY NETWORK' special purpose modules have been developed for each part independently. In order to test that each module performs its operation successfully, the modules have been run on the computer in stages starting with the modules at the lowest level and moving up to the highest level.

In order to demonstrate the utility of the simulation programs, the programs have been run in two stages:

1. The aim of the first stage is to provide an insight into the behaviour of each part of the simulation of the control network independently. Each part in this case was interfaced with an independent packet generation stream. Various performance measurements are presented for this case.

2. The aim of the second stage is to investigate the capacity of the control network in handling the various control requirements. For this purpose, a typical example of a control network is considered. The only traffic stream applied to the simulation programs in this case is the random generation of calls, where each call results in a specific sequence of packet transmissions as described in section 4.3.

In the remainder of the this chapter, results obtained from the two stages are presented. As an example of the structure of the simulation programs, the inter-city network simulation program is listed in Appendix II.

Simulation of the Cell Network

The simulation results presented here are obtained by running the cell network part of the simulation independently. In this case each mobile node is considered to be interfaced with a stream of packets generated at random with exponentially distributed intervals of time (Poisson traffic). The time unit is assumed to be the time taken for a packet to pass through the shared channel. The performance measurements enumerated include the following:

1. Channel traffic; this is the average number of packets transmitted over the shared radio channel during a time unit. This traffic consists of both newly generated packets as well as retransmissions of previously collided packets.

2. Channel throughput, this is the average number of packets transmitted successfully over the shared channel.

3. Average packet delay, this is the average length of time between the generation of a packet and the reception of an acknowledgement indicating that the packet has been successfully received.

4. Efficiency of transmission; this is the proportion of successful transmissions to the total over the shared radio channel.

The performance measurements have been determined by varying the rate of the traffic offered. The results are as follows:

1. The channel throughput is plotted against channel traffic in figure 4.6. The results agree closely with Abramson's analytical results for slotted Aloha systems[53]. This agreement provides more confidence in the accuracy of the simulation results.

2. In figure 4.7, the average packet delay is plotted against channel traffic. It is shown that delay increases as the traffic increases. The reason for this is that in such a case more retransmissions would be required, which causes longer delay.

3. The efficiency of transmission is plotted against channel traffic in figure 4.8. As expected it decreases continually as the traffic increases.

Simulation of the System-Unit Network

The simulation results presented here are obtained by running the system-unit network part of the simulation

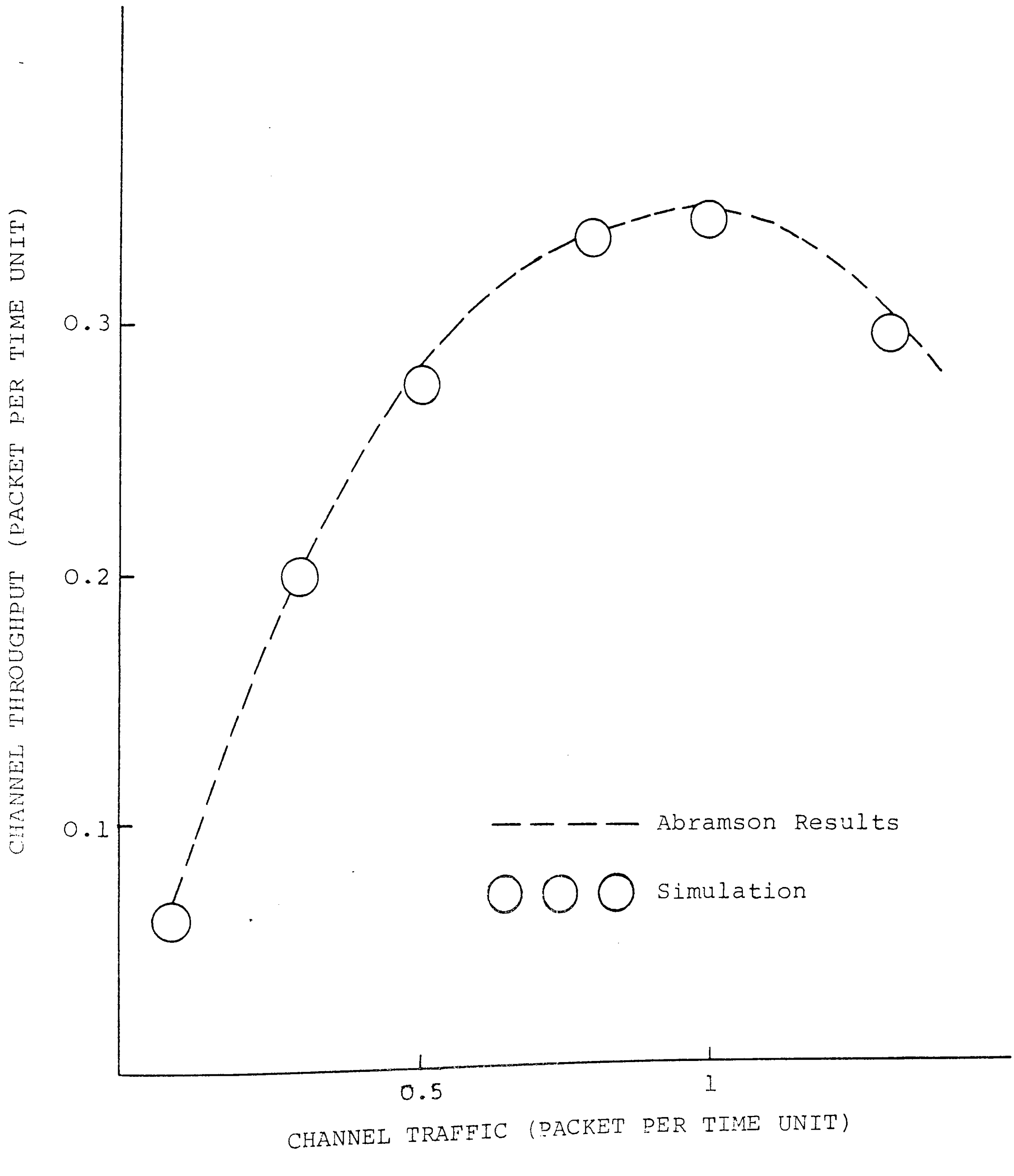

FIGURE (4.6)

CHANNEL THROUGHPUT VERSUS CHANNEL TRAFFIC IN A CELL NETWORK

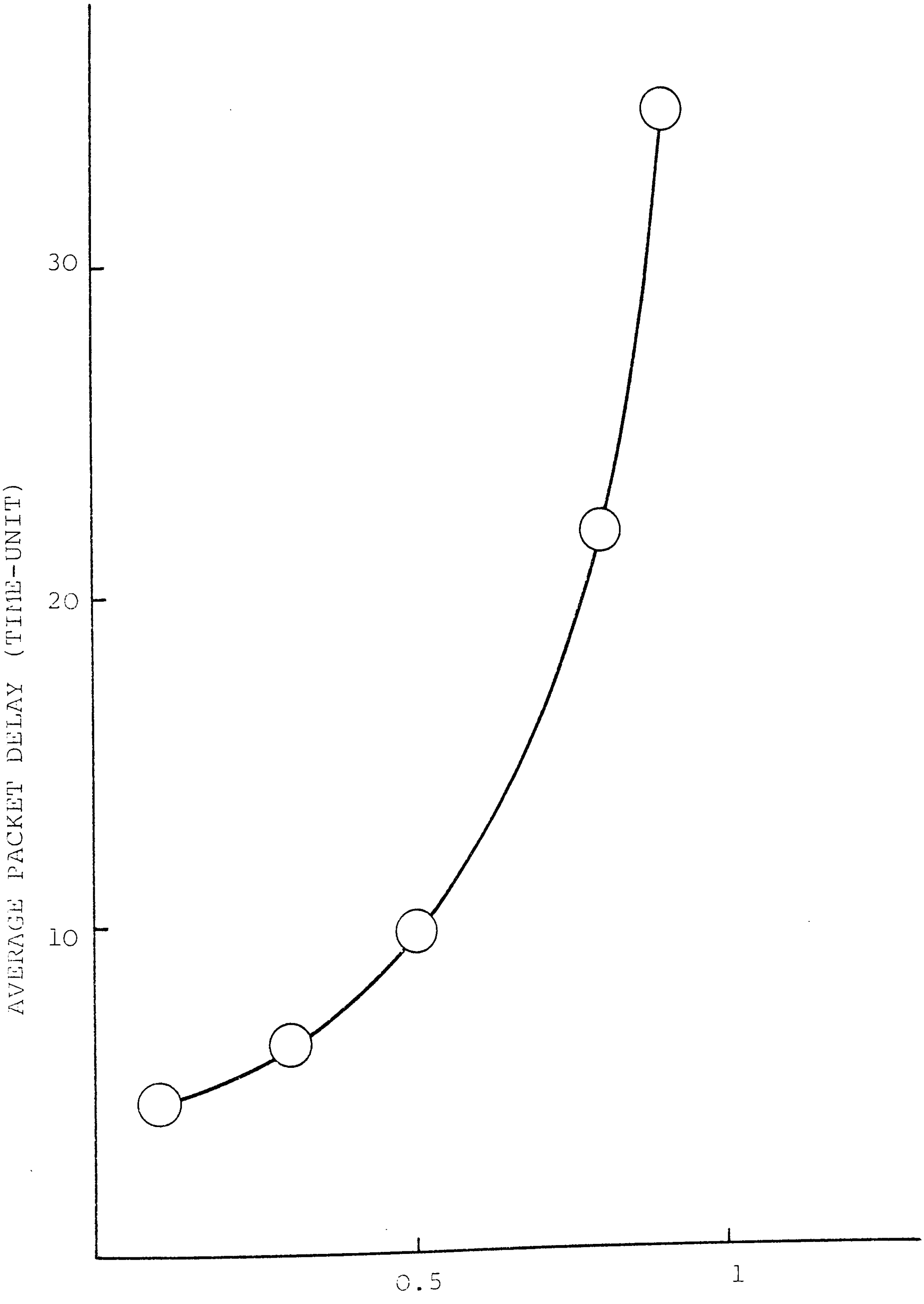

FIGURE (4.7)

AVERAGE PACKET DELAY VERSUS CHANNEL TRAFFIC IN A CELL NETWORK

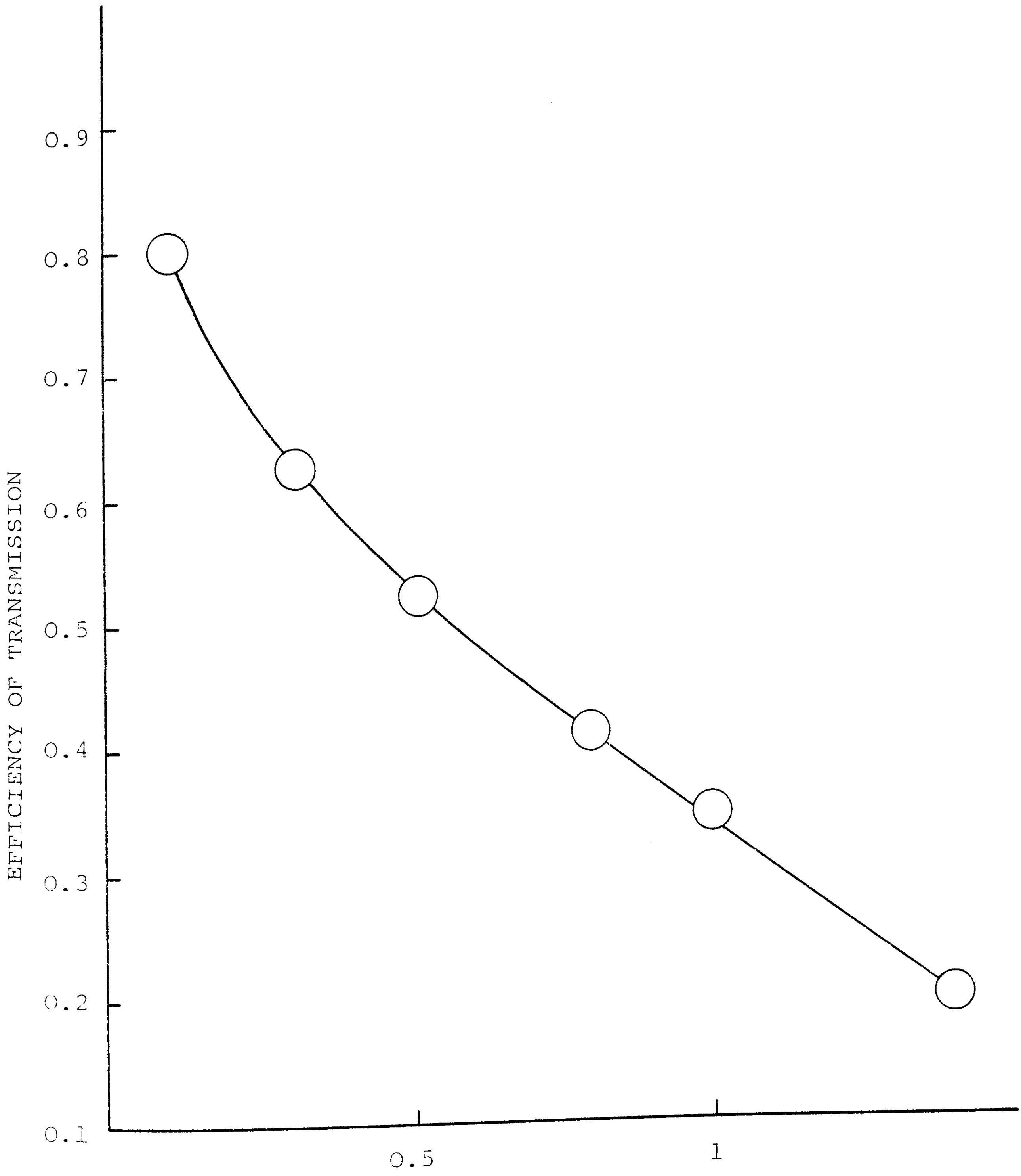

FIGURE (4.8)

EFFICIENCY OF TRANSMISSION VERSUS CHANNEL TRAFFIC IN A CELL NETWORK independently. The system-unit node is assumed to be linked to 10 other nodes via land channels as shown in figure 4.4. Each node is assumed to be interfaced with a Poisson stream of packets. All channels are assumed to be of equal capacity. The performance measurements enumerated here include the following;

1. The maximum buffer size used for the input and ouptut queues of a unit node. The unit for measuring the size of buffers is the page as described in chapter 3. A page has the size of a packet, as it is a packet accommodation.

2. Average packet delay as defined for the cell network case.

The results obtained are as follows:

1. The maximum length of queues is plotted against the traffic rate in figure 4.9. As would intuitively be expected, the length of queues increases as the traffic rate increases.

2. In figure 4.10, the average packet delay is plotted against the traffic rate. It is shown that the delay increases as the traffic rate increases. The reason for this is that as the traffic rate increases packets would have to wait in longer queues before being served.

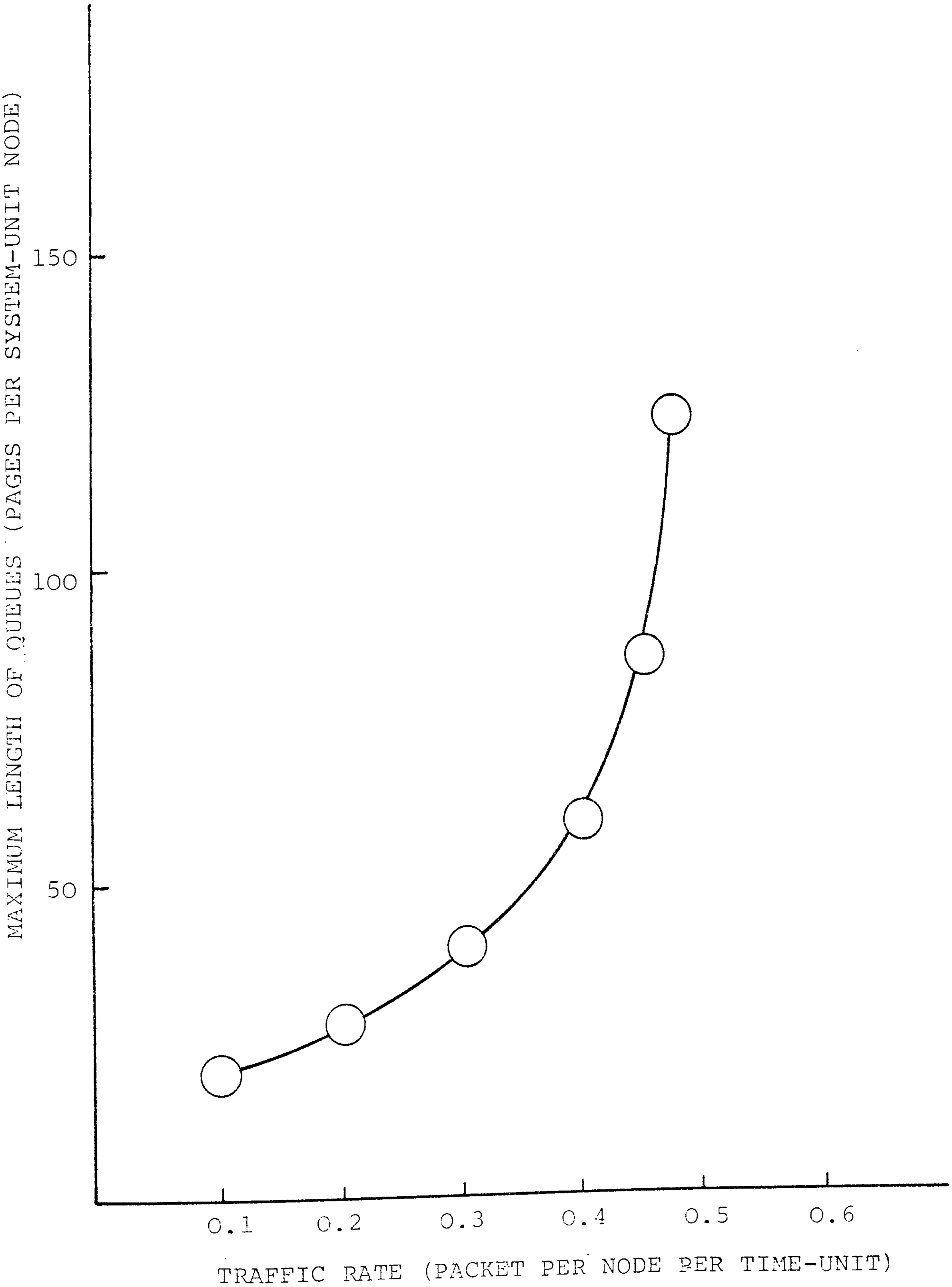

FIGURE (4.9)

MAXIMUM LENGTH OF QUEUES PER SYSTEM-UNIT NODE VERSUS TRAFFIC RATE IN A SYSTEM UNIT NETWORK

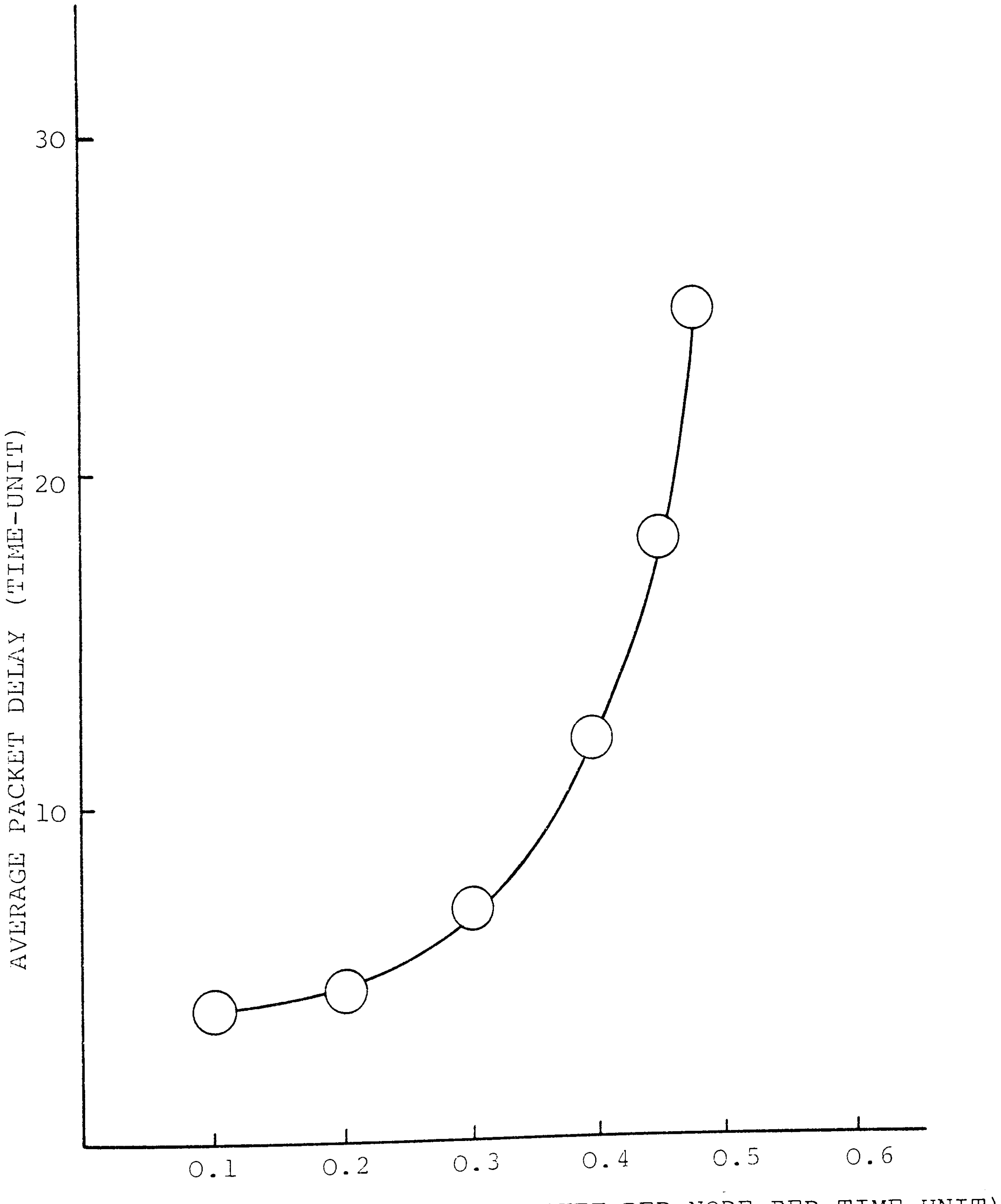

FIGURE (4.10)

AVERAGE PACKET DELAY VERSUS TRAFFIC RATE
IN A SYSTEM-UNIT NETWORK

Simulation of the Inter-City Network

The simulation results presented here are obtained by running the inter-city network part of the simulation independently. In this case, each city node is interfaced with a traffic host generating packets addressed to other city nodes and receiving packets addressed to its node. Two models for the generation of traffic have been used. In one, packets are generated at random with exponentially distributed intervals of time (Poisson traffic). In the other model packets are generated at equally spaced intervals (Uniform traffic). In addition two routing strategies have been considered.

