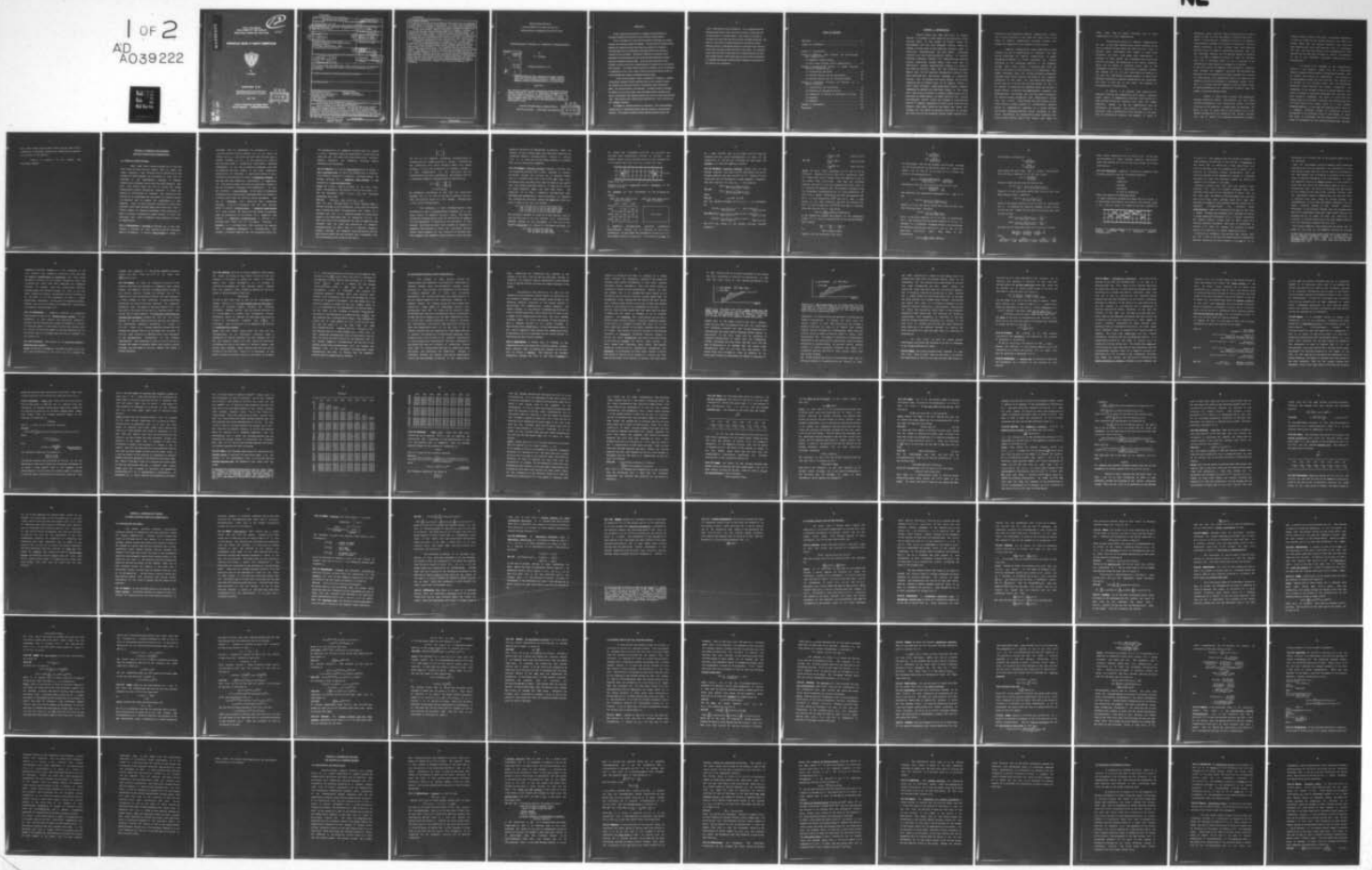
AD-A039 222 HARVARD UNIV CAMBRIDGE MASS DIV OF ENGINEERING AND --ETC F/G 17/2
DECENTRALIZED CONTROL IN COMPUTER COMMUNICATION.(U)
APR 77 F C SCHOUTE
UNCLASSIFIED TR-667 N00014-75-C-0648
NL

1 OF 2
AD A039222



AD A039222

Office of Naval Research
Contract N00014-75-C-0648 NR-372-012
National Science Foundation Grant ENG-76-11824

DECENTRALIZED CONTROL IN COMPUTER COMMUNICATION



By

F. C. Schoute

Technical Report No. 867

This document has been approved for public release and sale; its distribution is unlimited. Reproduction in whole or in part is permitted by the U. S. Government.

DDC
RECEIVED
MAY 10 1977
D

April, 1977

Division of Engineering and Applied Physics
Harvard University • Cambridge, Massachusetts

ACCESSION NO.
DDC FILE COPY

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER Technical Report No. 667	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) DECENTRALIZED CONTROL IN COMPUTER COMMUNICATION.		5. TYPE OF REPORT & PERIOD COVERED Interim Report. 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) F. C. Schoute		8. CONTRACT OR GRANT NUMBER(s) N00014-75-C-0648, NSF-ENG-76-11824
9. PERFORMING ORGANIZATION NAME AND ADDRESS Division of Engineering and Applied Physics Harvard University Cambridge, Massachusetts 02138		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS		12. REPORT DATE April 1977 13. NUMBER OF PAGES 136
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) TR-667		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

This document has been approved for public release and sale; its distribution is unlimited. Reproduction in whole or in part is permitted by the U.S. Government.

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

18. SUPPLEMENTARY NOTES

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

Decentralized Control
Non-Classical Control Theory
Computer Communication
Packet Switching
Team Theory
Computer Networks
Satellite Communication

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

In this report three problems in computer communications are considered within the framework of non-classical control theory.

First, in Chapter 2 we deal with the problem of sharing one communication wire among a number of stations. The fact that all communication stations are identical and that they share one objective of using the communication wire as efficiently as possible leads to the concept of symmetric team problems. Symmetric solutions to symmetric team problems are characterized by the restriction that all decision makers must have identical decision rules.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

163750

Unclassified
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20. Abstract continued

In the second section the access problem in multi-access wire communication is considered as a symmetric team problem. It is shown that the symmetric solution, which corresponds to randomized access rules, tends to give as good performance as the unrestricted when the number of stations becomes large.

In the second problem, which is considered in Chapter 3, stations communicate through a packet switched multi-access satellite channel. The stations can only share (control-) information with considerable delay: the round trip time to the satellite. A simple model is developed for which it can be shown under some assumptions that, because of the delay in information, the optimal decision rule can be an open-loop decision rule. This decision rule is determined separately for "new" packets and for "collided" packets.

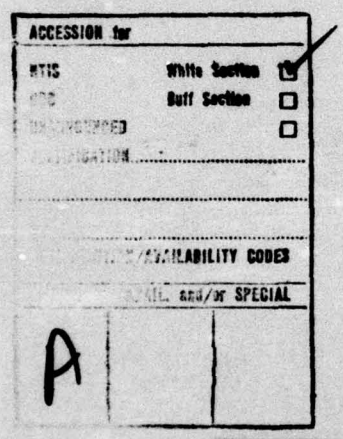
In Chapter 4, the last problem is considered. The communication medium here is a distributed packet switched computer communication network. The problem considered deals with the question what is the "best" information to base decisions on. It is recognized that the statistical parameters which describe the system, themselves are varying in time in a random fashion. This leads to a cascade of stochastic processes that describe the most essential parameter: the delay or travel time of a packet going from one node to another. Different classes of information policies correspond to the different levels of the cascade at which the delay is described. It is analyzed what is the best class of information for routing decisions and also other design choices, which affect how and what control information is exchanged between the communication computers at the nodes of the network, are considered.