1. In the first, each node chooses randomly, the outgoing link to which a packet destined to another node is to be routed. No information about the packet destination or its previous history is required. This method is called *blind random routing*.

2. The second routing strategy requires information on the previous node that the packet has come from to the present node. On this information, no packets are routed to the outgoing link that leads to the previous node, as the previous node is certainly not the destination. The choice of the

output link is then made randomly between the two possible outgoing links. This method is called *forward random routing.*

The simulation programs have been run for the eight city node network shown in figure 4.5. The destination, of the packets generated at each node are specified randomly. The performance measurements enumerated include the following:

1. The average packet delay of all the packets in their trip from source to destination.

2. The average number of city nodes passed by a packet in its route from source to destination. This is useful for comparing routine strategies.

3. The throughput of a city node measured in terms of the average number of packets passing through a node per time unit.

4. Channel usage measured in terms of the proportion of time during which a given channel is busy.

5. The maximum buffer size used for the input and output queues of a city node.

An illustration of how the routing strategy affects the volume of traffic flowing through the network is shown in figures 4.11 and 4.12. By varying the traffic arrival rate, the throughput of a node and the average channel usage are plotted for two different routing strategies, the blind random routing and the forward random routing. It is shown that for any one value of traffic arrival rate, more traffic is passed through the network when the blind routing is used than when the forward routing is used. The reason for this is that a packet in the first case goes through a longer route than it does in the second case. The average number of nodes passed in the first case is 10 and in the second is 5.

By affecting the volume of traffic flowing through the network, the routing strategy also affects the average packet delay. Figure 4.13 shows, as would be expected, that shorter packet delays result with forward random routing than with blind routing.

Figure 4.14 demonstrates how, for a given value of traffic loading, the maximum size of buffer queues required per node changes by changing the traffic pattern. It is illustrated that longer queues are needed for Poisson traffic than for uniform traffic.

Simulation of a Typical Control Network

The simulation results presented here are obtained by

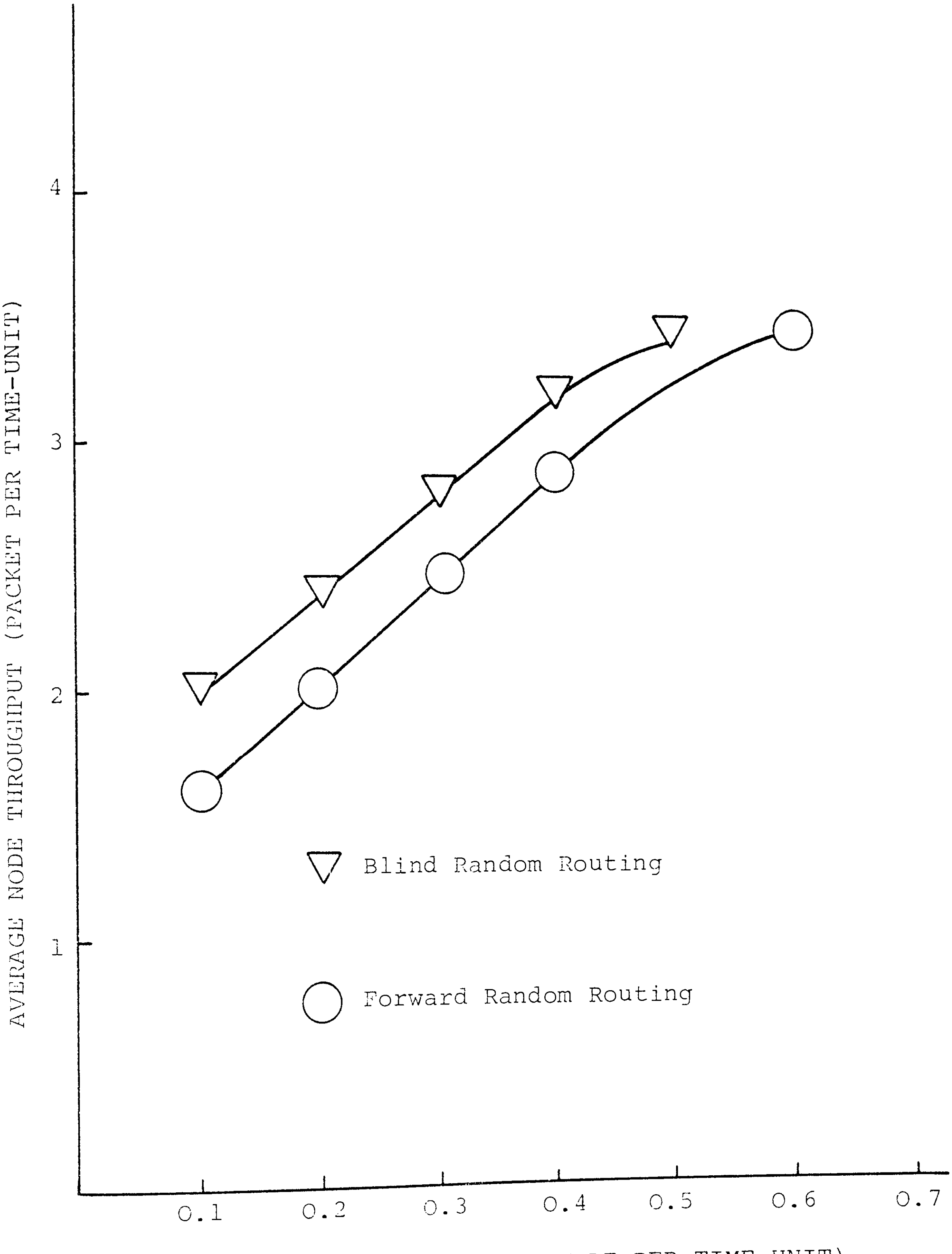

FIGURE (4.11)

AVERAGE NODE THROUGHPUT VERSUS TRAFFIC RATE FOR TWO DIFFERENT ROUTING STRATEGIES IN AN INTER-CITY NETWORK

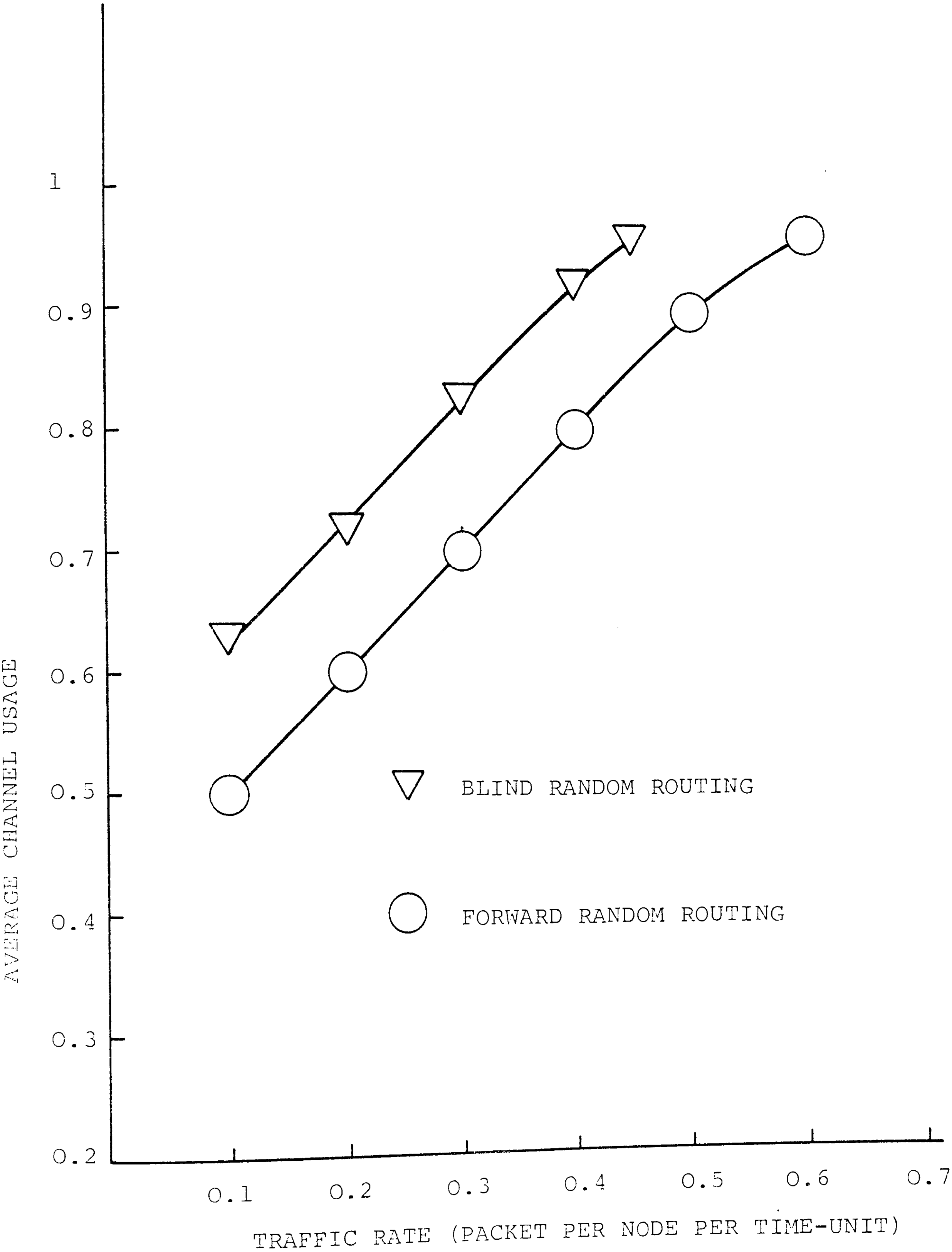

FIGURE (4.12)

AVERAGE CHANNEL USAGE VERSUS TRAFFIC RATE FOR TWO DIFFERENT ROUTING STRATEGIES IN AN INTER-CITY NETWORK

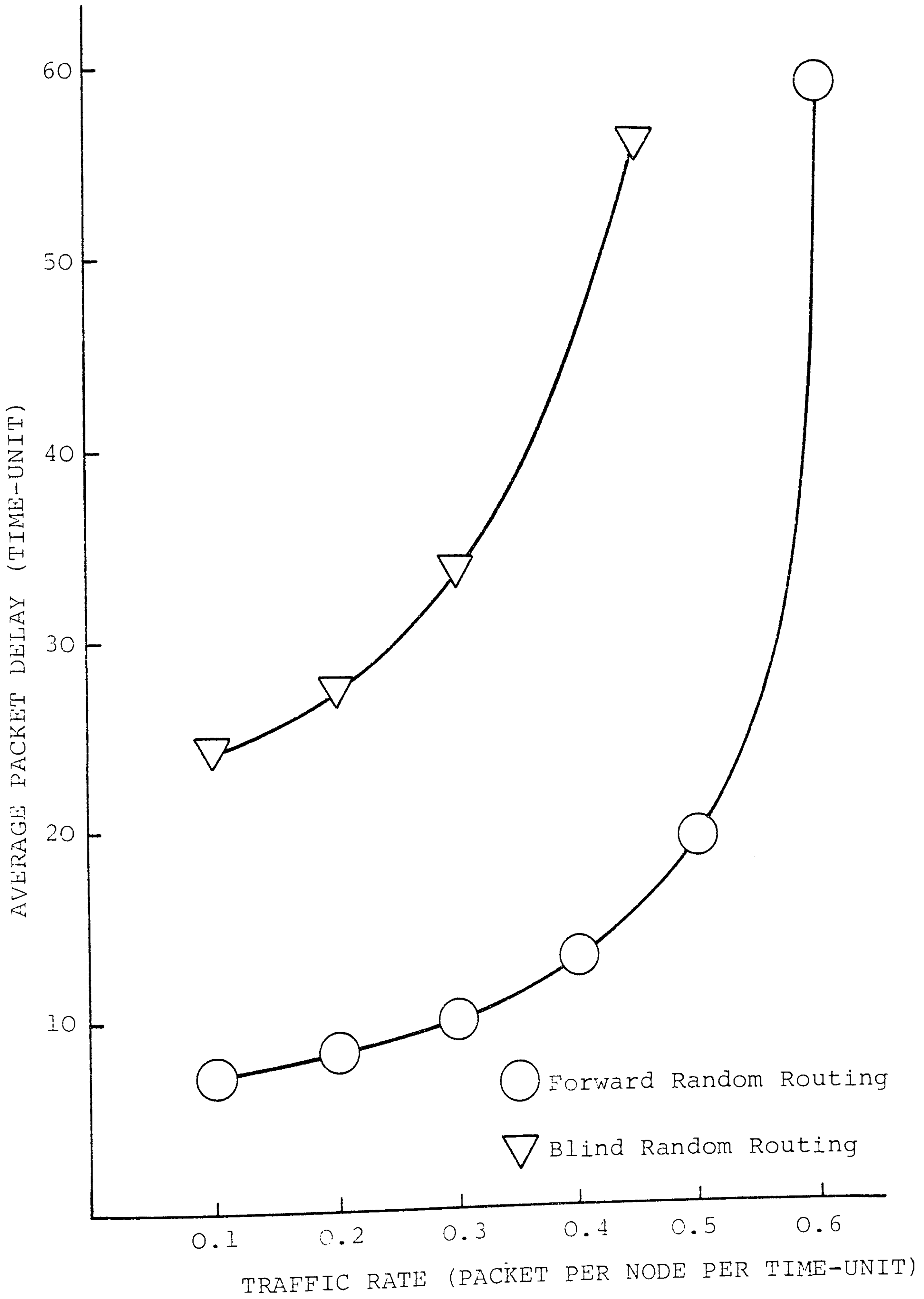

FIGURE (4.13)

AVERAGE PACKET DELAY VERSUS TRAFFIC RATE FOR TWO DIFFERENT ROUTING STRATEGIES IN AN INTER-CITY NETWORK

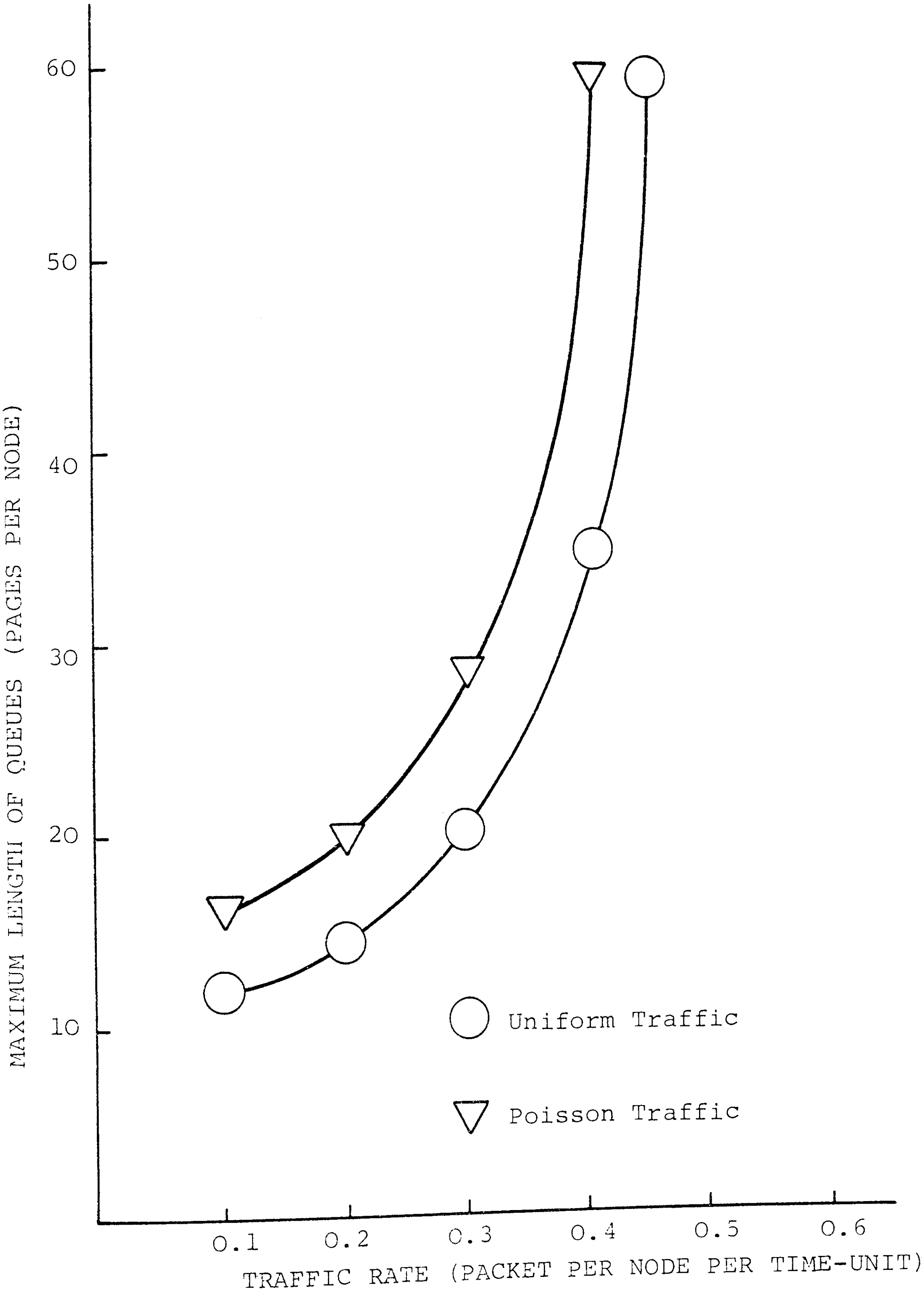

FIGURE (4.14)

MAXIMUM LENGTH OF QUEUES PER NODE VERSUS TRAFFIC RATE FOR TWO DIFFERENT TRAFFIC STREAMS IN AN INTER-CITY NETWORK running the simulation programs for a typical example of a control network. Each mobile is considered to be interfaced with a stream of calls generated at random with exponentially distributed interval of times. Each call is assumed to generate a specific sequence of packet transmissions for performing the various control functions required for the handling of calls. The control functions have been applied in steps, starting with the basic control functions in a single cell and moving to the more complex problems of the hand-over of calls and the address-changing of mobiles. The data traffic streams derived in section 4.3 are used here.