Office of Naval Research

Contract N00014-75-C-0648 NR-372-012

National Science Foundation Grant ENG 76-11824

DECENTRALIZED CONTROL IN COMPUTER COMMUNICATION



By

F. C. Schoute

Technical Report No. 667

This document has been approved for public release and sale; its distribution is unlimited. Reproduction in whole or in part is permitted by the U. S. Government.

April 1977

The research reported in this document was made possible through support extended the Division of Engineering and Applied Physics, Harvard University by the U. S. Army Research Office, the U. S. Air Force Office of Scientific Research and the U. S. Office of Naval Research under the Joint Services Electronics Program by Contract N00014-75-C-0648 and by the National Science Foundation Grant ENG 76-11824.

Division of Engineering and Applied Physics

Harvard University · Cambridge, Massachusetts

DDC
RECEIVED
MAY 10 1977
D

ABSTRACT

In this report three problems in computer communication are considered within the framework of non-classical control theory.

First, in Chapter 2 we deal with the problem of sharing one communication wire among a number of stations. The fact that all communication stations are identical and that they share one objective of using the communication wire as efficiently as possible leads to the concept of symmetric team problems. Symmetric solutions to symmetric team problems are characterized by the restriction that all decision makers must have identical decision rules. In the second section the access problem in multi-access wire communication is considered as a symmetric team problem. It is shown that the symmetric solution, which corresponds to randomized access rules, tends to give as good performance as the unrestricted when the number of stations becomes large.

In the second problem, which is considered in Chapter 3, stations communicate through a packet switched multi-access satellite channel. The stations can only share (control-) information with considerable delay: the round trip time to the satellite. A simple model is developed for which it can be shown under some assumptions that, because of the delay in information, the optimal decision rule can be an open-loop decision rule. This decision rule is determined separately for "new" packets and for "collided" packets.

In Chapter 4, the last problem is considered. The communication medium here is a distributed packet switched computer communication network. The problem considered deals with the question what is the

"best" information to base decisions on. It is recognized that the statistical parameters which describe the system, themselves are varying in time in a random fashion. This leads to a cascade of stochastic processes that describe the most essential parameter: the delay or travel time of a packet going from one node to another. Different classes of information policies correspond to the different levels of the cascade at which the delay is described. It is analyzed what is the best class of information for routing decisions and also other design choices, which affect how and what control information is exchanged between the communication computers at the nodes of the network, are considered.

TABLE OF CONTENTS

ABSTRACT..... iii

TABLE OF CONTENTS..... v

Chapter 1: Introduction 1

Chapter 2: Symmetric Team Problems and Multi-Access Wire Communication

2:1 Symmetric Team Problems 7

2:2 The Access Problem in Wire Communication . . . 24

Chapter 3: Decentralized Control in Packet Switched Satellite Communication

3:1 Introduction and Model 49

3:2 Optimal Control Law for New Packets 56

3:3 Optimal Control Law for Collided Packets . . . 68

Chapter 4: Information Policies for Routing in a Computer Network

4:1 Introduction and Definitions 79

4:2 Choosing an Information Policy 87

4:3 Comparison of Classes of Information Policies . 102

4:A Appendix 115

4:B Appendix 116

Chapter 5: Conclusions 124

References 128

CHAPTER 1: INTRODUCTION.

Control theory has been and still is largely devoted to problems in which there is one controller, alias decision maker. In this aspect of control theory the developments are now very advanced. Control theory for systems in which there is more than one decision maker is relatively underdeveloped, but this aspect --sometimes referred to as non-classical control theory-- is recently enjoying increased interest [1]. Many practical problems involve large-scale systems which --one could say: by definition-- require decentralized control, i.e. there is more than one decision maker and the different decision makers have not all identical information on which to base their decisions [2]. The control problems in large-scale systems, such as economic and social systems, are of great complexity and at present there is no general theory to solve those problems. Among large-scale systems, computer communication systems are relatively simple in the sense that many control problems in computer communication can be described by simple models. It seems therefore that joining non-classical control theory and problems in decentralized control of computer communication forms a good basis for new research. This report builds on this basis and shows that joining the two fields is mutually beneficial: in one way, the frame work of non-classical control theory enables us to

formulate a few problems in computer communication clearly and to find solutions, in the other way, there arise in the study of computer communication problems concepts that extend the framework of non-classical control theory.

Computer communication means communication among computers as well as communication by computers. The most suited technology for computer communication is packet switching as opposed to circuit switching [3]. Circuit switching is the technology employed in the telephone system; here, there exists a dedicated path from end point to end point for the duration of the connection. With packet switching there are strings of binary information (packets) which travel from one end point of the (logical) connection to the other, but the actual path taken by a packet need not to be the same for each packet. In fact it is decided by communication computers which way and when a packet travels. In general the communication computers are geographically distributed and can only share information for decision making through the same communication medium that they control. The problems that are treated in this report pertain to this process of decentralized decision making by communication computers in a packet switched environment. Although presently packet switching is used almost exclusively for communication among computers, this aspect is not crucial; most of the results also apply to

other other uses of packet switching such as telex communication or voice communication.

Decentralized control of computer communication, as well as any other decentralized control problem, is not only a question of what should the controls be in order to achieve good performance with respect to some given criterion, but also --and maybe most importantly-- a question of what should the information be on which the controls are based. The importance of the latter question, which can be stated in different words: what is the information structure of a multi-person decision problem, was pointed out in [4] and later made more explicit in [5] and [6]. The three following chapters, in which three different problems in computer communication are considered, all have, in one way or the other, the question of information as a central point.

In chapter 2 we consider some aspects of the problem of sharing one communication wire among a (possibly large) number of communication stations. This problem is a team problem because the stations also share one common objective, namely, to use the communication wire as efficiently as possible. The problem is symmetric, or invariant under a permutation of the stations, in the sense that all stations are identical. For example, it makes no

difference which stations have at a given point in time a packet to send, but only the number of stations that have a packet to send matters. A symmetric solution to a symmetric team problem is defined as a solution in which all decision makers (stations, in this context) must have identical decision rules. In terms of information structure this has the following meaning. Let the decision makers be numbered 1,2,...,N. Then a symmetric solution is just one decision rule (for N stations) which is not a function of the information of what number is assigned to a station.
In the first section of chapter 2 the concepts of symmetric team problems and symmetric solutions are developed and motivated. In the second section these concepts are applied to multi-access wire communication and it is shown how the symmetric solution corresponds to randomized access rules. Also it is shown that the symmetric solution tends to give as good performance as the unrestricted solution when the number of stations becomes large.

Chapter 3 deals with stations that communicate through a satellite channel. The stations can only share (control-)information with considerable delay: the round trip time to the satellite. A simple model is developed for which it can be shown that a) the private (i.e. not yet shared) information is not needed for the optimal decision rule, and b) given a few reasonable assumptions, beyond a

certain relative value of the delay, the delayed information is also not needed for the optimal decision rule. These two points imply that the optimal decision rule can be an open-loop decision rule. This decision rule is determined for both the case that packets to be transmitted are "new" packets and the case that, because of earlier interference of two or more stations, "collided" packets need to be retransmitted.

In chapter 4 the topology of the communication medium is very general, namely that of a distributed computer communications network, through which packets must be routed from source to destination. The analysis is correspondingly broader and less detailed. The perspective is broader in the specific sense that the statistical parameters which describe the system, themselves are recognized to be varying in time in a random fashion. This leads to cascade of stochastic processes that describe the most essential parameter: the delay or travel time of a packet going from one node to another. The cascade goes from the actual value of the delay, which is a very rapidly changing quantity, to the long term expected delay, which is a constant. Different classes of information policies correspond to the different levels of the cascade at which the delay is described. We will analyse what is the best class of information for the routing decisions and consider

also other design choices which affect how and what control information is exchanged between the communication computers at the nodes of the network.

Finally, in chapter 5 we will present some concluding remarks.

CHAPTER 2: SYMMETRIC TEAM PROBLEMS AND MULTI-ACCESS WIRE COMMUNICATION.

2:1 Symmetric Team Problems.