The example considered starts with initial values given to the capacity of the network channels, the processing requirements of nodes and the rate of calls per mobile. The main variable for which the simulation was run is the number of mobiles per cell. The main performance measure considered is responsiveness. The example enumerates responsiveness in terms of the time required for performing the various control functions.

The initial values given in the example are described in the following. The packet size has been considered to be 96 bits and the capacities of the data channels have been given as 2400 bps for the radio channel, 1200 bps for each of the unit node channels and 4800 bps for each inter-city channel. The transmission of a packet through a channel would require therefore, 40 ms for the radio data channel, 80 ms for a unit node channel and 20 ms for an inter-city channel.

The time spent by the processor of a cell node at each of its input buffers is considered to be 20 ms. This includes the time for testing and processing a packet. For a unit node, this time is 8 ms and for a city node, it is 5 ms. This is a reasonable assumption, since a time of 5 ms per task (89) seems to be typical in stored program control telephone exchange systems.

Each mobile is assumed to generate one call of each type per hour ($r_1=r_2=r_3=1$ call per hour). In the case where address-changing is required, each mobile is considered to transmit its identification number once every five minutes ($r=12$ times per hour). The proportion of calls that requires hand-over is assumed to be 10 percent, 5 percent for low-level and 5 percent for high-level hand-over ($\ell_1=\ell_2=0.05$, $\ell=0.10$). The proportion of mobiles which travel between cities is considered to be 10 percent ($e=0.10$).

The results obtained are presented in the following.

Figure 4.15 shows the average call set-up delay for a single-cell system. It is shown that up to 2500 mobiles can be accommodated in such a system for an average call set-up delay less than 3 seconds.

Figure 4.16 shows the average call set-up delay for two multi-cell systems. In the first, only the basic control functions are considered while in the second all the control functions are performed. It is shown that the first system can accommodate 1400 mobiles for an average call set-up delay less than 3 seconds, while the other system can only accommodate 650 mobiles for the same delay. The main reason for this is the flow of extra data traffic generated by the complementary control functions through the various parts of the network.

Figure 4.17 shows the average hand-over delay for two cases. In the first case, address-changing functions are not performed while in the second such functions are implemented. As may be expected, more mobiles can be served in the first case than in the second, at any given time delay level.

Figure 4.18 shows the average inter-city address-changing delay for two different routing strategies over the inter-city network. The routing strategies used are the blind random routing and the forward random routing described previously. It is shown that for a given level of time-delay more mobiles can be accommodated if a better routing scheme is used.

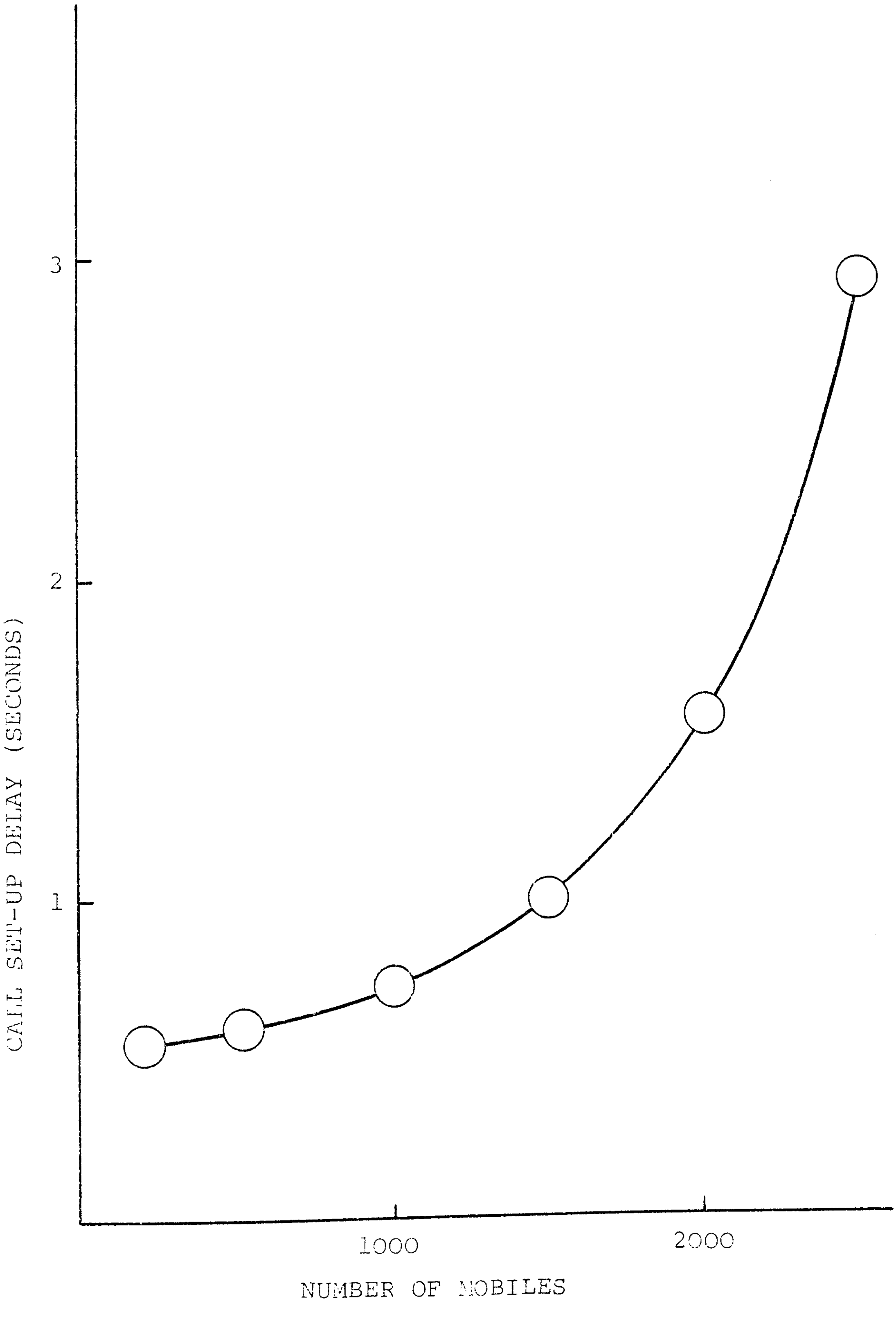

FIGURE (4.15)

CALL SET-UP DELAY IN A SINGLE-CELL SYSTEM

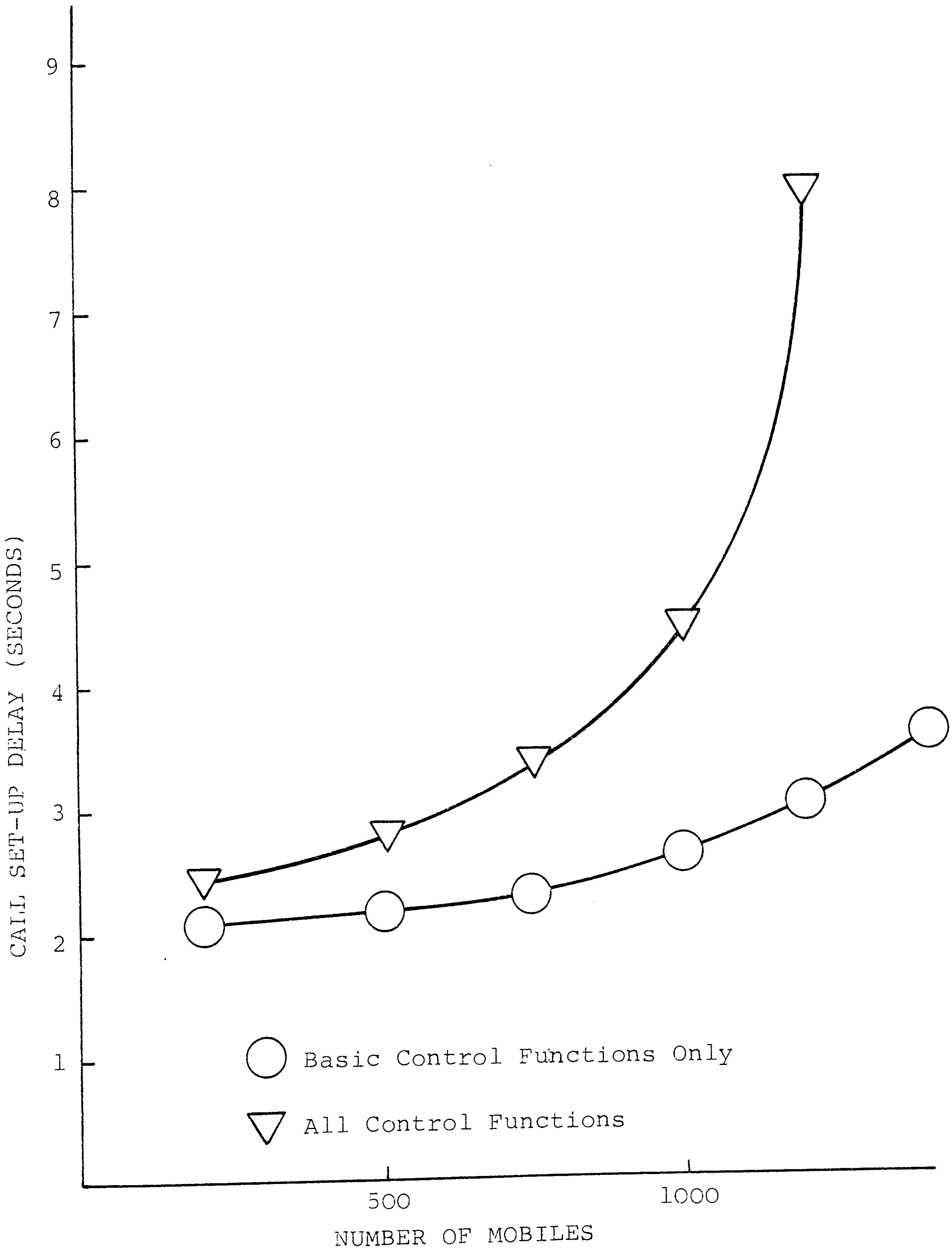

FIGURE (4.16)

CALL SET-UP DELAY IN A MULTI-CELL SYSTEM

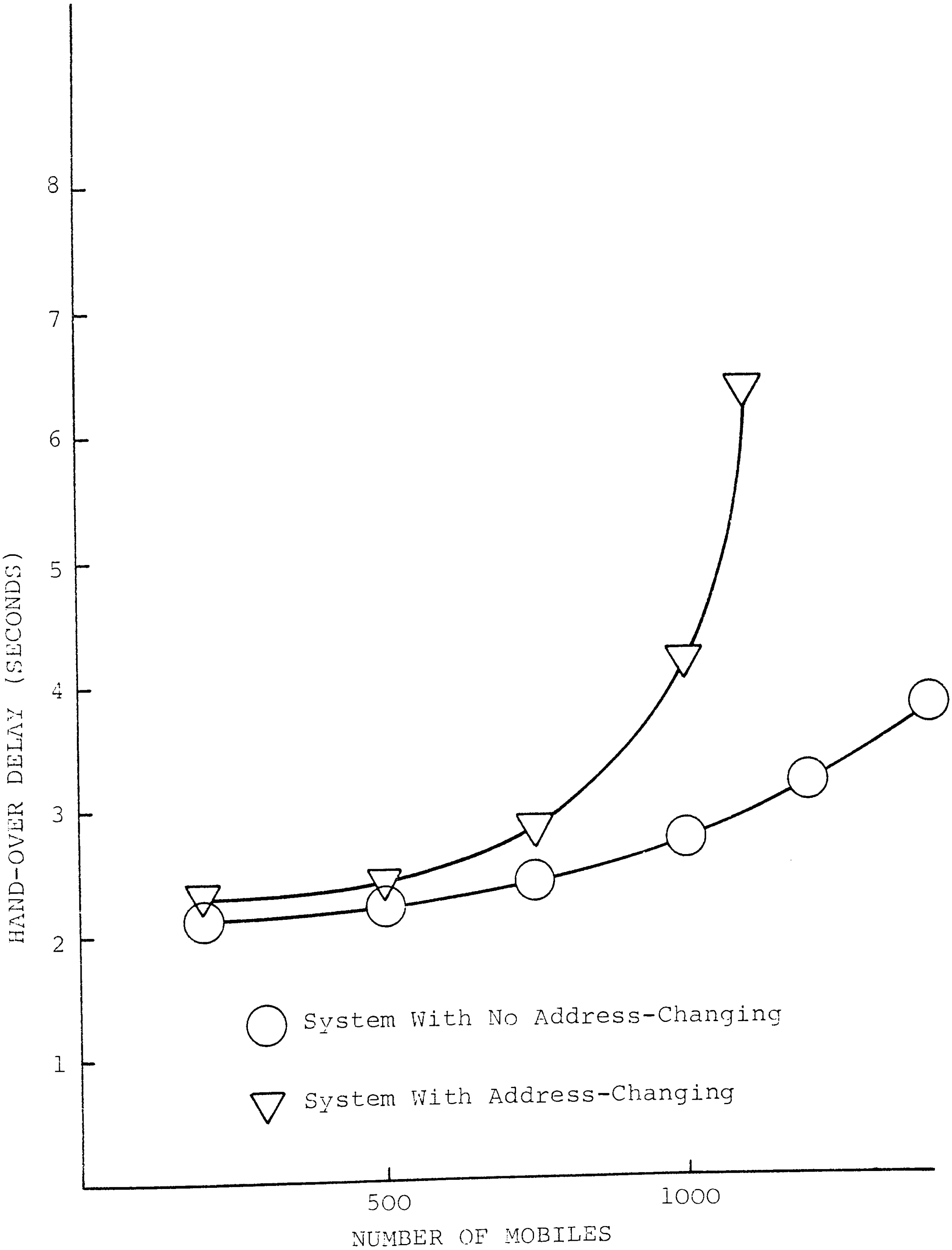

FIGURE (4.17)

HAND-OVER DELAY

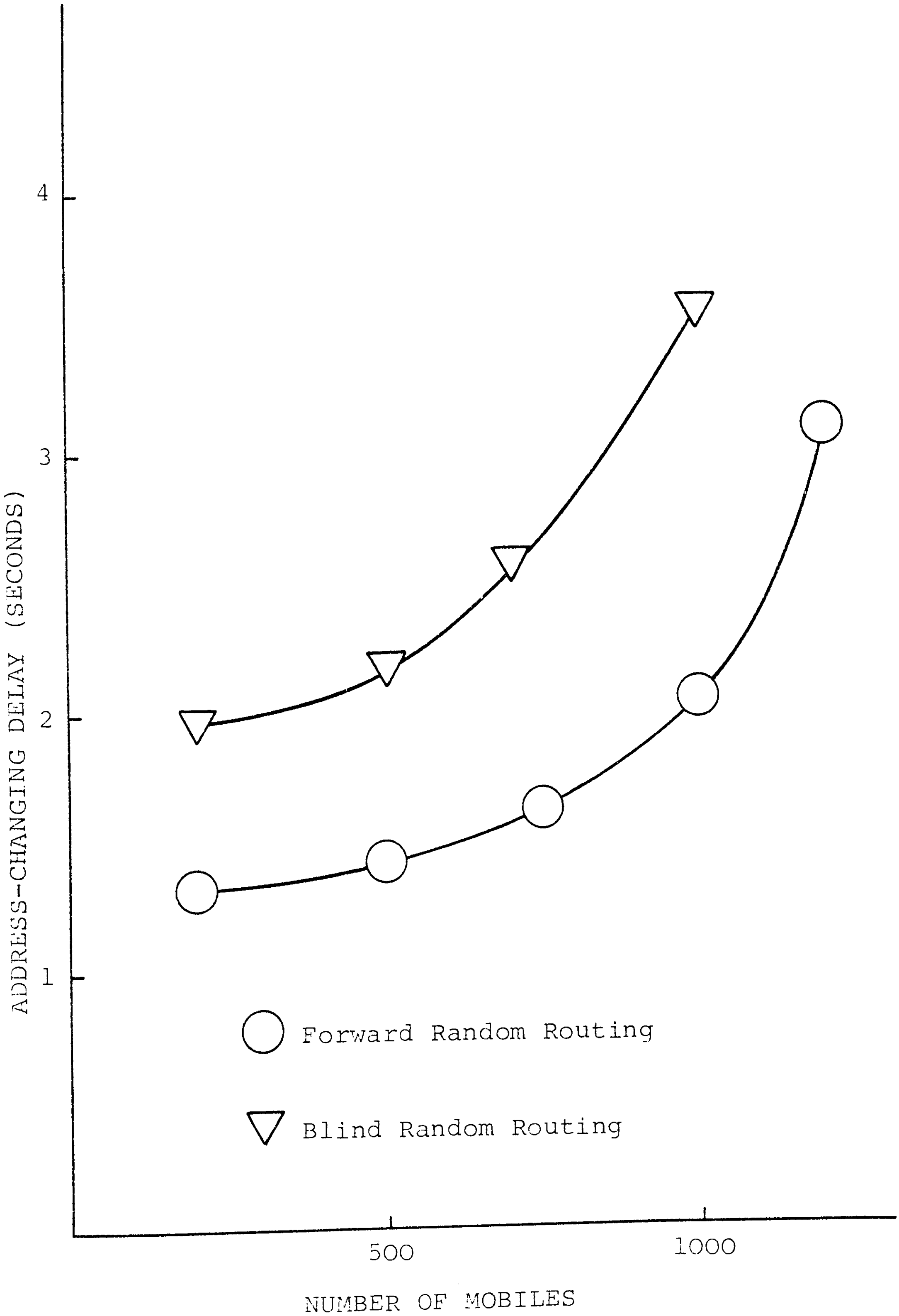

FIGURE (4.18)

ADDRESS-CHANGING DELAY

Evaluation

The simulation programs presented here enabled an investigation of a possible control network for cellular mobile radio systems to be carried out. Although several previous publications have considered the control requirements of such systems[23 - 28], non of these publications have used digital computer simulation for the investigations of such problems. While there are no previous results with which to compare the present results, the feasibility of cellular mobile radio systems is confirmed here from the data transmission viewpoint.

The simulation of each of the three parts of the proposed control network independently has provided a test on the accuracy of the simulation programs and has illustrated the flexibility of their use. The results obtained for the cell network agreed closely with the previously published analytical results for slotted Aloha systems[53] and the results obtained for the other parts of the control network agree with what would intuitively be expected. It has been shown that some network components are associated with one part of the network and are independent of the other parts. An example of such components is the routing strategy at the inter-city level. Such a component can be investigated independently using the part concerned.

The results obtained by the simulation of a typical control network show that small minicomputers interconnected via voice grade data channels should provide a satisfactory control performance. It has been illustrated that for an acceptable level of call set-up delay, the rate of calls that can be handled by the control network depends on the control functions performed. If the network performs the basic control functions only, the rate of calls that can be handled would be twice as much as in the case where all the control functions are performed. Although the simulation was run for a specific example with certain data traffic streams, other examples with different data streams could be considered, as the network model used is of a flexible nature and independent of the traffic sources.

CHAPTER FIVE

CONCLUSIONS AND FUTURE RESEARCH

5. CONCLUSIONS AND FUTURE RESEARCH

5.1 CONCLUSIONS

This thesis has described two main contributions to the design of computer controlled cellular mobile radio systems. The first is the derivation of new teletraffic formulas for single-cell and multi-cell systems. These formulas enable the grade of service in such systems to be related to the traffic offered and the number of frequency channels available for use. The second contribution is the development of digital computer simulation tools for the investigation of various teletraffic and computer control aspects of mobile radio systems. The simulation tools for the teletraffic aspects provide insight into the behaviour of the speech traffic of mobiles and the tools for the computer control aspects help in the design of computer networks for the handling of the speech traffic.

The simple analysis for single-cell systems shows that the effect of tromboning caused by the mobile-to-mobile calls can be taken into account while the Erlang and Engset formulas, or existing tables of their values,are used. For the congestion of one-channel calls, the Erlang and Engset formulas can be used directly. For the congestion of two-channel calls, the Erlang formula can be used but with the traffic and the number of channels halved and the Engset formula can be used with the number of mobiles and the number of channels halved. For the congestion averaged over all types of call, the results can be averaged with weights in proportion to the different types of traffic.

The second approach used for the analysis of single-cell systems has produced new teletraffic formulas with more complicated forms than the Erlang and Engset formulas. The results obtained from the two approaches have been evaluated by comparison with results obtained by using the simulation tools developed for the teletraffic aspects. While the formulas derived by the first approach provide a rough approximation, the formulas derived by the second approach produce a closer approximation.