Many large scale system problems are of the kind where there are many decision makers that all share one common objective. Such problems fall by definition in the category of team problems. As an instance of this kind of problems we are specifically thinking of computer communication problems, where there are many computers that must each decide when and how to access the common communication medium, sharing the objective of using the medium as efficiently as possible. The purpose of this section is to investigate the case where all decision makers are identical and to examine the consequences of this symmetry. Here we shall restrict ourselves to finite team problems, where each decision maker can choose only from a finite number of possible strategies. This restriction is made to provide a conceptually simple context, in which the phenomena that arise in symmetric team problems can easily be demonstrated.

(2:1.1) Definitions. A strategy is defined as a map that assigns a decision to each possible value of a decision maker's information. A (finite) team problem in normal or

strategic form is represented by a N-tensor $A(\cdot,\cdot,\ldots,\cdot)$ --in the case N=2 this is a matrix $A(\cdot,\cdot)$-- of which the element $A(j_1,j_2,\ldots,j_N)$ gives the cost when decision maker i chooses strategy j_i, i=1,...,N. The problem is to find for each decision maker a probability distribution (mutually independent) on his set of possible strategies, such that the expected cost with respect to the joint probability distribution is minimized. A particular choice of such a probability distribution is called a randomized strategy. In the case that the probability distribution is degenerate, i.e. the decision maker will pick one particular element of his set of possible strategies with probability 1, we call the strategy a pure strategy. So the space of randomized strategies also contains pure strategies which correspond in a trivial way to what we initially called strategies. From now on 'strategy' will implicitly mean 'randomized strategy'. A set of N strategies (one for each decision maker) is called a strategy-tuple. A symmetric team problem is a team problem for which the cost tensor is invariant under any permutation of its indices. For a symmetric team problem a symmetric strategy-tuple is defined as a strategy-tuple which has all strategies identical. A solution is a strategy-tuple that minimizes the expected cost. A symmetric solution,is a strategy-tuple that uniquely minimizes expected cost. The uniqueness means that

the implementation of a symmetric solution does not require a priori agreement among the team members on more than the model and cost. Note that with these definitions 'solution' implies optimality and 'symmetric solution' implies optimality and uniqueness.

(2:1.2) Theorem. For a given team problem there is always a pure strategy-tuple for which the minimum cost is achieved. If we restrict ourselves to symmetric strategy-tuples, the optimal strategy for a symmetric team problem is not necessarily a pure strategy-tuple.

Proof. Let $(j_1, j_2, \ldots, j_N)$ be an index of the cost tensor which corresponds to the smallest value of the tensor, i.e. for any meaningful index $(j_1, j_2, \ldots, j_N)$ the following inequality must hold:

(2:1.3) $$A(j_1, j_2, \ldots, j_N) \leq A(j_1, j_2, \ldots, j_N)$$

Then the pure strategy-tuple in which decision maker i chooses strategy j_i with probability 1 will give the minimum cost, because any other strategy-tuple will result in a minimum cost that is a weighted average of values which appear in the right-hand side of inequality (2:1.3). To show that this is not necessarily the case in symmetric team problems when we have the restriction of symmetric strategy-tuples, we shall look at a specific (counter) example. Consider the symmetric team problem with N=2 and each decision maker has two different pure strategies with the corresponding cost given by the matrix

$$\begin{bmatrix} 1 & 0 \\ 0 & a \end{bmatrix}$$

The set of all symmetric randomized strategy-tuples is parametrized by a single parameter u ($0 \leq u \leq 1$), representing the probability with which decision maker i will choose strategy 1 (i=1,2). The probability of choosing strategy 2 is then automatically 1-u. By definition the two randomizations are done independently. The expected cost for a randomized symmetric strategy-tuple is then given by

$$\begin{bmatrix} u & 1-u \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & a \end{bmatrix} \begin{bmatrix} u \\ 1-u \end{bmatrix}$$

$$= (1+a)u^2 - 2au + a.$$

By elementary calculation we see that the constrained minimum will be at u=0 for $a \leq 0$ and at u=a/(a+1) for a>0. Thus, for positive values of a, the optimal strategy-tuple is not a pure strategy-tuple.

The optimality of pure strategy-tuples in team problems is in fact a straightforward extension of a well known result in Bayesian decision theory. The need for randomized strategy-tuples under the restriction of symmetry is a new phenomenon. To demonstrate the usefulness of this imposed restriction we shall first consider an example of a symmetric team problem for which the restricted solution models some dynamics we may encounter in everyday live. This suggests that symmetric team problems do exist and that

symmetric solutions are implemented in practice. After the example we shall discuss some other desirable properties of randomized symmetric strategy-tuples. Finally in section (2:2) it is shown how one may impose symmetry in problems that are not necessarily symmetric.

(2:1.4) Example. (Corridor Problem) Consider a corridor that is conceptually divided into 3 lanes. Walking in opposite directions through the corridor are 2 decision makers (d.m.'s). Initially both d.m.'s are in the same lane, say, the middle lane, and they still have t=T steps to go, before they will either pass each other or, if they can not decide on using different lanes, run into each other. The objective is to find a symmetric strategy-tuple that minimizes the probability of such a collision. To describe the problem more precisely, define the state when there are still t steps to go, x(t), as follows

x(t)="n" if both d.m.'s are in the north lane
="m" if both d.m.'s are in the middle lane
="s" if both d.m.'s are in the south lane
="o" if the d.m.'s are in different lanes

The horizontal movement is fixed to be one step per unit of time for both d.m.'s, but the vertical (i.e. north-south) movement depends on the strategies of the d.m.'s. The possible decisions (i.e. values of a strategy) are given by

$u_i(t)$="u" to move one lane up
="h" to stay in same lane
="d" to move one lane down
i=1,2

Of course the strategies $u_i(t)$="u" or $u_i(t)$="d" are excluded when respectively x(t)="n" or x(t)="s". The problem ceases to be a problem as soon as x(t) becomes "o"; it is assumed that then both d.m.'s keep their lane.

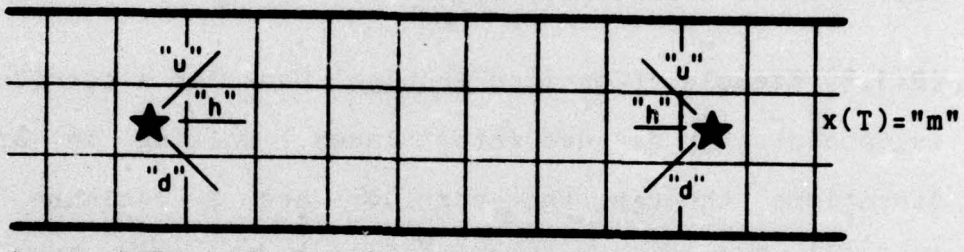


Figure 2:1.a. Initial <u>state and</u> possible <u>decisions</u> in the Corridor Problem.

The <u>dynamics</u> are then represented by the following two tables

table for next state x(t-1) when $u_1(t)=u_2(t)$

x(t)	u="u"	u="h"	u="d"
="n"		"n"	"m"
="m"	"n"	"m"	"s"
="s"	"m"	"s"	

table for next state x(t-1) when $u_1(t)\neq u_2(t)$

x(t-1)="o"

A symmetric strategy-tuple specifies probability distributions $(\tilde{u},\tilde{h},\tilde{d})$ as a function of the d.m.s' information. In this model the information of both d.m.'s is the present values of t and x(t). If we define the <u>cost</u> to

be 1 when x(0)≠"o" and to be 0 when x(0)="o", then the expected cost for a given strategy-tuple is equal to the probability of the two d.m.'s running into each other. The problem is to find the symmetric solution.

(2:1.5) Solution. (Corridor Problem) Define M(t) as the minimum expected cost when x(t)="m" and define S(t) as the minimum expected cost when x(t)="n" or "s" (clearly in both states the cost is the same by a simple symmetry argument). then M(t) and S(t), for t=0,1,2,... , are given by

$$M(t)=\frac{2}{\lambda_1^{t-1}(1+2\lambda_1)+\lambda_2^{t-1}(1+2\lambda_2)}$$

(2:1.6)

$$S(t)=\frac{4}{\lambda_1^{t-1}(1+3\lambda_1)+\lambda_2^{t-1}(1+3\lambda_2)}$$

where

(2:1.7) $$\lambda_1=1-\sqrt{2},\quad \lambda_2=1+\sqrt{2}$$

and the optimal strategy, for t=1,2,3,... , is represented by

$$(\tilde{u},\tilde{h},\tilde{d})=\frac{1}{M(t-1)+S(t-1)}(\ 0\ ,M(t-1),S(t-1)) \quad \text{when } x(t)="n"$$

(2:1.8)

$$(\tilde{u},\tilde{h},\tilde{d})=\frac{1}{2M(t-1)+S(t-1)}(M(t-1),S(t-1),M(t-1)) \quad \text{when } x(t)="m"$$

$$(\tilde{u},\tilde{h},\tilde{d})=\frac{1}{M(t-1)+S(t-1)}(S(t-1),M(t-1),\ 0\ ) \quad \text{when } x(t)="s"$$

For $t\to\infty$ the values of the optimal strategy converge rapidly to

$$\frac{1}{1+\sqrt{2}}(0,1,\sqrt{2}) \qquad \text{when } x(t)=\text{"n"}$$

(2:1.9)
$$\frac{1}{2+\sqrt{2}}(1,\sqrt{2},1) \qquad \text{when } x(t)=\text{"m"}$$

$$\frac{1}{1+\sqrt{2}}(\sqrt{2},1,0) \qquad \text{when } x(t)=\text{"s"}$$

Proof. We shall prove relations (2:1.6) and (2:1.8) by induction. Suppose (2:1.6) is true for a particular value of t. When there are t+1 steps to go, and both d.m.'s are in the middle lane (x(t+1)="m") they face the following problem: they have to find the probabilities $\tilde{u},\tilde{h},\tilde{d}$ such as to minimize the expected cost. The possible outcomes of the two independent randomizations are: a.) with probability $\tilde{h}^2$ both d.m.'s moved horizontally and have now expected cost M(t); b.) with probability $\tilde{u}^2+\tilde{d}^2$ both d.m.'s made a step aslant in the same direction and have now expected cost S(t); c.) with probability $1-\tilde{u}^2-\tilde{h}^2-\tilde{d}^2$ they came in different lanes and have now cost 0. This gives

$$M(t+1)=\min_{\tilde{u}+\tilde{h}+\tilde{d}=1}\{\tilde{h}^2M(t)+(\tilde{u}^2+\tilde{d}^2)S(t)\}$$

At the constrained minimum the gradient of the expression within "{ }" must be perpendicular to the surface $\tilde{u}+\tilde{h}+\tilde{d}=1$ which implies

$$\frac{\partial}{\partial\tilde{u}}\{..\} = \frac{\partial}{\partial\tilde{h}}\{..\} = \frac{\partial}{\partial\tilde{d}}\{..\}$$

i.e.

$$2\tilde{u}S(t)=2\tilde{h}M(t)=2\tilde{d}S(t)$$

together with the constraint this gives

$$\tilde{h}= \frac{S(t)}{2M(t)+S(t)}$$

$$\tilde{u}=\tilde{d}= \frac{M(t)}{2M(t)+S(t)}$$

at the minimum. Here we can already conclude that assuming (2:1.6) to be true for (a particular value of) t implies the second line of (2:1.8) to be true for t+1. Further, the value of the minimization is

$$M(t+1)= \frac{S^2(t)M(t)+2M^2(t)S(t)}{(2M(t)+S(t))^2} = \frac{S(t)M(t)}{2M(t)+S(t)}$$

Referring to (2:1.6), write $M(t)=2/a$ and $S(t)=4/b$ then

$$M(t+1)= \frac{(2/a)(4/b)}{4/a+4/b} = \frac{2}{a+b}$$

Now substitute the appropriate expressions for a and b to get

$$M(t+1)= \frac{2}{\lambda_1^{t-1}(2+5\lambda_1)+\lambda_2^{t-1}(2+5\lambda_2)}$$

Using (2:1.7) one can easily verify that $2+5\lambda_1=\lambda_1(1+2\lambda_1)$ and that $2+5\lambda_2=\lambda_2(1+2\lambda_2)$. Therefore

$$M(t+1)= \frac{2}{\lambda_1^{t}(1+2\lambda_1)+\lambda_2^{t}(1+2\lambda_2)}$$

which is the first equation of (2:1.6) with t+1 substituted for t. The recursion for the second equation of (2:1.6) is obtained in a completely analogous fashion, by considering the decision problem when both d.m.'s are in one of the side-lanes (x(t+1)="s", say). This leads to the minimization

$$S(t+1)=\min_{\tilde{u}+\tilde{h}=1}\{\tilde{u}^2M(t)+\tilde{h}^2S(t)\}$$

The minimum is attained at

$$\tilde{u}= \frac{S(t)}{M(t)+S(t)}$$

$$\tilde{h}= \frac{M(t)}{M(t)+S(t)}$$

which shows the third (and, with a trivial substitution, also the first) equation of (2:1.8), and further

$$S(t+1)= \frac{S(t)M(t)}{M(t)+S(t)}$$

Using the same substitutions as above we get

$$S(t+1)= \frac{4}{2a+b} = \frac{4}{\lambda_1^{t-1}(3+7\lambda_1)+\lambda_2^{t-1}(3+7\lambda_2)}$$

Using (2:1.7) one can easily verify that $3+7\lambda_1=\lambda_1(1+3\lambda_1)$ and that $3+7\lambda_2=\lambda_2(1+3\lambda_2)$. Therefore

$$S(t+1)= \frac{4}{\lambda_1^{t}(1+3\lambda_1)+\lambda_2^{t}(1+3\lambda_2)}$$

which is the second equation of (2:1.6) with t+1 substituted for t. To complete the induction argument we shall verify (2:1.6) for t=0. Using the fact that $1/\lambda_1+1/\lambda_2=-2$ and that $\lambda_1+\lambda_2=2$ we find

$$M(0)= \frac{2}{\lambda_1^{-1}(1+2\lambda_1)+\lambda_2^{-1}(1+2\lambda_2)} = \frac{2}{-2+4} =1$$

$$S(0)= \frac{4}{\lambda_1^{-1}(1+3\lambda_1)+\lambda_2^{-1}(1+3\lambda_2)} = \frac{4}{-2+6} =1$$

To prove (2:1.9) note that since $|\lambda_1|<1$ and $\lambda_2>1$ we have

$$\lim_{t\to\infty} \frac{S(t)}{M(t)} = \frac{4(1+2\lambda_2)}{2(1+3\lambda_2)} = \frac{6+4\sqrt{2}}{4+3\sqrt{2}} = \sqrt{2}$$

Thus

$$\frac{M(t)}{M(t)+S(t)} \to \frac{1}{1+\sqrt{2}} \; ; \; \frac{M(t)}{2M(t)+S(t)} \to \frac{1}{2+\sqrt{2}} \; ; \; \frac{S(t)}{M(t)+S(t)} \to \frac{\sqrt{2}}{1+\sqrt{2}}$$

which gives immediately (2:1.9) from (2:1.8). We say that the convergence is rapid, because numerical verification shows that already for t=3 $\tilde{u}$, $\tilde{h}$ and $\tilde{d}$ are within 1% of their limit-values.