The results obtained for single-cell systems apply to multi-cell systems, in which there is a fixed assignment of frequency channels to cells. The new teletraffic formulas derived for multi-cell systems with dynamic channel assignment show that although the dynamic case requires higher level of equipment provision than the fixed channel assignment case, the dynamic case has the advantage of increasing the teletraffic capacity of multi-cell systems. This confirms the prediction of the previously published simulation results on the subject.

It has been shown that the teletraffic formulas derived for the fixed and dynamic channel assignment cases can be combined to produce new teletraffic formulas which apply to systems using hybrid channel assignment. Such formulas are useful, as the hybrid channel assignment case offer a cost-benefit compromise between the level or equipment provision and the teletraffic capacity of multi-cell systems. The formulas agree closely with the published simulation results against which they have been compared.

The application of the top-down structured programming approach to the simulation of the teletraffic and computer control aspects of mobile radio systems has been illustrated. The structured approach has enabled the simulation programs to be written successfully in a way that permits modifications to the simulation to be easily incorporated. Although the programs were intended to simulate particular models, the flexibility resulting from the top-down approach permits other models, not originally envisaged, to be simulated with little extra effort.

The application of the time-true method and the roulette method to the simulation of the teletraffic aspects of mobile radio systems has been described. It has been demonstrated that while the time-true method is costly in terms of programming effort and computing time, the method provides a comprehensive insight into the behaviour of the simulated system. As one could expect, it has also been shown that although the roulette method does not give complete information about the behaviour of the system being simulated, the method is economical in computing time and programming effort. Coral 66 on the PDP-11/03 proved to be well-matched to the task of programming the roulette simulation and Algol 68 with its more elaborate facilities proved to be well-suited to the time-true method. The use of the two separate simulation methods was useful in providing a check on the validity of the simulations.

The simulation of the proposed control network in three parts each corresponding to one of its hierarchical levels has been useful in many ways. It enabled a check for accuracy to be made on the simulation. The results obtained by running the cell part of the network which uses the slotted Aloha technique of packet transmission agreed closely with previously published analytical results on the same technique. It has been shown that some network components such as the routing strategy at the city level can be investigated using one part of the simulation only. In addition, the three parts of the simulation have provided flexibility in the investigation of the control requirements in stages, starting with the basic control functions in a single-cell system and moving to the more complex problems of the hand-over of calls and the address-changing of mobiles in multi-cell systems.

The typical example of a control network and the data traffic streams considered for performing the various control functions have shown that large-scale mobile radio systems are feasible from the data transmission viewpoint. It has been illustrated that small mini-computers interconnected via voice grade data channels should provide a satisfactory control performance. In addition, it has been shown that for an acceptable level of call set-up delay, the capacity of a control network depends on the control functions performed. In the case where only the basic control function are required the capacity of the network would be twice as much as it would be in the case where all the control functions are performed.

5.2 FUTURE RESEARCH

The cellular concept appears to be a promising approach in increasing the capacity of mobile radio systems by providing an efficient utilization of the frequency spectrum available for mobile use. A further increase of the capacity of such systems would be of interest.

The present research has illustrated the effect of different channel assignment schemes on the capacity of cellular systems. It would be of interest for future research to investigate the effect of the interaction between the assignment of channels and the movement of mobiles on the capacity of such systems. For this purpose, new channel assignment and call rearrangement algorithms may be developed to operate in accordance with the movement of mobiles. Such algorithms could provide a further increase of the capacity of cellular systems.

Another method that may help in increasing the capacity of cellular systems is described in the following. Previous publications on such systems and the work presented in this thesis have considered the use of frequency channel switching for speech traffic. Although the packet switching technique was originally designed for use by data networks, recent publications suggest the use of this technique for speech transmission[90,93]. It would be of interest for future research to investigate the effect of using packet switching for speech communication on the capacity of cellular systems.

By using the Aloha technique together with the various control procedures designed to increase its efficiency[47-51], a further increase of the capacity of cellular systems could be obtained. However, problems such as packet formation delay and message reassembly may make such applications impracticable for some users. One method that may overcome such problems is to divide the mobile subscribers into two classes; a *disciplined user* class and a *free user* class. The first would include users with special tasks such as mobiles in a commercial fleet, trained on using speech communication by packet switching. The second class would include the usual common subscribers and would use frequency channel switching for speech traffic.

APPENDICES

APPENDIX I

COMPUTING TELETRAFFIC FORMULAS

In order to illustrate how the new teletraffic formulas derived in chapter 2 can be computed, the following problems are considered here.

1. Computing the channel occupancy in a cellular system requires computing the average number of cells in the frequency reuse buffering area. This problem is considered here for a cellular system of a given construction.

2. In some cases computing the teletraffic formulas using computers causes the occurrence of arithmetic overflows. Algorithms which may avoid such overflows have been developed. Some of these algorithms are presented here in the form of flowcharts.

3. In order to illustrate how computer programs may implement such algorithms, a Coral 66 program for computing complicated formulas is presented.

4. As an example of the results that can be obtained by computing the teletraffic formulas, traffic-congestion tables for single-cell and multi-cell systems are presented.

Computing the Average Number of Cells in the Frequency Reuse Buffering Area of a Cellular System

Consider a cellular system of equal sized hexagon-shaped cells. For cells towards the centre of the system, the first buffer belt will consist of 6 cells and the second belt will consist of 12 cells. Naturally, these numbers decrease for cells near the boundary of the system. If b_1 and b_2 are the numbers of cells in the first and second belts respectively around a particular cell, the average number of cells in these belts for the whole system is assumed $b_{1,av}$ and $b_{2,av}$ respectively. For a cellular system that considers a frequency reuse buffering area of one belt, the average number of cells in the buffer would be $g = b_{1,av}$. For a system with a buffer of two belts, g becomes $g = b_{1,av} + b_{2,av}$.

Figure I.1 shows the lay-out of a circular cellular system with a central cell and J circles of cells around it. If j is the counter of circles, each circle will consist of 6j cells structured on 6 sides, with j cells at each side. In addition, the following rules may be observed in this structure:

1. The total number of cells in the system may be given as $N = 1 + \sum_{j=0}^{J} 6.j.$

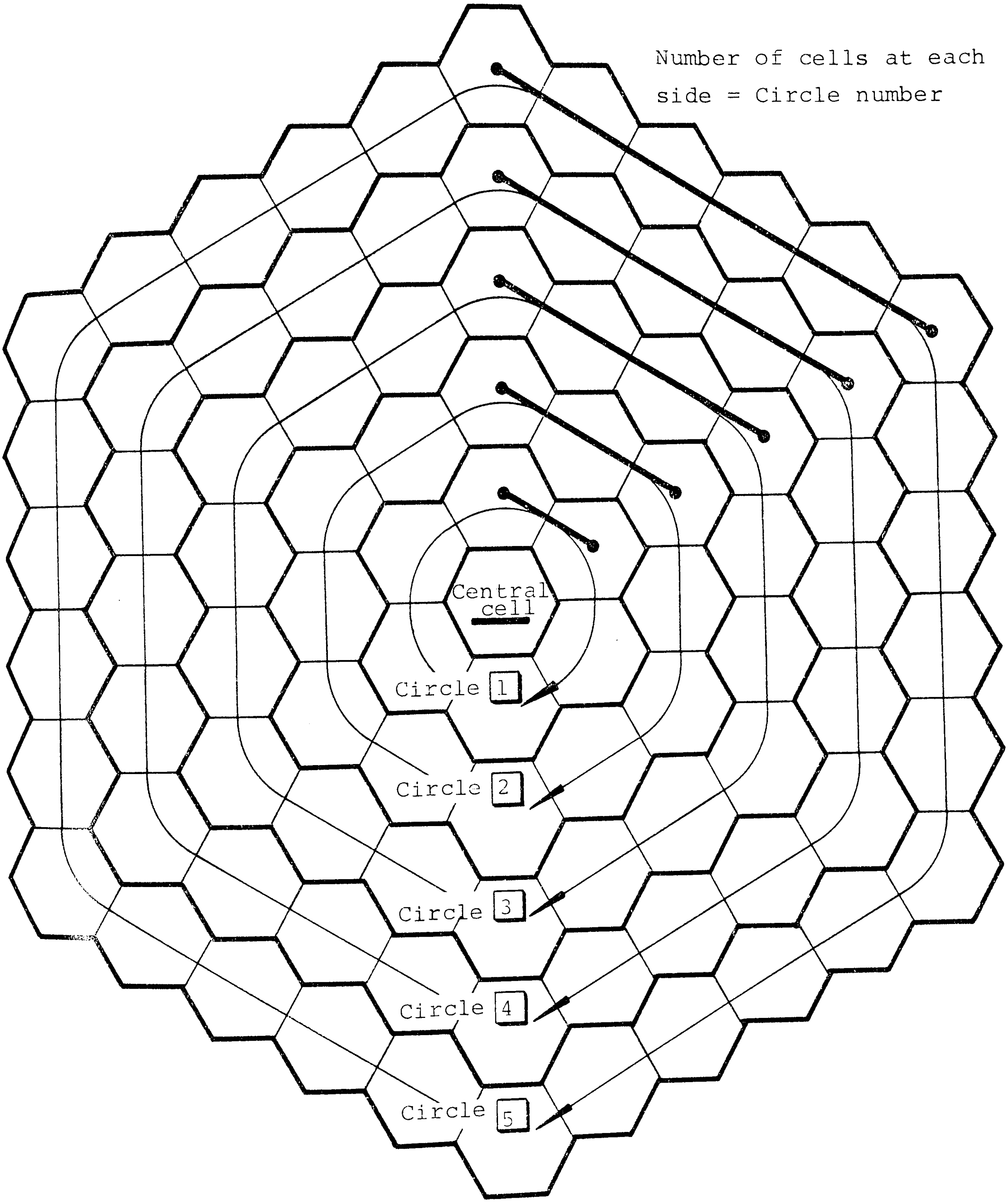

FIGURE (I.1)

THE LAY-OUT OF A CELLULAR SYSTEM WITH A CENTRAL CELL AND 5 CIRCLES OF CELLS AROUND IT

2. All cells at circles $j \leq (J-2)$ have $b_1=6$ cells and $b_2=12$ cells. The number of these cells is

$$1 + \sum_{j=0}^{J-2} 6.j \quad .$$

3. All cells at circle $(J-1)$ have $b_1=6$ and $b_2<12$. One cell at each side has $b_2=7$, while all the others have $b_2=9$.

4. At circle J of the system, one cell at each side has $b_1=3$, while all the others have $b_1=4$. In addition, at each side of circle J one cell has $b_2=5$ and two cells have $b_2=6$, while all the others have $b_2=7$.

Using the above rules, g may be computed. If the buffer is considered to be a single belt of cells g becomes

$$g = b_1,av = \frac{6. \sum_{j=0}^{J-1} 6.j + 24\ J}{1 + \sum_{j=0}^{J} 6.j}$$

If instead the buffer is considered to be of two belts of cells, g becomes

$$g = b_1,av+b_2,av = \frac{6 \sum_{j=0}^{J-1} 6.j + 12 \sum_{j=0}^{J-2} 6.j + 120\ J-78}{1 + \sum_{j=0}^{I} 6.j}$$

These formulas are applicable to systems with $J \geq 3$.

Algorithms for Computing the Teletraffic Formulas

Figures I.2 to I.7 show flowcharts for computing various mathematical expressions used by the teletraffic formulas derived in chapter 2.

A Computer Program for Computing Teletraffic Formulas

A Coral 66 program built in a modular construction for computing the following teletraffic formulas derived for a single cell system with a finite number of mobiles is listed after figure I.7.

$$B_1 = \frac{\displaystyle\sum_{x=0}^{C/2} \frac{(M+2x-C)!\,(a_1+a_2)^{2x}\,(M/2-x)!\,a_3^{(C/2-x)}}{(M-C)!\,(2x)!\,[(M-C)/2]!\,(C/2-x)!}}{\displaystyle\sum_{x=0}^{C}\sum_{y=0}^{<(C-x)/2>} \frac{(M-2y)!\,(a_1+a_2)^{x}\,[(M-x)/2]!\,a_3^{y}}{(M-2y-x)!\,x!\,[(M-x)/2-y]!\,y!}}$$

$$B_3 = B_1 + \frac{\displaystyle\sum_{x=0}^{C/2-1} \frac{(M+2x+2-C)!\,(a_1+a_2)^{(2x+1)}\,[(M-2x-1)/2]!\,a_3^{(C/2-x-1)}}{(M+1-C)!\,(2x+1)!\,[(M+1-C)/2]!\,(C/2-x-1)!}}{\displaystyle\sum_{x=0}^{C}\sum_{y=0}^{<(C-2)/2>} \frac{(M-2y)!\,(a_1+a_2)^{x}\,[(M-x)/2]!\,a_3^{y}}{(M-2y-x)!\,x!\,[(M-x)/2-y]!\,y!}}$$

$$B = [(a_1+a_2)B_1+a_3B_3]/(a_1+a_2+a_3)$$

Traffic-Congestion Tables

Tables I.1 to I.3 consider the congestion in single-cell systems. Tables I.4 and I.5 consider the congestion in multi-cell systems.

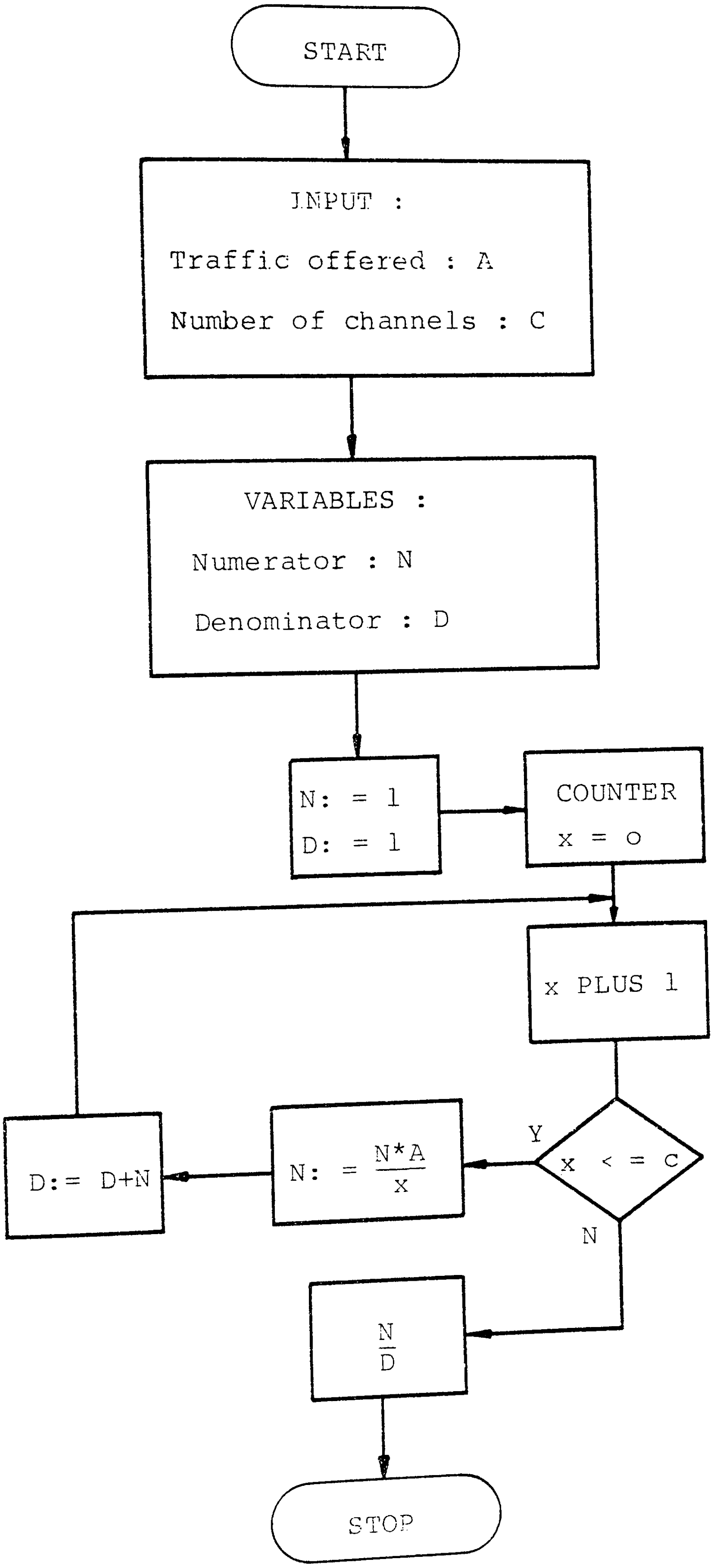

FIGURE (I.2)

Computing $\frac{A^C}{C!} \Big/ \sum_{x=0}^{C} \frac{A^x}{x!}$

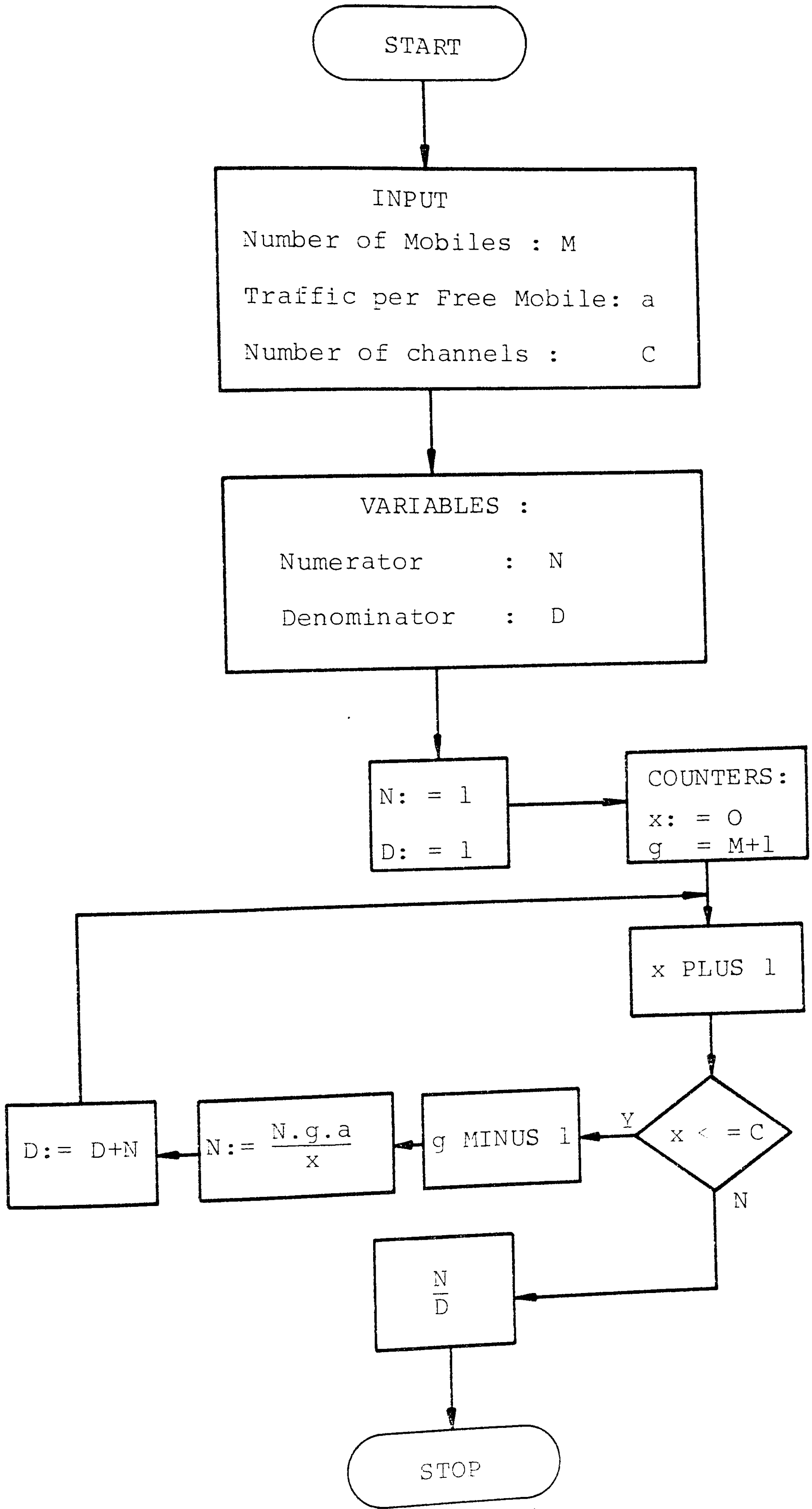

FIGURE (I.3)

Computing $\dfrac{\dfrac{M!}{(M-C)!}\cdot\dfrac{a^{C}}{C!}}{\sum_{x=0}^{C}\dfrac{M!}{(M-x)!}\cdot\dfrac{a^{x}}{x!}}$