(2:1.10) Properties. Symmetric solutions to symmetric team problems generally have the following properties

- realism
- simplicity
- fairness
- robustness
- no cost of convention

These four points are elaborated below

· A simple simulation program that makes the d.m.'s move on a video-screen according to the solution just derived, shows a very familiar scene. For example we may see a pattern as

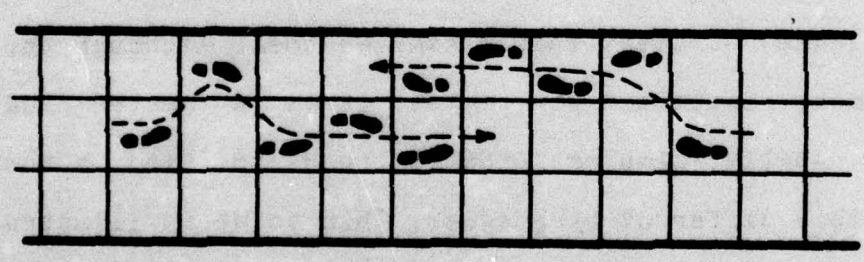


Figure 2:1.b. Sample pattern of the solution of the Corridor Problem, resulting from a randomized symmetric strategy-tuple.

in fig.2:1.b. This suggests that the concept of symmetry in team problems, as described in this section, is realistic in the sense that it models how, in some team problems that occur in everyday life, the asymmetry that is needed for a solution is introduced through randomization. Here the practical meaning of the mathematical concept of randomized decisions is that the decisions depend on factors too complex to be included in the model.

· Decision problems with more then one decision maker involved are, with our present knowledge about such problems, often hard (if not impossible) to solve. This is the case when the number of decision makers N=2, but even more so for large scale problems where N>>1. When such problems are symmetric team problems then the restriction to symmetric strategy-tuples can introduce great simplification. Under such restriction we need to find only a single strategy that minimizes cost. Although the search is among randomized strategies, it is a better understood and usually simpler problem then to find a tuple of N possibly different strategies. This point is illustrated in section (2:2) where we consider the problem of access control of N stations to a common communication wire.

· A solution to a symmetric team problem may require that one (or a few) of the decision makers makes an exceptional decision. Generally one would consider it as fair if it is

determined by a "lottery" who of the decision makers will be the exception.

· When the team members (decision makers) adopt randomized strategies that are not degenerate, the implication is that each decision maker does not count on specific strategies of his team members, but is prepared to cooperate with a whole range of possible strategies of his team members. As such, the symmetric solution is <u>robust</u> because it is not likely to break down when one of the team members makes an error, i.e. accidentally does not follow the strategy that was counted on. For example in the case of the Corridor Problem, when one team member insists on keeping the left lane then still a collision will most likely be avoided through the randomizations of the other team member (assuming T not too small). However, suppose we also allowed asymmetric strategies in the Corridor Problem, then the strategy-tuple where both team members keep their right lane*) achieves the minimum cost but it would not have the flexibility of avoiding a collision when one of the team members made the error of insisting on keeping the left lane.

· For a given symmetric team problem one can always get at least as low cost as the symmetric solution by allowing

---

*) Note that the strategy-tuple "both d.m.'s keep right" is not a symmetric solution, because it requires a priori agreement between the d.m.'s that not the equally good strategy-tuple "both d.m.'s keep left" will be used (cf. definition (2:1.1)).

asymmetric solutions. However it is not reflected in the cost function that asymmetric solutions in fact also bear the cost of establishing a convention that tells which asymmetric solution is to be chosen. Taking the latter cost in account one could find that sometimes the symmetric solution is cheaper then the unrestricted solution. Consider again the Corridor Problem. As long as we talk about pedestrian traffic, the expected cost of collision is so low that it is not worthwhile to have a widely established convention. In the case of automobile traffic, however, the cost of collision is so high that establishing a convention certainly pays off.

(2:1.11) Definition. A symmetric solution to a symmetric team problem is said to be asymptotically optimal if the difference in cost between the restricted (symmetric) solution and the unrestricted (possibly asymmetric) solution $\to 0$ when the number of time steps in which the problem must be solved $T \to \infty$ or the number of decision makers $N \to \infty$. In the latter case it is assumed that the cost is defined for all values of N.

(2:1.12) Corollary. The solution of the Corridor Problem is asymptotically optimal.

Proof. Obviously the asymmetric solution $(\tilde{u},\tilde{h},\tilde{d})=(1,0,0)$ for d.m.1 and $(\tilde{u},\tilde{h},\tilde{d})=(0,0,1)$ for d.m.2 at t=T, yields the

minimum cost possible: 0. The optimal symmetric solution yields cost M(T). From (2:1.6-7) it is clear that $\lim_{T\to\infty} M(T)=0$. Q.E.D.

(2.1:13) Remark. It would be valuable to be able to tell directly from the cost function of a symmetric team problem whether the symmetric solution is asymptotically optimal or not, without actually finding the minimum cost of the symmetric and the unrestricted solution. The next theorem is a first attempt in this direction. It gives sufficient conditions, in terms of the cost function, for asymptotic optimality when the number of decisions makers $N \to \infty$. The key point in the proof is the use of a limiting result (the strong law of large numbers). The sufficient conditions are stronger than necessary, because the cost for the access problem in the next section does not satisfy the condition but nevertheless asymptotic optimality is proved in (2:2.19). There is a strong connection between the next theorem and theorem (2:2.19) in the sense that the proof of (2:2.19) also relies on a limiting result (the convergence of the hypergeometric distribution to the binomial distribution). The difference is that in (2:2.19) we do not consider the number of decision makers that choose a certain strategy, but the number of decision makers that makes a certain decision.

(2:1.14) Theorem. The cost of a given symmetric team problem can always be written as $C(\underline{n})$, where the value of the j-th component of the vector $\underline{n}$ gives the number of decision makers that chose strategy j, j=1,...K (=number of alternative strategies for each decision maker). Suppose that the symmetric team problem is defined for any total number of decision makers N and that

$$C(\underline{n})=C(\lambda\underline{n})$$

for any $\underline{n}$ and λ such that $\underline{n}$ and $\lambda\underline{n}$ are (non-negative) integer vectors, i.e. the <u>cost depends only on the ratio</u> of numbers of decision makers choosing the different strategies. Suppose further that the obvious extension of $C(\cdot)$ to the space of vectors with non-negative rational components, which is defined by $C(\underline{q})=C(m\underline{q})$ with $q_j=w_j/v_j$ (=ratio of non-negative integers) and $m=\prod_{j=1}^{K} v_j$, is continuous. Then the solution to the symmetric team problem is <u>asymptotically optimal</u>.

<u>Proof.</u> The cost tensor which gives the cost for any strategy-tuple is by definition invariant under a permutation of indices. Therefore the cost depends not on who the decision makers are that choose the particular strategies but only on the number of decision makers that choose each strategy, which implies that the cost can be written in the form $C(\underline{n})$. Since $C(\cdot)$ is continuous on the space of positive rational vectors it can easily be extended

to a continuous function on the space of non-negative real vectors by $C(\underline{r})=\lim_{l\to\infty} C(\underline{q}(l))$ where $\{\underline{q}(l)\}_{l=1}^{\infty}$ is a sequence of non-negative rational vectors that converges to $\underline{r}$. Because $R_1\hat{=}\{\underline{r}\mid \sum_{j=1}^{K} r_j=1,\ r_j\geq 0\}$ is a compact set and because $C(\underline{r})=C(\lambda\underline{r})$, there exists a $\hat{\underline{r}}\in R_1$ such that for any non-negative vector $\underline{r}$: $C(\underline{r})\geq C(\hat{\underline{r}})$. Now consider the randomized symmetric strategy where each decision maker chooses with probability $\hat{r}_j$, the j-th strategy out of the finite set of possible strategies. Let the random vector $\underline{n}_i$ have all elements =0 except for the j-th element which is =1; j is determined by the randomization of decision maker i as the index of the strategy he chooses. Then $E\{\underline{n}_i\}=\hat{\underline{r}}$. According to the definition of $\underline{n}$ above we have $\underline{n}=\sum_{i=1}^{N}\underline{n}_i$ and the expected cost is $E\{C(\underline{n})\}=E\{C((1/N)\underline{n})\}$. From Kolmogorov's strong law of large numbers (cf. [7], p.124) we know that for any $\epsilon>0$ there is a N_ϵ such that for $N>N_\epsilon$ $\Pr\{|(1/N)\underline{n}-\hat{\underline{r}}|<\epsilon\}>1-\epsilon$. Together with the continuity of $C(\cdot)$ this implies that for N large enough, the expected cost of this randomized strategy-tuple will be arbitrarily close to $C(\hat{\underline{r}})$. Of course for every value of N, the expected cost of the optimal symmetric strategy-tuple and the cost of the optimal unrestricted strategy-tuple will lie between the expected cost of the randomized strategy-tuple under consideration and $C(\hat{\underline{r}})$. It follows that the symmetric solution must be asymptotically optimal.