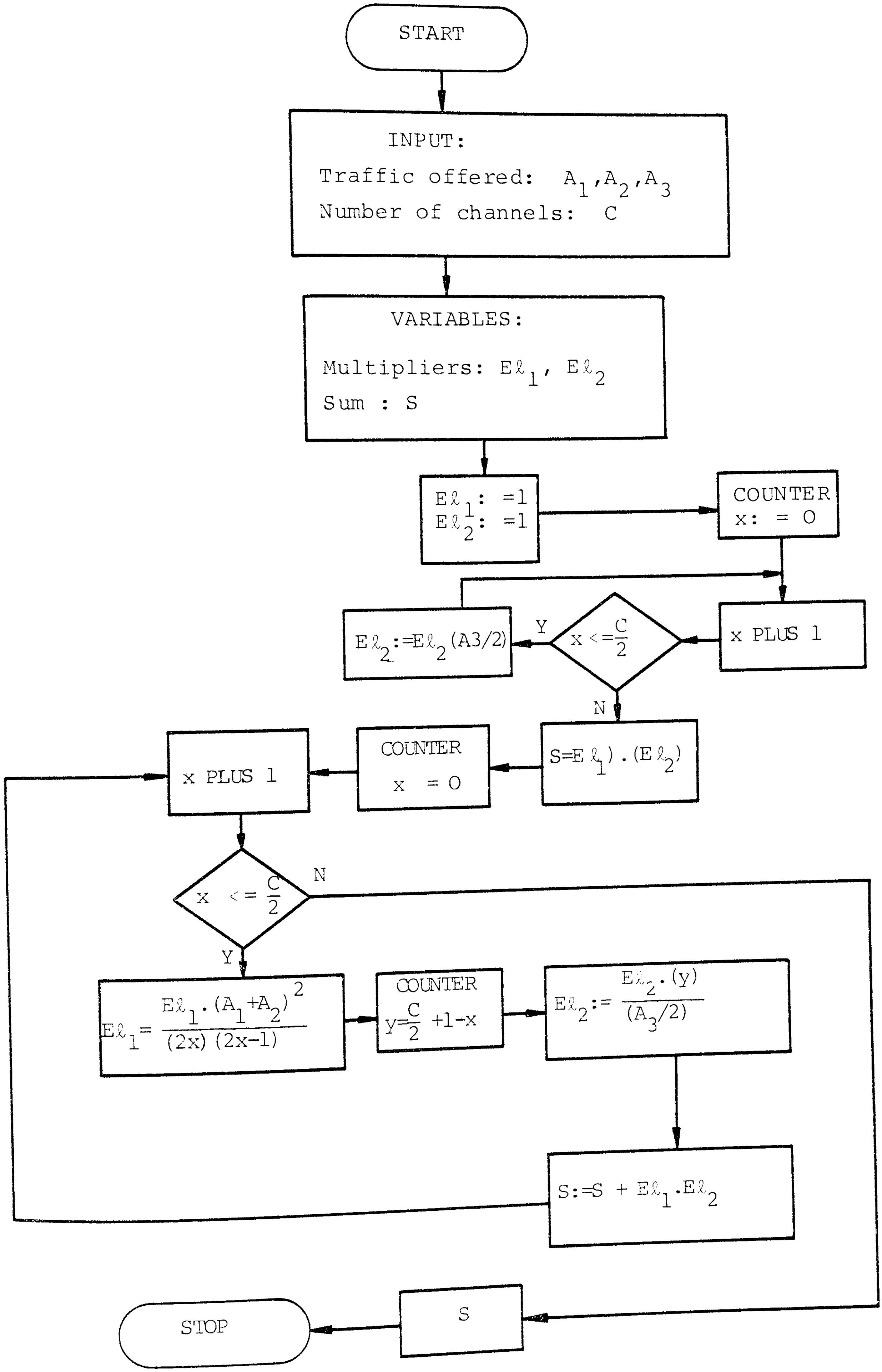

FIGURE (I.4)

Computing $\sum_{x=0}^{\frac{C}{2}} \frac{(A_1+A_2)^{2x}}{(2x)!} \cdot \frac{(A_3/2)^{(\frac{C}{2}-x)}}{(\frac{C}{2}-x)!}$

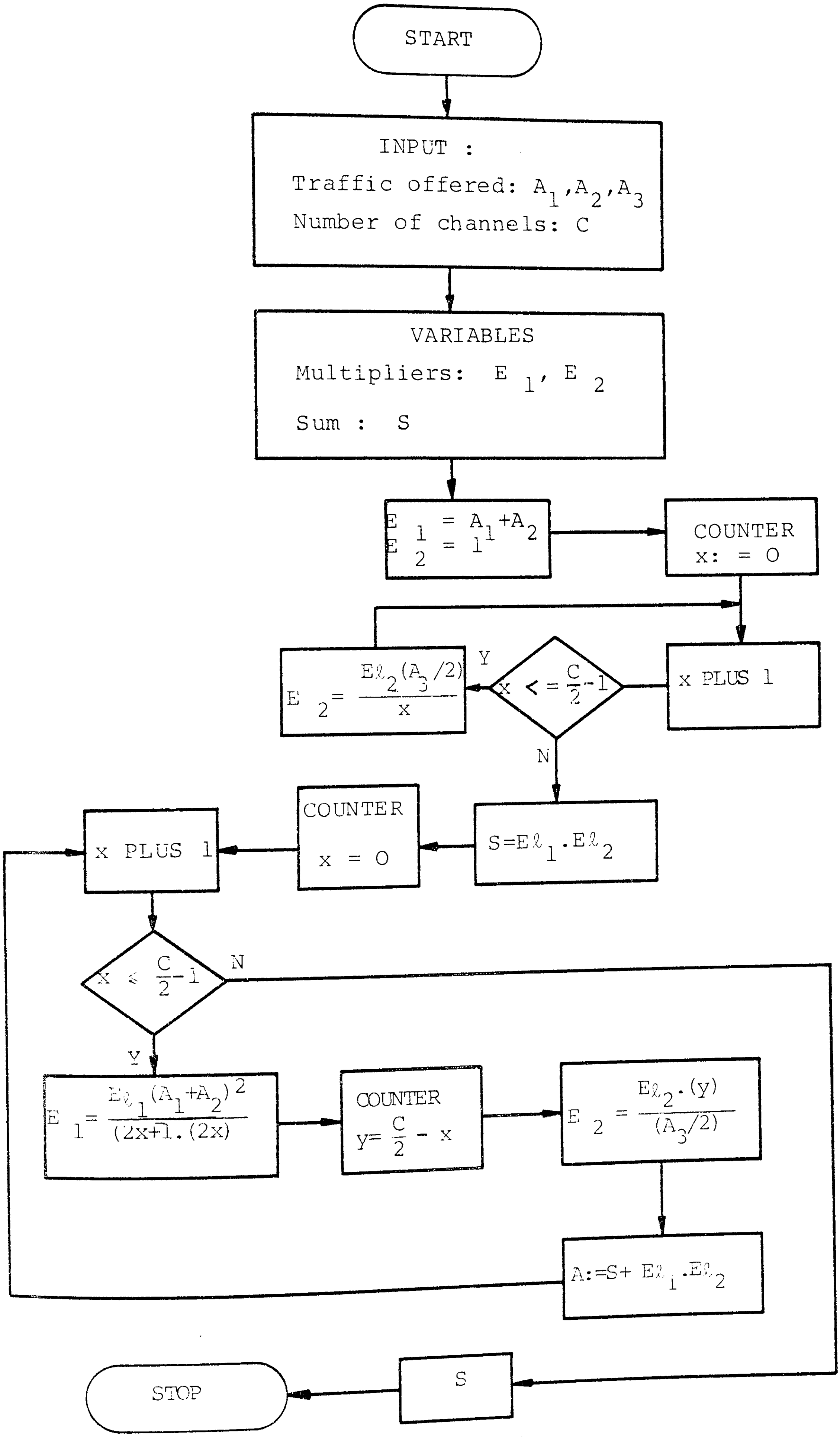

FIGURE (I.5)

Computing $\displaystyle\sum_{x=0}^{\frac{C}{2}-1} \frac{(A_1+A_2)^{(2x+1)}}{(2x+1)!} \cdot \frac{(A_3/2)^{(\frac{C}{2}-x-1)}}{(\frac{C}{2}-x-1)!}$

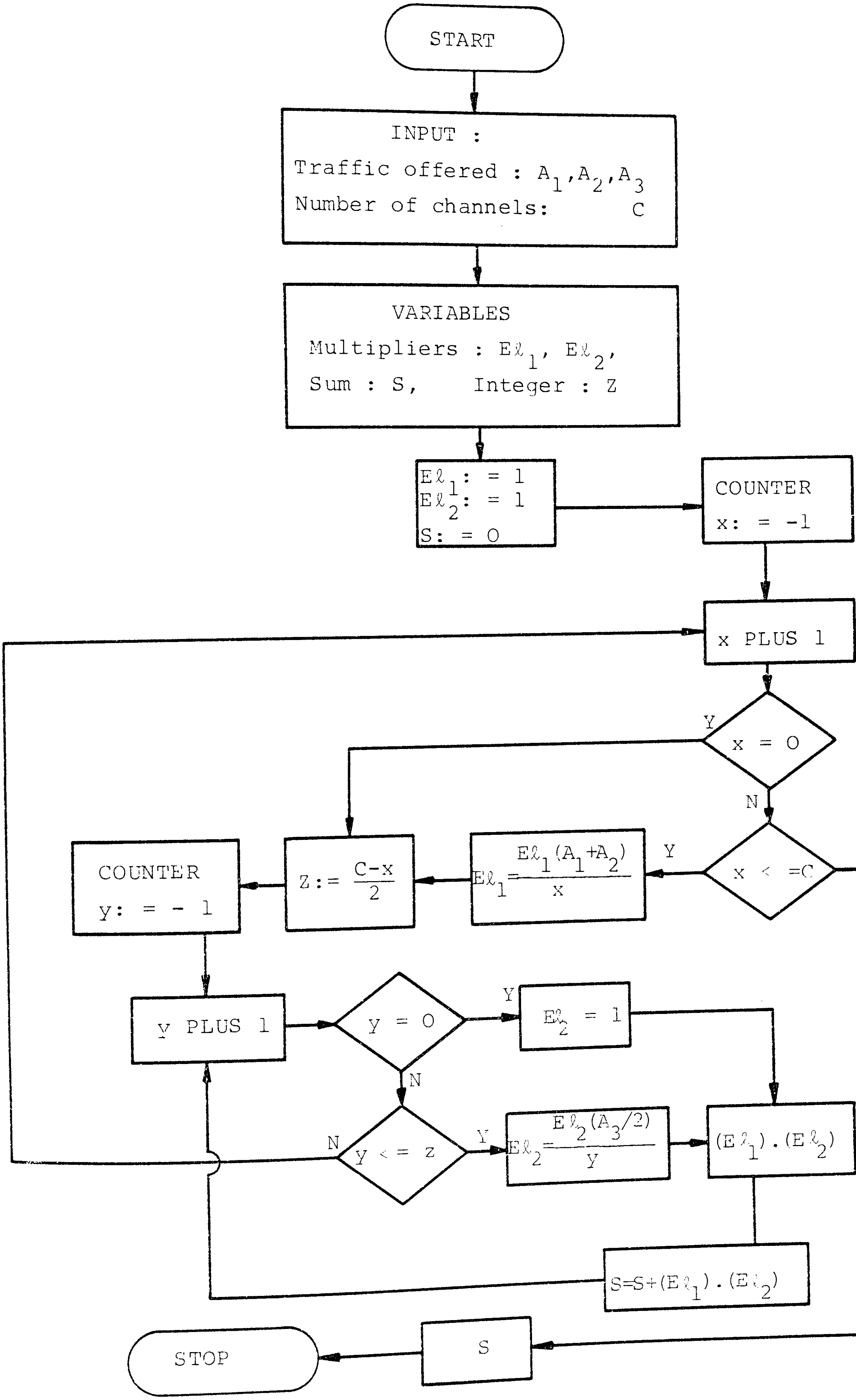

FIGURE (I.6)

Computing $\sum_{x=0}^{C} \sum_{y=0}^{<(C-x)/2>} \frac{(A_1+A_2)^x}{x!} \cdot \frac{(A_3/2)^y}{y!}$

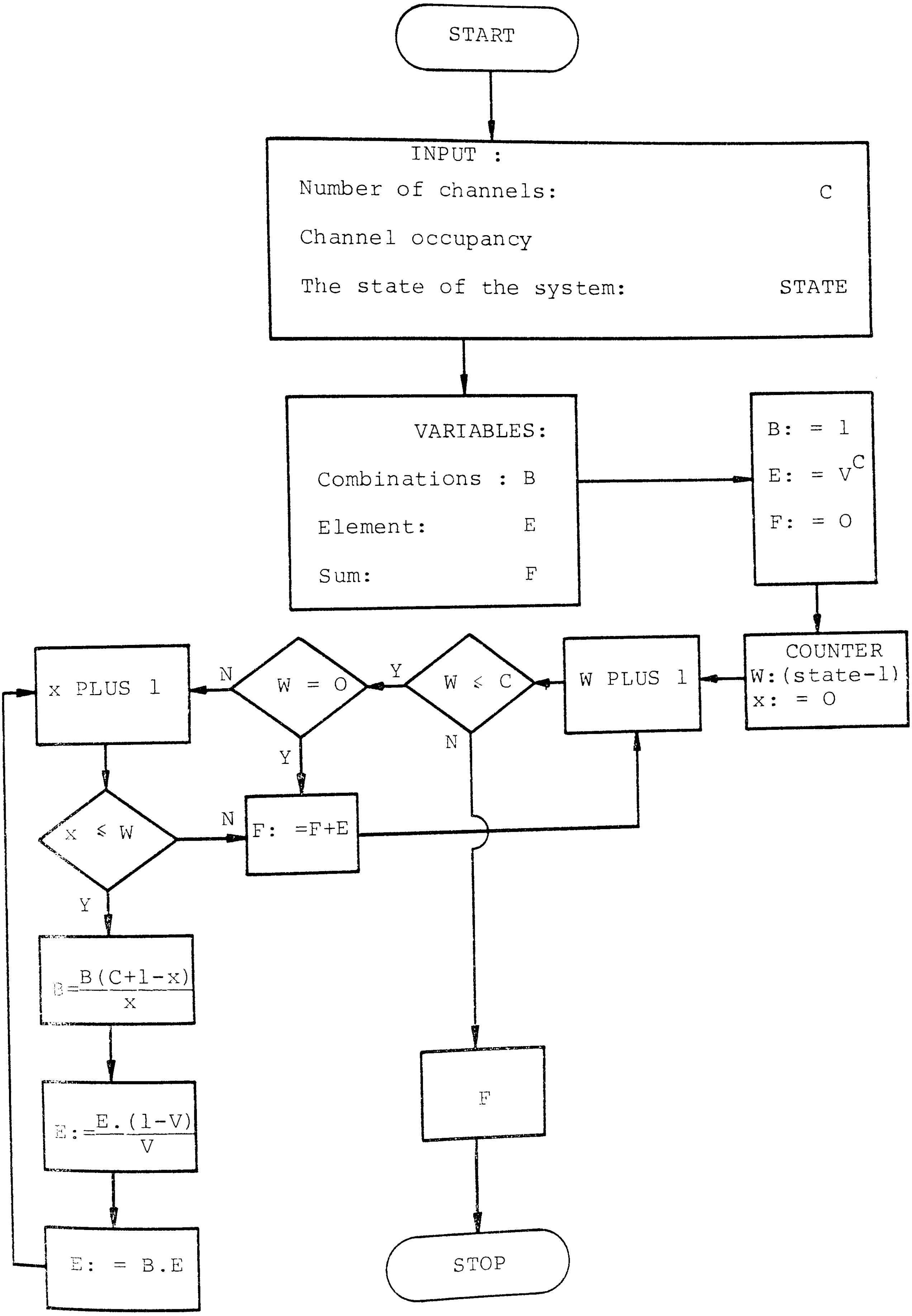

FIGURE (I.7)

Computing $F(\text{state}) = \sum_{W=\text{state}}^{C} \frac{C!}{W!(C-W)!} V^{C-W} (1-V)^{W}$

blocking for limited number of mobiles

```
LIBRARY 'CORAL.LIB'
LIBRARY 'CORALM.LIB'


BEGIN
        COMMENT computing the blocking states    ;

        FLOATING PROCEDURE blkstat

        ( VALUE INTEGER m , c   ;
          VALUE FLOATING a1 , a2 , a3   )   ;

        BEGIN

                INTEGER x , limit  ;
                FLOATING el1 , el2 , el , rs  ;
                el1 := 1   ;
                el2 := 1   ;
                limit := c / 2   ;

                FOR x:= 1 STEP 1 UNTIL limit DO
                        el2 := ( el2 * ( (m/2)-(c/2)+x ) * a3 ) / x   ;

                el := el1 * el2   ;
                rs:= el   ;

                FOR x := 1 STEP 1 UNTIL limit DO
                BEGIN
                        el1 := el1 * ( m-c+(2*x) ) * ( a1 + a2 )   ;
                        el1 := el1 * ( m-c+(2*x)-1) * ( a1 + a2 )   ;
                        el1:= el1 / ( 2 * x )   ;
                        el1 := el1 / ( (2*x) - 1 )   ;

                        el2 := el2 / ( ( (m/2) - x + 1 ) * a3 )   ;
                        el2 := el2 * ( (c/2) - x + 1 )   ;

                        el := el1 * el2   ;
                        rs := rs + el   ;
                END   ;
                ANSWER rs   ;
        END   ;

        COMMENT the states of one free channel   ;

        FLOATING PROCEDURE onestat

        ( VALUE INTEGER m , c   ;
          VALUE FLOATING a1 , a2 , a3   )   ;

        BEGIN

                INTEGER x , limit   ;
```

```
        FLOATING el1 , el2 , el , rs  ;
        el1 := a1 + a2  ;
        el1 := el1 * (m+2-c)  ;
        el2 := 1  ;
        limit := ( (c/2) - 1 )  ;

        FOR x:=1 STEP 1 UNTIL limit DO
                el2 := ( el2 * ( m-c+(2*x+1) ) * (a3/2) ) / x  ;

        el := el1 * el2  ;
        rs := el  ;

        FOR x:=1 STEP 1 UNTIL limit DO
        BEGIN
                el1 := ( el1*(m-c+2*x+2)*(a1+a2) ) / ( 2*x + 1 )  ;
                el1 := ( el1*(m-c+2*x+1)*(a1+a2) ) / (2*x)  ;

                el2 := el2 * ( (c/2) -x )  ;
                el2 := el2 / ( (m-(2*x+1)+2) * (a3/2) )  ;

                el := el1 * el2  ;
                rs := rs + el  ;

        END  ;
        ANSWER rs  ;
END  ;

COMMENT all possible states  ;

FLOATING PROCEDURE allstat

( VALUE INTEGER m , c ;
  VALUE FLOATING a1 , a2 , a3 )  ;

BEGIN

        FLOATING el1 , el2 , el , rs  ;
        FLOATING ARRAY e1[1:50]  ;
        INTEGER x , y , limit  ;
        el1 := 1  ;  el2 := 1  ;
        rs := 1  ;

        COMMENT states with no one channel calls  ;

        limit := (c/2)  ;
        FOR y := 1 STEP 1 UNTIL limit DO
        BEGIN
                el2 := ( el2 * (m+2-2*y) * (a3/2) ) / y  ;
                rs := rs + el2  ;
        END  ;

        COMMENT states with no two channel calls  ;

        FOR x := 1 STEP 1 UNTIL c DO
        BEGIN
                el1 := ( el1 * ( m+1-x ) * (a1+a2) ) / x  ;
                rs := rs + el1  ;
        END  ;

        COMMENT all other possible states  ;

        FOR y := 1 STEP 1 UNTIL 50 DO
                        e1[y] := 1  ;

        FOR x := 1 STEP 1 UNTIL c DO
        BEGIN
```

```
        limit := (c-x) / 2  ;

        IF limit >= 1
        THEN

        BEGIN

        el2 := 1  ;


        FOR y := 1 STEP 1 UNTIL limit DO
        BEGIN
                el[y] := (el[y]*(m+1-x-(2*y))*(a1+a2)) / x  ;

                el2 := (el2*(m-x+2-2*y)*(a3/2))/y  ;

                el := el[y] * el2  ;
                rs := rs + el  ;
        END  ;
        END  ;

        END  ;

        ANSWER rs  ;
END  ;


COMMENT define variables  ;

INTEGER freq , rounds  ;
INTEGER mob  ;
FLOATING tr1 , tr2 , tr3 , one , two , av  ;

COMMENT investigations  ;

FOR rounds := 1 STEP 1 UNTIL 10 DO

BEGIN

COMMENT read input  ;

newline ; newline ;
writetext('INPUT')  ;
newline ; newline ;
writetext('number of mobiles           :    ')  ;
mob := readnumber  ;
newline ;
writetext('number of frequencies       :    ')  ;
freq := readnumber  ;
newline ; newline ;
writetext('traffic offered by l-m calls  :    ')  ;
tr1 := readflo  ;
writetext('traffic offered by m-l calls  :    ') ;
tr2 := readflo ;
newline ;
writetext('traffic offered by m-m call   :    ')  ;
tr3 := readflo  ;
newline ; newline ;


newline ;
writetext('OUTPUT')  ;
newline ; newline ;
one := blkstat( mob , freq , tr1 , tr2 , tr3 )  ;
two := onestat ( mob , freq , tr1 , tr2 , tr3 )  ;
```

```
        av := allstat ( mob , free , tr1 , tr2 , tr3 ) ;
        writetext('blocking of one channel calls        :    ') ;
        one := one / av  ;
        writeflo (one , 6 , 6 )  ;
        newline ; newline ;
        writetext('blocking of two channel calls        :    ') ;
        two := one + ( two / av )  ;
        writeflo ( two , 6 , 6 )  ;
        newline ; newline ;
        writetext('average blocking of calls            :    ') ;
        av := ( (tr1+tr2) * one + (tr3 ) * two ) / ( tr1 + tr2 + tr3 )  ;
        writeflo ( av , 6 , 6 )  ;
        newline ; newline ;

        END  ;

END
FINISH
*
```