2:2 The Access Problem in Wire Communication.

The concept of many devices sharing one communication wire (data-bus) is already well established in computer systems. With micro-processors becoming widely available at very low cost, the usefulness of this concept is greatly enhanced and it finds ever increasing application. The devices that share the communication wire are not limited anymore to the traditional elements of a computer system. For example we may now find systems that consist of a large number of measurement devices which periodically send data on water- and air quality to data processing units and which periodically receive instructions on what and how to measure over the same communication wire. Or, we may find a system consisting of sensors, actuators and minicomputers that control some production process and share one communication wire. Or we may find a set of computer systems that are connected by a high speed data-bus to provide the possibility of distributed computing. These examples form a tiny fraction of all possible applications of multi-access wire communication. The major advantages of one shared wire over say a network of connections between every pair of devices which may need to communicate among each other, are efficiency and simplicity. Sharing is efficient, because the devices with bursty communication traffic can now alternate in the use of the communication

wire. Simplicity and flexibility are inherent to the concept of one wire --one can add and take away devices or establish and eliminate logical communication links between pairs of devices without altering the simple topology of the system.

The purpose of this section is to show how the framework of non-classical control theory, in particular the new concept of symmetric team problems, gives insight to the different possible solutions to the access problem in multi-access wire communication. Loosely speaking, the access problem is the problem of determining efficiently, and in a decentralized way, which device will be next to put a signal on the communication wire after the wire becomes available again. It is not the purpose of this section to give a complete solution to all control problems in multi-access wire communication. In the description below, a few control aspects of multi-access wire communication will be mentioned, and it is indicated why the access problem is relatively the main control problem.

(2:2.1) Description. A device that is attached to the communication wire and which has a decision making process, which controls when the device will transmit on the wire, will be called a station. The stations can transmit information through the wire in the form of packets. A

packet is a string of bits that is composed of a header which contains the destination address of the packet and possibly other control information, a body which contains arbitrary binary information, and a tail which contains a checksum to verify correct transmission. Packets can have any length, but it is assumed that their length in bits divided by the bit-rate of the wire is much larger than the propagation delay between the far ends of the wire. We say that a station needs access to the wire if there are packets queued at that station to be transmitted on the wire. If a station does not need access then that means that the queue is empty, and the queue is assumed to be of infinite capacity. The controls that each station has are like a traffic light (without yellow). A station transmits a packet on the wire (i.e. attempts to access the wire) if it has green light and a packet in its queue, otherwise the station is silent. The history of the wire consists of alternating intervals of contention (different stations attempt to access the wire), acquisition (one station has sole access to the wire) and silence (no station needs access). We shall assume that eventually every packet that becomes queued at a station will be successfully transmitted. The objective is to minimize the total delay. Consider a period that starts at the end of an interval of silence and that ends at the start of an interval of silence. The total delay

in that period would be certainly minimized if this period was just a succession of intervals of acquisition. In that case the total delay in that period corresponds to the

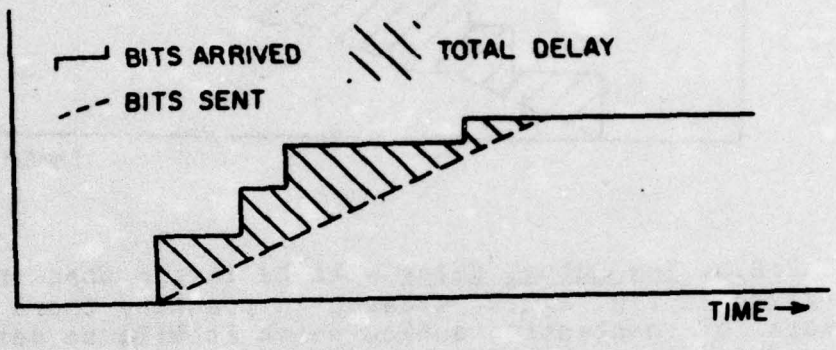


Figure 2:2.a. The shaded area gives a lower bound for the total delay incurred in a period between two intervals of silence. The area represents the total number of bit-seconds waiting for this particular sample of arrivals, when the wire is used at 100% of its capacity during this period.

shaded area in the graph given in fig. 2:2.a. However, given the fact that the packets to be sent are distributed over different stations and that we must determine in a decentralized way which station succeeds another station in acquisition of the wire, there will be intervals of contention during which no useful bits are transmitted, thereby increasing the total delay. See fig. 2:2.b (for the sake of exposition it is assumed that the period under consideration does not run into the next period). The actual delay can be brought as close as possible to its lower bound firstly by minimizing the number of intervals of

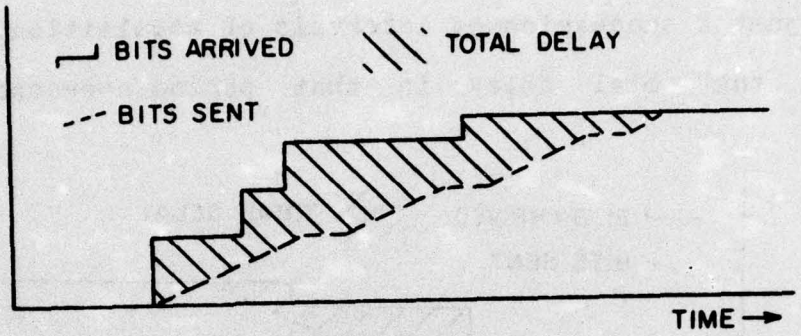


Figure 2:2.b. The actual delay will be larger than the lower bound given in fig. 2:2.a, because in practise there will be intervals of contention during which it will be determined which station is next in acquisition of the wire and no useful bits are transmitted then.

contention in a given period and secondly by minimizing the length of each period. To achieve the first point one obvious control is that each station will not transmit a packet on the wire as long as it "hears" that another station has not finished it's transmission. Apart from this control, which is known as "carrier sense" in packet radio applications [8], the number of intervals is predominantly determined by how many stations need access, which is mostly determined by outside events. So the main control problem relates to second objective: minimize the length of a contention interval; the rest of this section deals with this access problem.

The access problem can arise in two different ways. One: if during an interval of acquisition by one station at least

two other stations get a packet in their queue, then it is assumed that right after the present station gives up its acquisition all the stations which need access to the wire will start transmitting and will "hear" within one end-to-end delay time that there is a collision of packets. Two: if during an interval of silence two or more stations get a packet within one end-to-end delay time, in the same way these stations will transmit their packets concurrently and detect that there is a collision of packets. In both cases the stations involved are approximately synchronized (to within one end-to-end delay time). In the subsequent analysis we shall assume that the stations are perfectly synchronized. This assumption is justified if the time interval between subsequent decisions of a station ($\hat{=}$ one time slot) is constant and not less than three end-to-end delay times and if the stations transmit for exactly the first two-thirds of a time slot when there is a time slot of interference.

At this point we have the access problem sufficiently introduced and isolated to be able to represent it by a simple mathematical model.

(2:2.2) Model. The access problem arises whenever in some time slot, which we shall label as time slot 0, no station has acquisition of the wire but $n_0 \geq 2$ stations need access to

the wire out of a total population of N_0 stations. The n_0 stations will generate a collision in that time slot, thereby making the initial conditions known that there are N_0 stations out of which n_0 need access to the wire. The variables x_i describe which are those n_0:

$x_i=1$ if station i needs access
$=0$ if station i doesn't need access

and any subset of n_0 out of N_0 with $n_0 \geq 2$ is equally likely. Each station has to make at times $t=1,2,\ldots$ (where time t marks the beginning of time slot t) a binary decision $u_i(t)$:

$u_i(t)=1$ station i has green light during time slot t (has no effect when $x_i=0$)
$=0$ station i has not green light during time slot t

The state of the wire is represented by the variable r(t), which has as its value the number of stations that attempted to access the wire in time slot t:

$$r(t)=\sum_{i=1}^{N} x_i u_i(t)$$

(2:2.3) Problem. The problem is to find controls (=decisions) that minimize T, the duration of the interval of contention. T is given by:

$$T=\min\{t \mid u_i(t+1)x_i=1 \text{ for exactly one } i\in\{1,\ldots,N_0\}\}.$$

However the solution to this problem depends on what the information is on which the controls will be based. This will be specified in paragraph (2:2.5).

(2:2.4) Definition. A control law is defined as a map from the information of a station to the controls by that station.

(2:2.5) Model. (Information structure). The nature of the problem is such that the decisions have to be made in a decentralized way. Thus there is not a central controller who knows the vector $\underline{x}$, in which case the minimum cost would obviously be 0, because the central controller could just pick one of the n_0 stations with $x_i=1$. Decentralization implies that the only information on the initial conditions available to station i is the value of x_i (and implicitly N_0 and n_0). However if $x_i=0$ then the cost is independent of whatever controls are chosen. Therefore we can take the controls in the case $x_i=0$ the same as in the case $x_i=1$, which leads to the important conclusion that the control law of station i will not be a function of $\underline{x}$. With regard to further possible information we shall consider three different cases of information structures and thus create three different cases of problem (2:2.3) which we want to compare. The first case (case (u)) has the richest information structure and the corresponding solution is an unrestricted solution in the sense of section (2:1) in comparison with the second case. The second case (case (s)) differs from the first case only in the restriction that the corresponding solution must be a symmetric solution (cf. definition (2:1.1)). In terms of the information structure this means the control law can not be a function of the number assigned to a station, assuming we have numbered our

stations 1,2,...,N. In both cases, as the system evolves in time, each station will record the state history of the wire, on which future controls can be based; more specifically it is assumed that at time t each station knows of previous time slots how many of the n_0 stations that need access to the wire attempted to do so and whether it was one of those (i.e. the station also knows its control history). In the third case (case (m)) only the minimum information which is necessary to implement a control law is available. This is only one bit of information which indicates whether one station acquired the wire (i.e. the problem is over) or not (i.e. we still have the initial problem).

In summary we have in each case at time t the following information on which the controls can be based:

case (u)	i	:the number assigned to the station
	$r(t-1),r(t-2),\ldots,r(1)$	:history of number of stations with $x_i=1$ that attempted to access the wire
	$u_i(t-1),u_i(t-2),\ldots,u_i(1)$	:stations's control history
case (s)	$r(t-1),r(t-2),\ldots,r(1)$	:history of number of stations with $x_i=1$ that attempted to access the wire
	$u_i(t-1),u_i(t-2),\ldots,u_i(1)$	:stations's control history
case (m)	r(t-1)=1 (true or false)	whether 1 station acquired the wire

in both cases the initial information N_0, n_0 is implicitly assumed to be available. Note that if we consider the control laws which are in effect at the different stations to be available information, then in case (u) the control history becomes redundant information because it can always be reconstructed. In case (s), as we shall see in solution (2:2.10), the controls will be made dependent on random events (randomized decisions), in that case the control history also becomes redundant information when each station records the outcomes of its randomizer.

(2:2.6) Remark. It is somewhat unusual to assume the information r(t) to be available, customarily one would assume only a reduction R of r(t) to be available, given by R(0)="silence", R(1)="successful access" and R(r)="collision" for $r \geq 2$. The reason for our assumption is that it is easier to deal with a value r(t) than with a probability distribution over the possible values $1,\ldots,n_0$, which would otherwise be the case. Further in defense of this assumption we want to mention that it is not inconceivable that each station has a hardware device which can measure r(t) from the intensity of the signal on the wire. In the same way we can defend the assumption that a station knows n_0, the number of stations that need access to the wire initially. That is, the access problem will presumably arise only right after a collision has occurred

among the stations that need access to the wire. From this initial collision the stations will know the value of n_0.

(2:2.7) Solution. (Case (u)). Given the minor restriction that at each point in time the set of stations that are considered for possibly having green light form a subset of the original N_0 stations out of which a known number number need access, then the minimum expected length of the interval of contention is

$$T(N_0,n_0)$$

with $T(\cdot,\cdot)$ given by the recursive relation:

(2:2.8)

$$T(N,n)=\min_u\{\sum_{r=0}^{n}H(r|N,n,u)(L(r)+\min\{T(u,r),T(N-u,n-r)\})\}$$

where

$$L(r)=0 \text{ if } r=1$$
$$=1 \text{ otherwise}$$

and

$$H(r|N,n,u)=\frac{\binom{n}{r}\binom{N-n}{u-r}}{\binom{N}{u}} \qquad \text{(Hypergeometric distribution)}$$

the "boundary conditions" are given by

$$T(N,0)=\infty \ \forall N$$
$$T(N,1)=0 \ \forall N$$

The optimal controls are also given by (2:2.8): if at the beginning of time slot t there are N stations considered out of which n need access, then $\hat{u}$, the argument of the minimization in (2:2.8) gives the optimal number of stations in the set under consideration to have green light for time

slot t. Let the number of stations that transmit a packet in time slot t be r, then the stations to be considered for green light in the next time slot are the $\hat{u}$ stations that just had green light (out of which r need access) if $T(\hat{u},r) \leq T(N-\hat{u},n-r)$ otherwise it is the set of $N-\hat{u}$ stations that did not have green light (out of which n-r need access).

Proof. Suppose that the minimum expected cost for problem (2:2.4) with initial conditions N, n is given by T(N,n). From the point of minimizing cost it is only the total number of stations with green light --let's denote this total by u-- that matters. The latter is explained by the fact that the control laws will not be functions of the values x_i and that further the only difference between the stations is the number that is assigned to them. This difference is useful, even necessary, to design a set of control laws (one for each station) which generate controls such that the total number of stations with green light is u, but the effect of the controls is independent of whatever numbers have been assigned to the stations with green light. The result of having a total of u stations with green light is that (by definition) only the r stations that require access will put a signal on the wire. The probability distribution of r is a hypergeometric distribution with parameters N, n and u, because any combination of which n

out of N need access is equally likely*) . Given u and r we can distinguish two sets of stations: one set has u members out of which r need access to the wire, the other set has N-u members out of which n-r need access to the wire. For each set we face the same problem as the initial problem, with minimum expected costs T(u,r) and T(N-u,n-r), respectively. Given the restriction we shall proceed with that set which gives the lowest minimum expected cost, and this explains the inside minimization in (2:2.8). We have to add a term L(r) because whenever the attempt fails ($r \neq 1$), one time slot is "wasted". Finally, in (2:2.8) the expected value is obtained by a weighted summation, and the minimization over u gives the minimum expected cost and indicates the optimal control. To initialize the recursion note that for n=1 and n=0 the minimum expected time before one station acquires the wire is obvious.

(2:2.9) Table. The following table gives for entries N,n the optimal number of stations $\hat{u}$ to have green light (i.e. the argument of the minimization over u in (2:2.8)) and the minimum expected cost. The elements in the table have the format

*) Remember the probability-theorist's vase in which there are N beads, n red ones and the other N-n are white. Out of the vase we take (blindfolded) u beads. Then the number of red beads in our drawing is modeled by a random variable that has a hypergeometric distribution with parameters N, n and u.

$\hat{u}$;T(N,n).