Number of Channels	Traffic Offered (Erlang)	Congestion		
		One-Channel Calls	Two-Channel Calls	Average
8	3.00 3.45 3.90 4.35 4.80	0.014809 0.024132 0.035993 0.050151 0.066246	0.045278 0.069316 0.097952 0.130170 0.164870	0.024965 0.039193 0.056646 0.076824 0.099121
12	6.00 6.60 7.20 7.80 8.40	0.017866 0.026941 0.038129 0.051245 0.066012	0.047809 0.069196 0.094382 0.122668 0.153254	0.027847 0.041026 0.056880 0.075053 0.095093
16	9.00 9.90 10.80 11.70 12.60	0.016875 0.027256 0.040508 0.056322 0.074231	0.042812 0.066426 0.095221 0.128135 0.163912	0.025521 0.040313 0.058746 0.080260 0.104125
20	12.00 13.20 14.40 15.60 16.80	0.015002 0.025878 0.040269 0.057776 0.077751	0.036977 0.061304 0.092059 0.127889 0.167118	0.022327 0.037686 0.057533 0.081147 0.107540

TABLE (I.1)

TRAFFIC-CONGESTION TABLE FOR A SINGLE-CELL SYSTEM WITH AN INFINITE NUMBER OF MOBILES AND EQUAL PROPORTIONS OF TYPE 1, TYPE 2 AND TYPE 3 TRAFFIC

Number Of Channels	Traffic Offered per Mobile (Erlang)	Congestion		
		One-Channel Calls	Two-Channel Calls	Average
8	0.0150 0.0165 0.0180 0.0195 0.0210	0.013015 0.018281 0.024604 0.031957 0.040280	0.040835 0.054944 0.071129 0.089167 0.108792	0.022288 0.030502 0.040113 0.051027 0.063117
12	0.0300 0.0330 0.0360 0.0390 0.0420	0.013480 0.020617 0.029604 0.040362 0.052723	0.037535 0.055065 0.076151 0.100337 0.127033	0.021499 0.032100 0.045119 0.060353 0.077493
16	0.0450 0.0495 0.0540 0.0585 0.0630	0.010379 0.017325 0.026604 0.038178 0.051842	0.027794 0.044520 0.065876 0.091402 0.120347	0.016184 0.026390 0.039695 0.055919 0.074677
20	0.0600 0.0660 0.0720 0.0780 0.0840	0.007116 0.013030 0.021503 0.032624 0.046236	0.018789 0.033020 0.052514 0.077042 0.105887	0.011007 0.019693 0.031840 0.047430 0.066120

TABLE (I.2)

TRAFFIC-CONGESTION TABLE FOR A SINGLE-CELL SYSTEM WITH 200 MOBILES AND EQUAL PROPORTIONS OF TYPE 1, TYPE 2 AND TYPE 3 TRAFFIC

Number of Channels	Proportion of Mobile-to-Mobile Calls	Traffic Offered (Erlang)	Congestion
8	0.50	3.00 3.45 3.90 4.35 4.80	0.034390 0.051289 0.071092 0.093172 0.116881
	0.25	3.00 3.45 3.90 4.35 4.80	0.020462 0.033237 0.049374 0.068467 0.089948
	0.10	3.00 3.45 3.90 4.35 4.80	0.012817 0.022738 0.036210 0.053060 0.072841
	0.01	3.00 3.45 3.90 4.35 4.80	0.008583 0.016617 0.028262 0.043544 0.062131

TABLE (I.3)

TRAFFIC-CONGESTION TABLE FOR A SINGLE-CELL SYSTEM WITH AN INFINITE NUMBER OF MOBILES AND DIFFERENT PROPORTIONS OF MOBILE-TO-MOBILE CALLS

Number of Channels Per Cell	Traffic Offered (Erlang)	Congestion	
		Dynamic Assignment	Fixed Assignment
2	0.10	0.000012	0.005786
	0.15	0.000112	0.011429
	0.20	0.000493	0.018531
	0.25	0.001481	0.026851
	0.30	0.003500	0.036179
	0.35	0.007003	0.046330
	0.40	0.012408	0.057144
	0.45	0.020046	0.068482
	0.50	0.030125	0.080223
	0.55	0.042722	0.092265
4	0.80	0.000475	0.008366
	0.85	0.000778	0.010079
	0.90	0.001223	0.011986
	0.95	0.001854	0.014089
	1.00	0.002721	0.016389
	1.05	0.003877	0.018884
	1.10	0.005380	0.021575
	1.15	0.007285	0.024457

TABLE (I.4)

TRAFFIC-CONGESTION TABLE FOR A MULTI-CELL SYSTEM USING FIXED OR DYNAMIC CHANNEL ASSIGNMENT. THE FREQUENCY REUSE BUFFERING AREA IS 6 CELLS. THE PROPORTION OF MOBILE-TO-MOBILE CALLS WITHIN A SINGLE-CELL IS 0.01

Number of Channels Per Cell	Number of Fixed Channels Per Cell	Number of Dynamic Channels Per Cell	Traffic Offered (Erlang)	Congestion
8	8	0	3.00 3.25 3.50 3.75 3.90 4.00	0.008583 0.012617 0.017728 0.023969 0.028262 0.031351
	6	2	3.00 3.25 3.50 3.75 3.90 4.00	0.000159 0.000816 0.003189 0.009848 0.017526 0.024780
	4	4	3.00 3.25 3.50 3.75 3.90 4.00	0.000055 0.000327 0.001415 0.004693 0.008640 0.012462

TABLE (I.5)

TRAFFIC-CONGESTION TABLE FOR A MULTI-CELL SYSTEM USING FIXED OR HYBRID CHANNEL ASSIGNMENT. THE FREQUENCY REUSE BUFFERING AREA IS 6 CELLS. THE PROPORTION OF MOBILE-TO-MOBILE CALLS WITHIN A SINGLE-CELL IS 0.01.

APPENDIX II

SIMULATION PROGRAMS

In order to give a flavour of the simulation programs described in this thesis, two programs are listed in the following. The first is related to the speech traffic of mobiles while the second is concerned with the computer control of mobiles.

A Coral 66 Program for the Simulation of the Speech Traffic of a Mobile Radio System

The Coral 66 program listed in the following simulates using the roulette method the speech traffic of mobiles for a single-cell system with a finite number of mobiles.

```
roulette simulation for mobile radio traffic

LIBRARY 'CORAL.LIB'
LIBRARY 'CORALM.LIB'

ABSOLUTE ( INTEGER traffic/OCTAL(167774) ) ;

BEGIN

      COMMENT roulette simulation for a mobile radio system ;



      COMMENT define procedures for the simulation of the system ;


      COMMENT ( 1 ) initiate available channles ( free channles ) ;

      PROCEDURE clear (INTEGER ARRAY n ; INTEGER k ) ;
            BEGIN
                  INTEGER i ;
                  FOR i:=1 STEP 1 UNTIL k DO n[i]:=0 ;
            END ;


      COMMENT ( 2 ) generate an integer random number of maximum (max) ;

      INTEGER PROCEDURE randn ( INTEGER max ) ;
            BEGIN
                  INTEGER i,j ;
                  COMMENT pick-up a random number from a storage location (abs.) ;
                  i:=traffic ;
                  FOR j:=1,j+1
                  WHILE i<=0 OR i>max DO
                  BEGIN i:=traffic END ;
                  ANSWER i ;
            END ;


      COMMENT ( 3 ) number of the first free channle ;

      INTEGER PROCEDURE serch ( INTEGER ARRAY n ; INTEGER k ) ;
            BEGIN
                  INTEGER i ;
                  FOR i:=0,i+1 WHILE n[i]<>0 AND i<=k DO ;
                  ANSWER i ;
            END ;


      COMMENT ( 4 ) number of the second free channle ;

      INTEGER PROCEDURE msrch ( INTEGER ARRAY n ; INTEGER k ) ;
            BEGIN
                  INTEGER i,j ;
                  i:=serch(n,k) ;
                  IF i>k
                  THEN j:=k+1
                  ELSE
```

```
            FOR j:=i,j+1
            WHILE n[j]<>0 AND j<=k DO  ;
        ANSWER j  ;
    END  ;


COMMENT ( 5 ) number of a channle busy in a one channle call  ;

INTEGER PROCEDURE srbsy ( INTEGER ARRAY n ; INTEGER k )  ;
    BEGIN
        INTEGER i  ;
        FOR i:=0,i+1 WHILE ( ( n[i]=2 OR n[i]=0 ) AND i<=k ) DO  ;
        ANSWER i  ;
    END  ;


COMMENT ( 6 ) number of a channle busy in a two channle call  ;

INTEGER PROCEDURE bsone ( INTEGER ARRAY n ; INTEGER k )  ;
    BEGIN
        INTEGER i  ;
        FOR i:=0,i+1 WHILE ( ( n[i]=1 OR n[i]=0 ) AND i<=k ) DO  ;
        ANSWER i  ;
    END  ;


COMMENT ( 7 ) number of another channle busy in a two channle call  ;

INTEGER PROCEDURE bstwo ( INTEGER ARRAY n ; INTEGER k )  ;
    BEGIN
        INTEGER i,j  ;
        i:=bsone(n,k)  ;
        IF i>=k
        THEN j:=k+1
        ELSE
            FOR j:=i,j+1
            WHILE ( ( n[j]=1 OR n[j]=0 ) AND j<=k ) DO  ;
        ANSWER j  ;
    END  ;



COMMENT the main procedure  ;


COMMENT simulation of the main events of the system  ;

PROCEDURE mobil
( INTEGER ARRAY n  ;
  INTEGER k,m,a1,a2,a3,d  ;
  LOCATION INTEGER s1,s2,s3,b1,b2,b3,x1,x2,x3 )  ;
    BEGIN
        INTEGER i,j  ;
        INTEGER ran,max,cnst,x,l1,l2,l3,l4,l5  ;
        cnst:=600  ;
        x:=x1+x2+2*x3  ;
        l1:=x1*cnst  ;
        l2:=l1+x2*cnst  ;
        l3:=l2+x3*cnst  ;
        l4:=l3+(m-x)*a1*d  ;
        l5:=l4+(m-x)*a2*d  ;
        max:=l5+(m-x)*a3*d  ;
        ran:=rangn(max)  ;

        COMMENT release a land-mobile call  ;
```

```
IF ran<=11
THEN
     BEGIN
          i:=srbss(n,k)   ;
          n[i]:=0   ;
          x1:=x1-1   ;
     END   ;

COMMENT release a mobile-land call   ;

IF ran>11 AND ran<=12
THEN
     BEGIN
          i:=srbss(n,k)   ;
          n[i]:=0   ;
          x2:=x2-1   ;
     END   ;

COMMENT release a mobile-mobile call   ;

IF ran>12 AND ran<=13
THEN
     BEGIN
          i:=bsone(n,k)   ;
          j:=bstwo(n,k)   ;
          n[i]:=0   ;
          n[j]:=0   ;
          x3:=x3-1   ;
     END   ;

COMMENT set-up a land-mobile call   ;

IF ran>13 AND ran<=14
THEN
     BEGIN
          i:=serch(n,k)   ;
          IF i<=k
          THEN
               BEGIN
               n[i]:=1   ;
               s1:=s1+1   ;
               x1:=x1+1   ;
               END
          ELSE
               b1:=b1+1   ;
     END   ;

COMMENT set-up a mobile-land call   ;

IF ran>14 AND ran<=15
THEN
     BEGIN
          i:=serch(n,k)   ;
          IF i<=k
          THEN
               BEGIN
               n[i]:=1   ;
               s2:=s2+1   ;
               x2:=x2+1   ;
               END
          ELSE
               b2:=b2+1   ;
     END   ;

COMMENT set-up a mobile-mobile call   ;
```

```
        IF ran>15 AND ran<=max
        THEN
             BEGIN
                  i:=serch(n,k)   ;
                  j:=msrch(n,k)   ;
                  IF i<k AND j<=k
                  THEN
                       BEGIN
                       n[i]:=2   ;
                       n[j]:=2   ;
                       s3:=s3+1   ;
                       x3:=x3+1   ;
                       END
                  ELSE
                       b3:=b3+1   ;
             END ;
     END  ;
COMMENT allocate space for posible channles  ;

INTEGER ARRAY t[1:100]  ;

COMMENT define system requirements  ;

INTEGER chn,mbl,ar1,ar2,ar3,drn,rnd  ;
INTEGER sc1,sc2,sc3,bl1,bl2,bl3,oc1,oc2,oc3,p  ;

COMMENT read system description  ;

newline  ;  newline  ;  newline  ;
writetext('SYSTEM DESCRIPTION')  ;
newline  ;  newline  ;  newline  ;
writetext('number of available channles     :      ')  ;
chn:=readnumber  ;
newline  ;
writetext('number of mobiles                :      ')  ;
mbl:=readnumber  ;
newline  ;
writetext('rate of land-mobile calls        :      ')  ;
ar1:=readnumber  ;
writetext('(1/10)call/free mobile/hour')  ;
newline  ;  newline  ;
writetext('rate of mobile-land calls        :      ')  ;
ar2:=readnumber  ;
writetext('(1/10)call/free mobile/hour')  ;
newline  ;  newline  ;
writetext('rate of mobile-mobile calls      :      ')  ;
ar3:=readnumber  ;
writetext('(1/10)call/free mobile/hour')  ;
newline  ;  newline  ;
writetext('average call duration            :      ')  ;
drn:=readnumber  ;
writetext('minutes  ')  ;
newline  ;  newline  ;
writetext('the system will run for          :      ')  ;
rnd:=readnumber  ;
writetext('rounds')  ;
newline  ;  newline  ;  newline  ;  newline  ;


COMMENT initiate performance measures  ;

sc1:=0  ;  sc2:=0  ;  sc3:=0  ;
bl1:=0  ;  bl2:=0  ;  bl3:=0  ;
```

```
        COMMENT initiate counters of calls in progress  ;

        oc1:=0  ;  oc2:=0  ;  oc3:=0   ;


        COMMENT initiate available channles  ;

        clear(t,chn)  ;


        COMMENT system at work  ;

        FOR p:=1 STEP 1 UNTIL rnd DO
        mobil(t,chn,mbl,ar1,ar2,ar3,drn,sc1,sc2,sc3,bl1,bl2,bl3,oc1,oc2,oc3)  ;



        COMMENT performance measurements  ;

        newline  ;  newline  ;  newline  ;
        writetext('PERFORMANCE MEASUREMENTS')  ;
        newline  ;  newline  ;  newline  ;
        writetext('land-mobile calls')  ;
        newline  ;
        writetext('successful      :      ')  ;
        writenumber(sc1)  ;
        newline  ;
        writetext('blocked         :      ')  ;
        writenumber(bl1)  ;
        newline  ;  newline  ;
        writetext('mobile-land calls')  ;
        newline  ;
        writetext('successful      :      ')  ;
        writenumber(sc2)  ;
        newline  ;
        writetext('blocked         :      ')  ;
        writenumber(bl2)  ;
        newline  ;  newline  ;
        writetext('mobile-mobile calls')  ;
        newline  ;
        writetext('successful      :      ')  ;
        writenumber(sc3)  ;
        newline  ;
        writetext('blocked         :      ')  ;
        writenumber(bl3)  ;


    END
    FINISH
    *
```

An Algol 68 Program for the Simulation of a Computer Control Network

The Algol 68 program listed in the following simulates the inter-city computer control network used for the address-changing of mobiles travelling between cities as described in chapter 4. The program consists of ten parts:

1. Network description.
2. Traffic generation procedures.
3. System initiation procedures.
4. Reporting procedures.
5. Packet handling procedures.
6. Routing procedures.
7. Communication procedures.
8. Input-Output interfacing procedures.
9. Network function procedure.
10. The body of the program.

NTER CITY COMPUTER NETWORK

```
BEGIN'

        'C'      PART  : :  ONE                                          'C'

        'C'      NETWORK  DESCRIPTION                                    'C'

        'INT' N , ISZ , OSZ , RUN  ;
        'REAL' RATE  ;
        'C' N     :  NUMBER  OF  INTER CONNECTED  COMPUTERS              'C'
        'C' ISZ   :  SIZE  OF  INPUT  BUFFER  QUEUE                      'C'
        'C' OSZ   :  SIZE  OF  OUTPUT  BUFFER  QUEUE                     'C'
        'C' RUN   :  EXECUTION  TIME                                     'C'
        'C' RATE  :  RATE  OF  PACKET  ARRIVAL  AT  EACH  CITY  NODE'C'
        READ ((
        N , NEWLINE ,
        ISZ , NEWLINE ,
        OSZ , NEWLINE ,
        RUN , NEWLINE ,
        RATE , NEWLINE , NEWLINE )) ;

        'MODE''PACKET' = 'STRUCT'('CHAR' SOURCE      ,
                                         DESTINATION ,
                                         TRANSIT ,
                                  'INT'  COUNTER ,
                                         TIME ,
                                         WATCH          ) ;

        'MODE''PAGE'   = 'STRUCT'('CHAR' CH1 , CH2 , CH3 ,
                                  'INT'  CN1 , CN2 , CN3  ) ;

        'MODE''INPUT' = 'STRUCT' ( 'INT' ARRIVAL ,
                                         MAX     ,
                                  'REAL'SUC      ,
                                         BLK     ,
                                         IDL     ,
                                  [1:ISZ]'PAGE' BUFFER  ) ;

        'MODE''OUTPUT' = 'STRUCT' ( 'CHAR' LINK ,
                                    'INT'  MAX  ,
                                    'REAL' SUC  ,
                                           BLK  ,
                                           IDL  ,
                                           CHSUC  ,
                                           CHBLK  ,
                                           CHIDL  ,
                                    [1:OSZ]'PAGE' BUFFER  ) ;

        'MODE''MEASUREMENT' = 'STRUCT' ( 'REAL'TRAFSUC ,
                                               TRAFBLK ,
                                               TRAFTRN ,
                                               TRAFARV ,
                                               CLOCK   ,
                                               DELAY   ,
                                               PATH    ,
                                               OPERATION  ) ;

        'MODE''CITY' = 'STRUCT' ( 'CHAR' NAME  ,
                                  'INPUT' INPUT ,
                                  [1:3]'OUTPUT'OUTPUT ,
                                  'MEASUREMENT' MEASUREMENT  ) ;

        'MODE''PERFORMANCE' = 'STRUCT' ( 'REAL' AVDELAY ,
                                                AVPASS ,
                                                EFFICIENCY ,
                                                UTILIZATION  ) ;

        'MODE''NETWORK' = 'STRUCT' ( [1:N]'CITY' CITY ,
                                     'PERFORMANCE' PERFORMANCE  ) ;

        INT 'OF' NUMBERSTYLE := 4  ;

                                                                         'C'
        'C'      PART  : : TWO
                                                                         'C'
        'C'      TRAFFIC  GENERATION  PROCEDURES
```

```
        'C'    GENERATE  A  UNIFORM  RANDOM  INTEGER                    'C'
        'PROC' UNRAN = ( 'INT' MAX , MIN  )'INT':
             (
             'INT' K  ;
             K := 0  ;
             'WHILE' K <= MIN 'DO'
             K := 'ENTIER'(MAX*RANDOM+1)  ;
             K
             )  ;

        'PROC' EXRAN = ( 'INT' AVERAGE )'INT':
             (
             'INT' RN  ;
             'REAL' RL  ;
             RL := -LN ( RANDOM )  ;
             RN := 'ENTIER' ( AVERAGE * RL +1 )  ;
             RN
             )  ;


        'PROC' POSAR =( 'INT' CLOCK , 'REAL' RATE )'INT':
             (
             'INT' ARRIVAL  ;
             ARRIVAL := CLOCK + 5  ;
             ARRIVAL
             )  ;

        'PROC' NEWAR = ( 'REF''INPUT' INPUT ,
                         'INT' CLOCK , 'REAL' RATE ) 'VOID':
            (
             ARRIVAL 'OF' INPUT := POSAR ( CLOCK , RATE )
             )  ;


        'PROC' DESTN = ( 'CHAR' SOURCE , 'INT' N ) 'CHAR':
             (
             'INT' UR  ;
             'CHAR' DESTINATION  ;
             DESTINATION := SOURCE  ;
             'WHILE' DESTINATION  = SOURCE 'DO'
             (
             UR := UNRAN(N+32 , 32)  ;
             DESTINATION := 'REPR'( UR )
             )  ;
             DESTINATION
             )  ;

        'PROC' PKGEN = ( 'CHAR' SOURCE , 'INT' N , CLOCK )'PACKET':
             (
             'PACKET' PACKET  ;
             SOURCE 'OF' PACKET := SOURCE  ;
             DESTINATION 'OF' PACKET := DESTN(SOURCE,N)  ;
             TRANSIT 'OF' PACKET := SOURCE  ;
             COUNTER 'OF'PACKET := 1  ;
             TIME 'OF' PACKET := CLOCK  ;
             WATCH 'OF' PACKET := CLOCK  ;
             PACKET
             )  ;




         'C'    PART  : : THREE                                          'C'

         'C'    SYSTEM  INITIATION  PROCEDURES                           'C'
```

```
'PROC' CLPAG = ( 'REF''PAGE' PAGE ) 'VOID':
     (
     PAGE := ( "X" , "X" , "X" , 0 , 0 , 0 )
     )  ;

'PROC' CLBUF = ( 'REF'[ ] 'PAGE' BUFFER )'VOID':
     (
     'FOR' I 'FROM' 'LWB' BUFFER 'TO' 'UPB' BUFFER 'DO'
     CLPAG ( BUFFER[I] )
     )  ;


'PROC' CLINP = ('REF''INPUT' INPUT ) 'VOID':
     (
     ARRIVAL 'OF' INPUT :=0  ;
     SUC 'OF' INPUT := 0  ;
     BLK 'OF' INPUT := 0  ;
     IDL 'OF' INPUT := 0  ;
     MAX 'OF' INPUT := 0  ;
     CLBUF ( BUFFER 'OF' INPUT )
     )  ;

'PROC' CLMSR = ( 'REF''MEASUREMENT' M )'VOID':
     (
     M := ( 0 , 0 , 0 , 0 , 0 , 0 , 0 , 0 )
     )  ;

'PROC' INOTP = ( 'REF''OUTPUT' OUTPUT ) 'VOID':
     (
     READ (( LINK 'OF' OUTPUT , NEWLINE ))  ;
     SUC 'OF' OUTPUT := 0  ;
     BLK 'OF' OUTPUT := 0  ;
     IDL 'OF' OUTPUT := 0  ;
     CHSUC 'OF' OUTPUT := 0  ;
     CHBLK 'OF' OUTPUT := 0  ;
     CHIDL 'OF' OUTPUT := 0  ;
     MAX   'OF' OUTPUT := 0  ;
     CLBUF ( BUFFER 'OF' OUTPUT )
     )  ;

'PROC' INCTY = ( 'REF''CITY' CITY , 'INT' IDN , 'REAL' RATE )
     (
     NAME 'OF' CITY := 'REPR' ( IDN + 32 )  ;
     CLINP ( INPUT 'OF' CITY )  ;
     NEWAR ( INPUT 'OF' CITY , 0 , RATE )  ;
     'FOR' I 'TO' 3 'DO'
     INOTP (  ( OUTPUT 'OF' CITY )[I]  )  ;
     READ ( NEWLINE )  ;
     CLMSR ( MEASUREMENT 'OF' CITY )
     )  ;

'PROC' CLPER = ( 'REF''PERFORMANCE' P ) 'VOID':
     (
     P := ( 0.0 , 0.0 , 0.0 , 0.0  )
     )  ;
'PROC' INNET = ( 'REF''NETWORK' NETWORK ,
                 'INT' N , 'REAL' RATE ) 'VOID':
     (
     'FOR' I 'TO' N 'DO'
     INCTY ( (CITY'OF'NETWORK)[I] , I , RATE )  ;
     CLPER ( PERFORMANCE 'OF' NETWORK )
     )  ;
```

```
        'C'     PART  : : FOUR                                          'C'

        'C'     SYSTEM  REPORTING  PROCEDURES                           'C'


        'PROC' PKSTR = 'VOID':
             (
             PRINT ((
             "THE STRUCTURE OF A PACKET " , NEWLINE ,
             "************************" , NEWLINE , NEWLINE ,
             "SOURCE NODE ", NEWLINE , NEWLINE ,
             "DESTINATION NODE " , NEWLINE , NEWLINE ,
             "TRANSIT NODE " , NEWLINE , NEWLINE ,
             "NUMBER OF NODES PASSED " , NEWLINE , NEWLINE ,
             "TIME OF BIRTH " , NEWLINE , NEWLINE ,
             "TIME NOW " , NEWLINE , NEWLINE , NEWLINE  ))
             )  ;

        'PROC' RPBUF = ( [ ]'PAGE' BUFFER ) 'VOID':
             (
             PRINT ((
             "THE CONTENT OF THE BUFFER ",NEWLINE,
             "**************************",NEWLINE,NEWLINE ))  ;
             'FOR' I 'FROM' 'LWB' BUFFER 'TO' 'UPB' BUFFER 'DO'
             PRINT ((  BUFFER[I] , NEWLINE ))  ;
             PRINT ((NEWLINE , NEWLINE , NEWLINE ))
             )  ;

        'PROC' RPINP = ( 'INPUT' INPUT ) 'VOID':
             (
             PRINT ((
             "THE CONTENT OF THE INPUT ",NEWLINE,
             "************************",NEWLINE , NEWLINE ,
             "THE EXPECTED TIME FOR A PACKET ARRIVAL  :  ",
             ARRIVAL 'OF' INPUT ,NEWLINE , NEWLINE  ,
             NEWLINE  ))  ;
             PRINT ((
             "NUMBER OF SUCCESSFUL  OPERATIONS           :  ",
             SUC 'OF' INPUT , NEWLINE , NEWLINE ,
             "NUMBER OF BLOCKED OPERATIONS               :  ",
             BLK 'OF' INPUT , NEWLINE , NEWLINE ,
             "NUMBER OF EVENTS ( NO OPERATION )          :  ",
             IDL 'OF' INPUT , NEWLINE , NEWLINE ,
             "MAXIMUM BUFFER SIZE REQUIRED               :  ",
             MAX 'OF' INPUT , NEWLINE , NEWLINE  ))  ;
             RPBUF ( BUFFER 'OF' INPUT )
             )  ;

        'PROC' RPOTP = ( 'OUTPUT' OUTPUT ) 'VOID':
             (
             PRINT ((
             "THE OUTPUT IS LINKED TO   :   ",LINK 'OF' OUTPUT , NEWLINE ,
             "**********************************",NEWLINE , NEWLINE))  ;
             PRINT (( NEWLINE ,
             "NUMBER OF SUCCESSFUL OPERATIONS           :  ",
             SUC 'OF' OUTPUT , NEWLINE , NEWLINE ,
             "NUMBER OF BLOCKED OPERATIONS              :  ",
             BLK 'OF' OUTPUT , NEWLINE , NEWLINE ,
             "NUMBER OF EVENTS ( NO OPERATION )         :  ",
             IDL 'OF' OUTPUT , NEWLINE , NEWLINE ,
             "MAXIMUM BUFFER SIZE REQUIRED              :  ",
             MAX 'OF' OUTPUT , NEWLINE , NEWLINE ,
             "NUMBER OF SUCCESSFUL TRANSMISSIONS        :  ",
             CHSUC 'OF' OUTPUT , NEWLINE , NEWLINE ,
```

```
          "NUMBER OF BLOCKED TRANSMISSIONS          :  ",
          CHBLK 'OF' OUTPUT , NEWLINE , NEWLINE ,
          "NUMBER OF EVENTS ( NO TRANSMISSION )      :  ",
          CHIDL 'OF' OUTPUT , NEWLINE , NEWLINE ,
          "CHANNEL UTILIZATION                       :  ",
          ( (CHSUC'OF'OUTPUT)+(CHBLK'OF'OUTPUT) ) /
          ( (CHSUC'OF'OUTPUT)+(CHBLK'OF'OUTPUT)+(CHIDL'OF'OUTPUT) ) ,
          NEWLINE , NEWLINE , NEWLINE  )) ;
          RPBUF ( BUFFER 'OF' OUTPUT )
          ) ;


     'PROC' RPMSR = ( 'REF''MEASUREMENT' M , 'INT' CLOCK ) 'VOID':
          (
          CLOCK 'OF' M := CLOCK  ;
          PRINT ((
          "MEASUREMENTS", NEWLINE ,
          "************", NEWLINE , NEWLINE ,
          "TRAFFIC OFFERED                   :  ",
          ( TRAFSUC 'OF' M ) + ( TRAFBLK 'OF' M ) , NEWLINE , NEWLINE ,
          "TRAFFIC BLOCKED                   :  " ,
          TRAFBLK 'OF' M , NEWLINE , NEWLINE ,
          "RATE OF TRAFFIC OFFERED           :  " ,
          ( (TRAFSUC'OF'M)+(TRAFBLK'OF'M) ) / CLOCK 'OF' M ,
          NEWLINE , NEWLINE ,
          "NODE THROUGHPUT                   :  " ,
          ( TRAFTRN 'OF' M ) + ( TRAFARV 'OF' M ), NEWLINE , NEWLINE ,
          "RATE OF THROUGHPUT                :  " ,
          ( (TRAFTRN'OF'M)+(TRAFARV'OF'M) ) / CLOCK , NEWLINE , NEWLINE ,
          "AVERAGE PACKET DELAY              :  " ,
          ( DELAY 'OF' M ) / ( TRAFARV 'OF' M ) , NEWLINE , NEWLINE ,
          "AVERAGE NUMBER OF NODES PASSED    :  " ,
          ( PATH  'OF' M ) / ( TRAFARV 'OF' M ) , NEWLINE , NEWLINE ,
          NEWLINE  ))
          ) ;

     'PROC' RPCTY = ( 'REF''CITY' CITY , 'INT' CLOCK ) 'VOID':
          (
          RPMSR ( MEASUREMENT 'OF' CITY , CLOCK ) ;
               PRINT (( NEWLINE , NEWLINE , NEWLINE , NEWLINE , NEWLINE))
          ) ;

     'PROC' RPNET = ( 'REF''NETWORK'NETWORK ,
                      'INT' N,ISZ , OSZ , CLOCK , 'REAL' RATE ) 'VOID':
          (
          PRINT ((
          "COMPUTER COMMUNICATION NETWORK UNDER INVESTIGATIONS",NEWLINE,
          "***************************************************"  ,
          NEWLINE , NEWLINE , NEWLINE ,
          "RATE OF TRAFFIC PER CITY          :  ",
          RATE , NEWLINE , NEWLINE ,
          "TIME  NOW                         :  " ,
          CLOCK , NEWLINE , NEWLINE , NEWLINE )) ;
          'FOR' I 'TO'N 'DO'
          RPCTY ( (CITY 'OF' NETWORK)[I] , CLOCK )  ;
          PRINT ((
               NEWLINE ))
          ) ;



     'C'     PART  : : FIVE                                        'C'

     'C'     PACKET  HANDLING  AND  MONITORING  PROCEDURES         'C'

```

```
    'PROC' TSTEP =  ( 'PAGE' PAGE ) 'BOOL':
         (
         'IF'
         CH1 'OF' PAGE = "X"   'AND'
         CH2 'OF' PAGE = "X"   'AND'
         CH3 'OF' PAGE = "X"   'AND'
         CN1 'OF' PAGE =  0    'AND'
         CN2 'OF' PAGE =  0    'AND'
         CN3 'OF' PAGE =  0
         'THEN' 'TRUE'
         'ELSE' 'FALSE'
         'FI'
         )  ;

    'PROC' TSTRP = ( 'PAGE' PAGE , 'INT' CLOCK ) 'BOOL':
         (
         'IF'
         TSTEP (PAGE) = 'FALSE'  'AND'
         CN3 'OF' PAGE <= CLOCK
         'THEN' 'TRUE'
         'ELSE' 'FALSE'
         'FI'
         )  ;

    'PROC' TSTBL = ( [ ]'PAGE' BUFFER )'BOOL':
         (
         'IF'
         TSTEP ( BUFFER [ 'UPB' BUFFER ] ) = 'FALSE'
         'THEN' 'TRUE'
         'ELSE' 'FALSE'
         'FI'
         )  ;

    'PROC' PKFPG = ( 'REF''PAGE' PAGE ) 'PACKET':
         (
         'PACKET' PACKET  ;
         PACKET := PAGE 'AS' 'PACKET'  ;
         CLPAG ( PAGE )  ;
         PACKET
         )  ;

    'PROC' SHIFT = ( 'REF'[ ]'PAGE' BUFFER ) 'VOID':
         (
         'FOR' I 'FROM' 'LWB' BUFFER 'TO' ( 'UPB' BUFFER - 1 ) 'DO'
         'IF'
         TSTEP ( BUFFER[I] )
         'THEN'
         BUFFER[I] := BUFFER[I+1]  ;
         CLPAG ( BUFFER[I+1] )
         'FI'
         )  ;

    'PROC' EMPAG = ( [ ]'PAGE' BUFFER ) 'INT':
         (
         'INT' T  ;
         T := 'UPB' BUFFER + 1  ;
         'FOR' I 'FROM' 'UPB' BUFFER 'BY' -1 'TO' 'LWB' BUFFER
         'WHILE' TSTEP ( BUFFER[I] ) 'DO' T := I  ;
         T
         )  ;

    'PROC' PKIPG = ( 'PACKET' PACKET , 'REF'[ ]'PAGE' BUFFER ,
                     'REF''INT' MAX ) 'VOID':
         (
         'INT' F  ;
         F := EMPAG ( BUFFER )  ;
```

```
          BUFFER [ F ] := PACKET 'AS' 'PAGE'  ;
          'IF' F > MAX
          'THEN' MAX := F
          'FI'
          )  ;


     'PROC' PKTIM = ( 'REF''PACKET' PACKET , 'INT' CLOCK ) 'VOID':
          (
          WATCH 'OF' PACKET := CLOCK + 1
          )  ;

     'PROC' PKCNT = ( 'REF''PACKET' PACKET ) 'VOID':
          (
          COUNTER 'OF' PACKET 'PLUS' 1
          )  ;

     'PROC' PKTRN = ( 'REF''PACKET' PACKET , 'CHAR' TRN ) 'VOID':
          (
          TRANSIT 'OF' PACKET  :=  TRN
          )  ;

     'PROC' COUNT = ( 'REF''REAL'COUNTER ) 'VOID':
          (
          COUNTER 'PLUS' 1
          )  ;

     'PROC' PKDLY = ( 'PACKET' PACKET ) 'INT':
          (
          WATCH 'OF' PACKET - TIME 'OF' PACKET
          )  ;



     'C'     PART  :  SIX                                          'C'

     'C'     ROUTING  PROCEDURES                                   'C'


     'PROC' TSTRT = ( 'CITY' CITY , 'INT' CLOCK , 'CHAR' LINK )'BOOL':
          (
          'BOOL' TEST := 'FALSE'  ;
          'FOR' I 'TO' 'UPB' ( BUFFER 'OF' ( INPUT 'OF' CITY ) )
          'WHILE'
          TSTRP ( (BUFFER 'OF' (INPUT 'OF' CITY))[I],CLOCK) 'AND'
          TEST = 'FALSE'
          'DO'
          'IF'
          CH2 'OF'(BUFFER 'OF' (INPUT 'OF' CITY))[I]
          # NAME 'OF' CITY
          'THEN'
          TEST := 'TRUE'
          'ELSE' 'SKIP'
          'FI'  ;
          TEST
          )  ;

     'PROC' PKFRT = ( 'REF''CITY' CITY ,
                          'INT' CLOCK , 'CHAR' LINK ) 'PACKET':
          (
          'BOOL' FOUND := 'FALSE'  ;
          'PACKET' PACKET := ("X","X","X",0,0,0)  ;
          'FOR' I 'TO' 'UPB' ( BUFFER 'OF' (INPUT'OF'CITY) )
          'WHILE'
          TSTRP ( (BUFFER 'OF' (INPUT'OF'CITY) )[I] , CLOCK )
          'AND'
```

```
          FOUND = 'FALSE'
          'DO'
          'IF'
          CH2 'OF' ( BUFFER 'OF' (INPUT'OF'CITY) )[I]
          # NAME 'OF' CITY
          'THEN'
          PACKET := PKFPG ( (BUFFER 'OF' (INPUT'OF'CITY) )[I] )  ;
          PKTIM ( PACKET , CLOCK )  ;
          PKTRN ( PACKET , NAME 'OF' CITY )  ;
          SHIFT ( BUFFER 'OF' (INPUT'OF'CITY) )  ;
          FOUND := 'TRUE'
          'ELSE'
          'SKIP'
          'FI'  ;
          PACKET
          )  ;


      'PROC' SQROT = ( 'REF''CITY' CITY ,
                       'INT' CLOCK ) 'VOID':
          (
          'PACKET' PACKET  ;
          'FOR' I 'TO' 'UPB' ( OUTPUT 'OF' CITY )
          'DO'
          'IF'
          TSTRT ( CITY , CLOCK , LINK 'OF' (OUTPUT'OF'CITY)[I] )
          'THEN'
          'IF'
          TSTBL ( BUFFER 'OF' (OUTPUT'OF'CITY)[I] )
          'THEN'
          COUNT ( BLK 'OF' (OUTPUT'OF'CITY)[I] )
          'ELSE'
          PACKET :=
          PKFRT ( CITY , CLOCK , LINK 'OF' (OUTPUT'OF'CITY)[I] )  ;
          PKIPG ( PACKET , BUFFER 'OF' (OUTPUT'OF'CITY)[I] ,
                           MAX    'OF' (OUTPUT'OF'CITY)[I] )  ;
          COUNT ( SUC 'OF' (OUTPUT'OF'CITY)[I] )  ;
          COUNT ( TRAFTRN 'OF' (MEASUREMENT'OF'CITY) )
          'FI'
          'ELSE'
          COUNT ( IDL 'OF' (OUTPUT'OF'CITY)[I] )
          'FI'
          )  ;


      'C'    PART  : : SEVEN                                          'C'

      'C'    COMMUNICATION  PROCEDURES                                'C'


      'PROC' CTCTY = ( 'REF''OUTPUT' TRANSMITTER ,
                       'REF''INPUT'  RECEIVER    ,
                       'INT' CLOCK ) 'VOID':
          (
          'PACKET' PACKET  ;
          'IF'
          TSTRP ( (BUFFER 'OF' TRANSMITTER)['LWB'(BUFFER'OF'TRANSMITTER)]
                , CLOCK )
          'THEN'
          'IF'
          TSTEP( (BUFFER'OF'RECEIVER)['UPB'(BUFFER'OF'RECEIVER)] )
          'THEN'
          PACKET := PKFPG ( (BUFFER'OF'TRANSMITTER)['LWB'
                             (BUFFER'OF'TRANSMITTER)]  )  ;
          SHIFT ( BUFFER 'OF' TRANSMITTER )  ;
          PKTIM ( PACKET , CLOCK )  ;
```

```
              PKCNT ( PACKET )  ;
              COUNT ( SUC 'OF' RECEIVER )  ;
              COUNT ( CHSUC 'OF' TRANSMITTER )  ;
              PKIPG ( PACKET , BUFFER 'OF' RECEIVER ,
                               MAX    'OF' RECEIVER )
              'ELSE'
              COUNT ( BLK 'OF' RECEIVER )  ;
              COUNT ( CHBLK 'OF' TRANSMITTER )
              'FI'
              'ELSE'
              COUNT ( IDL 'OF' RECEIVER )  ;
              COUNT ( CHIDL 'OF' TRANSMITTER )
              'FI'
              )  ;


        'PROC' FRWRD = ( 'REF''CITY' TRANSMITTER ,
                         'REF''NETWORK' NET ,
                         'INT' CLOCK ) 'VOID':
              (
              'FOR' I 'TO' 'UPB' ( OUTPUT 'OF' TRANSMITTER )
              'DO'
              'FOR' J 'TO' 'UPB' ( CITY 'OF' NET )
              'DO'
              'IF'
              LINK 'OF' ( OUTPUT 'OF' TRANSMITTER )[I]
              = NAME 'OF' ( CITY 'OF' NET )[J]
              'THEN'
              CTCTY ( (OUTPUT 'OF' TRANSMITTER)[I] ,
                      INPUT 'OF' (CITY'OF'NET)[J] ,
                      CLOCK )
              'ELSE'
              'SKIP'
              'FI'
              )  ;



        'C'      PART  : : EIGHT                                          'C'

        'C'      INPUT / OUTPUT  INTERFACING  PROCEDURES                  'C'


        'PROC' TRFIN = ( 'REF''CITY' CITY ,
                         'INT' N , CLOCK , 'REAL' RATE ) 'VOID':
              (
              'PACKET' PACKET  ;
              'IF'
              ARRIVAL 'OF' ( INPUT 'OF' CITY ) = CLOCK
              'THEN'
                   NEWAR ( INPUT 'OF' CITY , CLOCK , RATE )  ;
                   'IF'
                   TSTBL ( BUFFER 'OF' ( INPUT 'OF' CITY ) )
                   'THEN'
                   COUNT (   TRAFBLK 'OF' ( MEASUREMENT 'OF' CITY  )  )  ;
                   COUNT ( BLK 'OF' ( INPUT 'OF' CITY ) )
                   'ELSE'
                   PACKET := PKGEN ( NAME 'OF' CITY , N , CLOCK )  ;
                   PKTIM ( PACKET , CLOCK )  ;
                   PKIPG ( PACKET , BUFFER 'OF' ( INPUT 'OF' CITY ) ,
                                    MAX    'OF' ( INPUT 'OF' CITY ) )  ;
                   COUNT (   TRAFSUC 'OF' ( MEASUREMENT 'OF' CITY  )  )  ;
                   COUNT ( SUC 'OF' ( INPUT 'OF' CITY ) )
                   'FI'
              'ELSE'
              COUNT ( IDL 'OF' ( INPUT 'OF' CITY ) )
```

```
              'FI'
              )  ;


        'PROC' TRMNT = ( 'REF''CITY' CITY , 'INT' CLOCK ) 'VOID':
              (
              'BOOL' FOUND := 'FALSE'  ;
              'PACKET' PACKET  ;
              'FOR' I 'TO' 'UPB' ( BUFFER 'OF' (INPUT 'OF' CITY) )
              'WHILE' TSTRP ( (BUFFER 'OF' (INPUT 'OF' CITY) )[I] , CLOCK )
              'AND'   FOUND = 'FALSE'
              'DO'
              'IF'
              CH2 'OF' ( BUFFER 'OF' (INPUT 'OF' CITY) )[I] =
              NAME 'OF' CITY
              'THEN'
              PACKET := PKFPG ( (BUFFER 'OF' (INPUT 'OF' CITY))[I] )  ;
              PKTIM ( PACKET , CLOCK )  ;
              COUNT ( TRAFARV 'OF' (MEASUREMENT 'OF' CITY) )  ;
              ( PATH 'OF' (MEASUREMENT'OF'CITY))'PLUS' COUNTER 'OF' PACKET  ;
              (DELAY 'OF' (MEASUREMENT'OF'CITY))'PLUS' PKDLY ( PACKET )  ;
              SHIFT ( BUFFER 'OF' (INPUT 'OF' CITY ) )  ;
              FOUND := 'TRUE'
              'ELSE'
              'SKIP'
              'FI'
              )  ;



        'C'      PART  : : NINE                                     'C'

        'C'      NETWORK  FUNCTION  PROCEDURE                       'C'


        'PROC' NETFN = ( 'REF''NETWORK' NETWORK , 'INT' CLOCK ,
                         'REAL' RATE ) 'VOID':
              (
              'FOR' I 'TO' 'UPB' ( CITY 'OF' NETWORK )
              'DO' (
              TRFIN ( ( CITY 'OF' NETWORK )[I] ,
                      'UPB' ( CITY 'OF' NETWORK ) ,
                      CLOCK , RATE )  ;
              SQROT ( ( CITY 'OF' NETWORK )[I] , CLOCK )  ;
              FRWRD ( ( CITY 'OF' NETWORK )[I] ,
                      NETWORK , CLOCK )  ;
              TRMNT ( ( CITY 'OF' NETWORK )[I] , CLOCK )
                  )
              )  ;



        'C'      PART  : : TEN                                      'C'

        'C'      THE  BODY  OF  THE  SIMULATION  PROGRAM            'C'


        'NETWORK' NETWORK  ;

        INNET ( NETWORK , N,RATE )  ;

        'FOR' I 'TO' RUN 'DO'
        (
        NETFN ( NETWORK , I , RATE )  ;
        'IF'
        I = 50  'OR'
```

```
            I = 100 'OR'
            I = 150 'OR'
            I = 200 'OR'
            I = 250 'OR'
            I = 300 'OR'
            I = 350 'OR'
            I = 400 'OR'
            I = 500 'OR'
            I = 600 'OR'
            I = 700 'OR'
            I = 800 'OR'
            I = 900 'OR'
            I = 1000
            'THEN'
            RPNET ( NETWORK , N , ISZ , OSZ , I , RATE )
            'FI'
            )

    'END'
    'FINISH'
    ****

2.57+ ****
2.57 0.03 FINISHED : 1 LISTFILES
2.57 JOBTIME USED 3 ; MAXIMUM CORE USED 0
2.57 JOB UNITS 20

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXX                                  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXX         NUMBER OF PAGES   12     XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXX                                  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

REFERENCES

REFERENCES

1. W.C. Jakes, *Microwave Mobile Communications*, Chapter 7, Wiley, 1974.

2. J. Martin, *Future Development in Telecommunication*, Chapter 12, Prentice-Hall, 1977.

3. News of the months, "Cellular mobile radio going ahead", Wireless World, p.40, June 1977.

4. News of the month, "MRUA call for mobile radio 'Annan'", Wireless World, p.49, July 1977.

5. R. Gundlach, "A Tand T starts tests of cellular mobile telephone system", Electronics, Vol.50, pp. 75-76, June 9, 1977.

6. N. Ehrlich, "The advanced mobile phone service", IEEE Communication Magazine, Vol. 17, No. 2, pp. 9-15, March 1977.

7. W.M. Pannel, "Spectrum usage in the Middle East", Communication International, Vol. 6, No. 6, pp. 30-42, June 1979.

8. W.R. Young, "Advanced mobile phone service: Introduction, background, and objectives", The Bell System Technical Journal, Vol. 58, No. 1, pp. 1-14, January 1979.

9. V.H. Macdonald, "Advanced mobile phone service: The cellular concept", The Bell System Technical Journal, Vol. 58, No. 1, pp.15-41, January 1979.

10. R.H. Frenkiel, "A high-capacity mobile radio telephone system model using a coordinated small-zone approach", IEEE Transactions on Vehicular Technology, Vol. VT-19, No.2, pp. 173-177, May, 1970.

11. A.C. Stocker, "Application of the small-zone concept to police communications in population centres: an exploratory study", IEEE Transactions on Vehicular Technology, Vol. VT-25, No.2, pp. 50-54, May 1976.

12. T. Nomura and N. Yoskikawa, "Multiple radio zone plans in mobile radio systems", IEEE Transactions on Vehicular Technology, Vol. VT-25, No.3, pp. 68-74, August 1976.

13. G.D. Ott, "Vehicle location in cellular mobile radio systems", IEEE Transactions on Vehicular Technology, Vol.VT-26, No. 1, pp. 43-46, February 1977.

14. L. Schiff, "Traffic capacity for three types of common-user mobile radio communication systems", IEEE Transaction on Communication Technology, Vol. Com-18, No. 1, pp.12-21, February 1970.

15. D.C. Cox and D.O. Reudink, "Dynamic channel assignment in high-capacity mobile communications systems", The Bell System Technical Journal, Vol. 50, No. 6, pp.1833-1857, August 1971.

16. D.C. Cox and D.O. Reudink, "A comparison of some channel assignment strategies in large-scale mobile communication systems", IEEE Transactions on Communications, Vol. Com-20, No.2, pp. 190-195, April 1972.

17. D.C. Cox and D.O. Reudink, "The behaviour of dynamic channel-assignment mobile communications ystems as a function of numbers of radio channels", IEEE Transactions on Communications, Vol. Com-20, No.3, pp.471-479, June 1972.

18. D.C. Cox and D.O. Reudink, "Dynamic channel assignment in two-dimensional large-scale mobile radio systems", The Bell System Technical Journal, Vol.51, No.7, pp. 1611-1630, September 1972.

19. D.C. Cox and D.O. Reudink, "Increasing channel occupancy in large-scale mobile radio systems: dynamic channel reassignment", IEEE Transactions on communications, Vol. Com-21, No. 11, pp.1302-1306, November 1973.

20. J.S. Engel, "Statistically-optimum dynamic server assignment in systems with interfering servers", IEEE Transactions on Communications, Vol. Com-21, No.11, pp.1287-1293, November 1973.

21. L.G. Anderson, "A simulation study of some dynamic channel assignment algorithms in a high capacity mobile telecommunications system", IEEE Transactions on Communications, Vol. Com-21, No. 11, pp. 1294-1301, November 1973.

22. T.J. Kahwa and N.D. Georganas, "A hybrid channel assignment scheme in large-scale, cellular-structured mobile communication systems", IEEE Transactions on Communications, Vol. Com126, No.4, pp. 432-438, April 1978.

23. H. Staras and L. Schiff, "A dynamic space division multiplex mobile radio system", IEEE Transactions on Vehicular Technology, Vol. VT.19, No.2, pp.206-213, May 1970.

24. P.T. Porter, "Supervision and control features of small-zone radio telephone system", IEEE Transactions on Vehicular Technology, Vol. VT-20, No.3, pp.75-79, August 1971.

25. Z.C. Fluhr and E. Nussbaum, "Switching plan for a cellular mobile telephone system", IEEE Transactions on Communications, Vol. Com-21, No.11, pp.1281-1286, November 1973.

26. V. Hackenburg, B.D. Holm and J.I. Smith, "Data signalling functions for a cellular mobile telephone system". IEEE Transactions on Vehicular Technology, Vol. VT-26, No.1, pp. 82-87, February 1977.

27. K.J.S. Chadha, C.F. Hunnicutt, S.R. Peck and J. Tebes, "Advanced mobile phone service: mobile telephone switching office", The Bell System Technical Journal, Vol. 58, No.1, pp.71-95, January 1979.

28. Z.C. Fluhr and P.T. Porter, "Advanced mobile phone service: control architecture", The Bell System Technical Journal, Vol.58, No.1, pp.43-69, January 1979.

29. S.H. Bakry and M.H. Ackroyd, "Computer control for cellular mobile radio", IEE Colloquium on the Potential of Digital Processors in Radio-communication, Digest No. 1978/6, pp.1-5, January 1978.

30. D.W. Davies and D.L. Barker, *Communication Networks for Computers*, Wiley, 1973.

31. N. Abramson and F. Kuo, *Computer Communication Networks*, Prentice-Hall, 1973.

32. J. Martin, *Teleprocessing Network Organization*, Prentic Hall, 1970.

33. "Communication premier", IBM student text, 1970.

34. L.G. Roberts and B.D. Wessler, "Computer network development to achieve resource sharing", AFIPS Conference proceedings, Vol.36, pp.543-549, May 1970.

35. I.T. Frish and H. Frank, "Computer communications. How we got where we are", AFIPS Conference proceedings, Vol.44, pp.109-117, May 1975.

36. W. Chow, "Computer communication networks. The parts make up the whole", AFIPS Conference proceedings, Vol.44, pp.119-128, May 1975.

37. P.T. Kirstein, "Data communication by packet switching", Electronics and Power, November 15, 1973.

38. W.R. Crowther, F.E. Heart, A.A. McKenzie, J.M. Mc Quillan and D.C. Walden, "Issues in packet switching network design", AFIPS Conference proceedings, vol.44, pp.161-175, May 1975.

39. J.C. Majithia and R. Bhar, "Optimum packet size in a packet switching data network", Electronics Letters, Vol. 10, No.10, pp.191-193, May 1974.

40. J.C. Majithia and R. Bhar, "Analysis of overheads in packet-switched data networks", Proceedings of the IEE, Vol. 121, No.11, pp.1375-1376, November 1974.

41. G.M. Schneider, "Computer network protocols: a hierarchical viewpoint", Computer, IEEE Computer Society, Vol. 12, No.9, pp.8-10, September 1979.

42. M.Gerla, "The Design of Store and Forward Networks for Computer Communications", a Ph.D. thesis, School of Engineering and Applied Science, University of California, Los Angeles, 1973.

43. S.M. Ornstein, F.E. Heart, W.R. Crowther, H.R. Rising, S.B. Russell and A. Michel, "The terminal IMP for the ARPA computer network", AFIPS Conference proceedings, Vol. 40, pp.243-254, May 1972.

44. R. Bhar and J.C. Majithia, "A study of routing strategies in a packet switching computer network", CCNG report E-20, University of Waterloo, May 1974.

45. R.L. Grimsdale, B.H. Pardoe, M.C. Daniels and R. Mc Nulty, "Investigation of computer-based data communication network", A report of work supported by the Science Research Council, School of Applied Sciences, The University of Sussex, October 1976.

46. N. Abramson, "The Aloha system -another alternative for computer communications", AFIPS Conference proceedings, Vol. 37, pp.281-285, November 1970.

47. L. Kleinrock and S.S. Lam, "Packet switching in a slotted satellite channel", AFIPS Conference proceedings, Vol-42, pp.703-710, June 1973.

48. L. Kleinrock and F.A. Tobagi, "Packet switching in radio channels: Part I - Carrier sense multiple-access modes and their throughput delay characteristics", IEEE Transactions on Communications, Vol. Com-23, No.12, pp.1400-1416, December 1975.

49. F.A. Tobagi and L. Kleinrock, "Packet switching in radio channels : Part II - the hidden terminal problem in carrier sense multiple-access and the busy tone solution", IEEE Transactions on Communications, Vol. Com-23, No.2, pp.1417-1433, December 1975.

50. L. Kleinrock and S.S. Lam, "Packet switching in a multiaccess broadcast channel : performance evaluation", IEEE Transactions on Communications, Vol. Com-23, No.4, pp.410-423, April 1975.

51. L. Kleinrock and S.S. Lam, "Packet switching in a multiaccess broadcast channel: dynamic control procedures", IEEE Transactions on Communication, Vol. Com-23, No.9, pp.891-904, September 1975.

52. C.L. Heitmeyer, J.H. Kullback and J.E. Shore, "A survey of packet switching techniques for broadcast media", Naval Research Laboratory, Report 8035, Washington D.C., October 1976.

53. N. Abramson, "The throughput of packet broadcasting channels", IEEE Transactions on communications, Vol. Com-25, No.1 pp.117-128, January 1977.

54. P. Beckmann, *Probability in Communication Engineering*, Harcourt Brace and World, 1967.

55. P. Beckmann, *Introduction to Elementary Queuing Theory and Telephone Traffic*, The Golem Press, 1968.

56. D. Bear, *Principles of Telecommunication -Traffic Engineering*, Peter Peregrinus, 1976.

57. J.E. Flood, *Telecommunication Networks*, Peter Peregrinus, 1975.

58. L. Kleinrock, *Queueing Systems, Volume 1: Theory*, Wiley, 1975.

59. L. Kleinrock, *Queueing Systems, Volume 2: Computer Applications*, Wiley, 1976.

60. S.H. Bakry and M.H. Ackroyd, "Teletraffic analysis for single-cell mobile radio telephone systems", unpublished report, Department of Electrical and Electronic Engineering, University of Aston in Birmingham, England, December 1979.

61. S.H. Bakry and M.H. Ackroyd, "Teletraffic analysis for multi-cell mobile radio telephone systems", unpublished report, Department of Electrical and Electronic Engineering, University of Aston in Birmingham, England, March 1980.

62. S.H. Bakry and M.H. Ackroyd, "Teletraffic analysis and simulation for large-scale mobile radio systems". IERE International Conference on Land Mobile Radio, pp. 401-405, September 1979.

63. G. Gordon, *System Simulation*, Prentic Hall, 1978.

64. R.E. Shannon, *System Simulations. The Art and Science*, Prentice-Hall, 1975.

65. G.W. Evans, G.F. Wallace and G.L. Sutherland, *Simulation Using Digital Computers*, Prentice-Hall, 1967.

66. D.C.S. Shearn, "Discrete event simulation in Algol 68", Software Practice and Experience, Vol.5, pp.279-293,1975.

67. E.K. Bowdon, S.A. Marmrak and F.R. Salz, "Simulation - a tool for performance evaluation in network computers", AFIPS Conference proceedings, Vol.42, pp.121-131, June 1973.

68. Z. Ziemski, "A simulation study of a localized computer network", Ph.D. thesis, Computer Centre, University of Aston, 1977.

69. M.H. Ackroyd, "Increasing the efficiency of the roulette simulation of teletraffic", Electronic Letters, Vol.14, No.9, p.270, 1978.

70. S.H. Bakry and M.H. Ackroyd, "Little v. large: mobile radio system simulation on PDP.11/03 and 1904s", DECUS UK Conference, pp.29-33, April 1979.

71. S.H. Bakry and M.H. Ackroyd, "Structured programming for the simulation of data communication systems", Fifth National Computer Conference, Saudi Arabia, pp. 285-302, March 1979.

72. J.K. Hughes and J.I. Michtom, *A Structured Approach to Programming*, Prentice-Hall, 1977.

73. F.T. Baker, "Structured programming, special tutorial", IMB (UK), March 1975.

74. O.J. Dahl, E.W. Dijkstra and C.A.R. Hoare, *Structured Programming*, Academic Press, 1972.

75. E.F. Miller and G.E. Lindamood, "Structured programming: top-down approach", Datamation, Vol.19, pp.55-57, December, 1973.

76. J.R. Donaidson, "Structured programming", Datamation, Vol. 19, pp.52-54, December 1973.

77. J.D. Halliwell and T.A. Edwards, *A Course in Standard Coral 66*, National Computing Centre Publications, 1977.

78. J.T. Webb, *Coral 66 Programming*, National Computing Centre Publications, 1978.

79. P.M. Woodward, P.R. Wetherall and B. Gorman, *Official Definition of Coral 66*, Her Majesty's Stationery Office, 1978.

80. R. Firth and P.B. de Bono, "Programmer's guide for the RMCs Coral 66 compiler for the PDP-11", HPAC Limited.

81. C.H. Lindsey and S.G. van der Meulen, *Informal Introduction to Algol 68*, North-Holland Publishing Company, 1971.

82. A.Learner and A.J. Powell, *An Introduction to Algol 68 Through Problems*, Mac Millan, 1974.

83. P.M. Woodward and S.G. Bond, *Algol 68-R Users Guide*, Her Majesty's Stationery Office, 1975.

84. Royal Radar Establishment Algol 68-R System Installation and Maintenance, 1975.

85. The University of Aston Algol 68 Users Handbook, 1978.

86. J.S. Bendat and A.G. Piersal, *Measurement and Analysis of Random Data*, Chapter 4, Wiley, 1966.

87. D.E. Knuth, *The Art of Computer Programming:* Volume 2, Chapter 3, Addison-Wesley Publishing Company, 1969.

88. R.P. Parmele, T.I. Peterson, C.C. Tillman and D.J. Hatfield, "Virtual storage and virtual machine concepts", IBM Systems Journal, No. 2, 1972.

89. J.E. Brand and J.C. Warner, "Processor call carrying capacity estimation for stored program control switching systems", Proceedings of the IEEE, Vol. 65, No.9, pp.1342-1349, September 1977.

90. J.W. Forgie, "Speech transmission in packet-switched store-and-forward networks", AFIPS Conference proceedings, Vol.44, pp.137-142, May 1975.

91. B. Gold, "Digital speech networks", Proceedings of the IEEE, Vol.65, No.2, pp.1636-1658, December 1977.

92. D. Minoli, "Optimal packet length for packet voice communication", IEEE Transactions on Communications, Vol. Com-27, No.3, pp.607-611, March 1979.

93. G. Barberis and D. Pazzaglia, "Analysis and optimal design of a packet-voice receiver", IEEE Transactions on Communications, Vol. Com-28, No.2, pp.217-227, February 1980.