N=	n=2	n=3	n=4	n=5	n=6	n=7	n=8	n=9
2	1;0.0							
3	1;0.3	1;0.0						
4	2;0.3	1;0.3	1;0.0					
5	2;0.5	2;0.4	1;0.2	1;0.0				
6	3;0.5	2;0.4	1;0.4	1;0.2	1;0.0			
7	3;0.6	3;0.5	2;0.5	1;0.3	1;0.1	1;0.0		
8	4;0.6	3;0.5	2;0.5	2;0.5	1;0.3	1;0.1	1;0.0	
9	4;0.6	3;0.6	2;0.6	2;0.5	1;0.4	1;0.3	1;0.1	1;0.0
10	5;0.7	4;0.6	2;0.6	2;0.6	2;0.5	1;0.4	1;0.2	1;0.1
11	5;0.7	4;0.6	3;0.7	2;0.6	2;0.5	1;0.5	1;0.3	1;0.2
12	6;0.7	5;0.6	3;0.7	2;0.6	2;0.6	2;0.5	1;0.4	1;0.3
13	6;0.7	5;0.6	4;0.7	2;0.7	2;0.6	2;0.6	2;0.5	1;0.4
14	7;0.7	6;0.6	4;0.7	3;0.7	2;0.6	2;0.6	2;0.5	1;0.5
15	8;0.7	6;0.7	4;0.8	3;0.7	2;0.7	2;0.6	2;0.6	2;0.5
16	8;0.7	6;0.7	4;0.8	4;0.8	3;0.7	2;0.6	2;0.6	2;0.6
17	9;0.8	7;0.7	4;0.8	4;0.8	3;0.8	2;0.7	2;0.6	2;0.6
18	9;0.8	7;0.7	4;0.8	4;0.8	3;0.8	2;0.7	2;0.7	2;0.6
19	10;0.8	8;0.7	5;0.8	4;0.8	3;0.8	3;0.8	2;0.7	2;0.6
20	10;0.8	8;0.7	5;0.9	4;0.8	4;0.8	3;0.8	2;0.7	2;0.7
21	10;0.8	8;0.7	6;0.9	4;0.8	4;0.8	3;0.8	2;0.8	2;0.7
22	11;0.8	9;0.7	6;0.9	4;0.9	4;0.8	3;0.8	3;0.8	2;0.7
23	12;0.8	9;0.7	6;0.9	4;0.9	4;0.8	4;0.8	3;0.8	2;0.7
24	12;0.8	10;0.7	7;0.9	5;0.9	4;0.9	4;0.8	3;0.8	3;0.8
25	12;0.8	10;0.7	7;0.9	5;0.9	4;0.9	4;0.8	3;0.8	3;0.8
26	13;0.8	10;0.7	8;0.9	6;0.9	4;0.9	4;0.8	4;0.8	3;0.8
27	14;0.8	11;0.7	8;0.9	6;0.9	4;0.9	4;0.9	4;0.8	3;0.8
28	14;0.8	11;0.7	8;0.9	6;0.9	4;0.9	4;0.9	4;0.8	3;0.8
29	14;0.8	12;0.7	8;0.9	6;0.9	5;0.9	4;0.9	4;0.9	4;0.8
30	15;0.8	12;0.7	8;0.9	6;1.0	5;0.9	4;0.9	4;0.9	4;0.8
31	15;0.8	13;0.7	8;0.9	7;1.0	6;0.9	4;0.9	4;0.9	4;0.9
32	16;0.8	13;0.7	8;0.9	7;1.0	6;0.9	4;0.9	4;0.9	4;0.9
33	16;0.8	13;0.7	9;0.9	7;1.0	6;1.0	5;0.9	4;0.9	4;0.9
34	17;0.9	14;0.7	9;0.9	8;1.0	6;1.0	5;0.9	4;0.9	4;0.9
35	17;0.9	14;0.7	10;1.0	8;1.0	6;1.0	5;1.0	4;0.9	4;0.9

N=	n=2	n=3	n=4	n=5	n=6	n=7	n=8	n=9
36	18;0.9	15;0.7	10;1.0	8;1.0	6;1.0	6;1.0	4;0.9	4;0.9
37	19;0.9	15;0.7	10;1.0	8;1.0	7;1.0	6;1.0	4;0.9	4;0.9
38	19;0.9	15;0.7	11;1.0	8;1.0	7;1.0	6;1.0	5;0.9	4;0.9
39	19;0.9	16;0.7	11;1.0	8;1.0	7;1.0	6;1.0	5;1.0	4;0.9
40	20;0.9	16;0.7	12;1.0	8;1.0	8;1.0	6;1.0	5;1.0	4;0.9
41	20;0.9	17;0.7	12;1.0	8;1.0	8;1.0	6;1.0	6;1.0	4;0.9
42	21;0.9	17;0.7	12;1.0	8;1.0	8;1.0	6;1.0	6;1.0	5;1.0
43	21;0.9	18;0.7	12;1.0	9;1.0	8;1.0	7;1.0	6;1.0	5;1.0
44	22;0.9	18;0.7	13;1.0	9;1.0	8;1.0	7;1.0	6;1.0	5;1.0
45	22;0.9	18;0.7	13;1.0	10;1.0	8;1.0	7;1.0	6;1.0	5;1.0
46	23;0.9	19;0.7	14;1.0	10;1.0	8;1.0	7;1.0	6;1.0	6;1.0
47	24;0.9	19;0.7	14;1.0	10;1.0	8;1.0	8;1.0	6;1.0	6;1.0
48	24;0.9	20;0.7	14;1.0	10;1.0	8;1.0	8;1.0	6;1.0	6;1.0
49	25;0.9	20;0.7	14;1.0	11;1.0	8;1.0	8;1.0	7;1.0	6;1.0
50	25;0.9	20;0.7	15;1.0	11;1.0	8;1.0	8;1.0	7;1.0	6;1.0

(2:2.10) Solution. (Case (s)). Given the same minor restriction as in solution (2:2.7), and in addition the restriction that --according to the specification of case (s)-- the solution must be symmetric, then the minimum expected length of the interval of contention is

$$T(n_0)$$

with $T(\cdot)$ given by the recursive relation:

(2:2.11)

$$T(n)=\min_{v}\left\{\frac{B(0|n,v)+B(n|n,v)+\sum_{r=2}^{n-1}B(r|n,v)(1+\min\{T(r),T(n-r)\})}{1-B(0|n,v)-B(n|n,v)}\right\}$$

where

$$B(r|n,v)=\binom{n}{r}v^r(1-v)^{n-r}$$ (Binomial distribution)

the "boundary conditions" are given by

$$T(0)=\infty$$
$$T(1)=0$$

and the optimal controls are also given by (2:2.11) in the following way: if at the beginning of time slot t there are N stations considered for possibly having green light out of which n need access, then $\hat{v}$, the argument of the minimization in (2:2.11) gives the optimal probability with which the stations in the set under consideration are to have green light for time slot t. Let the number of stations that transmit a packet in time slot t be r. Then the set of stations to be considered for green light in the next time slot are the stations that just had green light (out of which r need access) if $T(r) \leq T(n-r)$ otherwise it is the subset of the stations that just were under consideration but that did not have green light (out of which n-r need access).

Proof. Assume that (2:2.11) is already proven to be correct when the argument of $T(\cdot)$ takes values $2,\ldots,n-1$.

In the symmetric case it is necessary to consider randomized decisions (cf. theorem (2:1.2)). For each time slot a binary decision has to be made by the stations under consideration, therefore we have to specify for each time slot the probability of success for the Bernoulli trials that will determine at each station in the set under consideration the probability with which that station will have green light. Let the probability of the Bernoulli trials be v. Then the probability distribution of $\tilde{r}$, the number of stations with

$x_i=1$ within the set under consideration that had green light, depends only on n, the number of stations within the set under consideration that initially needed access --only those stations can contribute to $\tilde{r}$-- and has the binomial distribution with parameters n and v. After a given outcome r of the random variable $\tilde{r}$ there are two sets of stations: one set of stations that just had green light out of which r need access and another set of stations that did not have green light out of which n-r need access. Given the minor restriction we shall proceed optimally with that set with the smaller expected time before one station acquires the wire. Lets denote the expected time before one station acquires the wire with this strategy by $T_v(n)$. With probabilities $B(0|n,v)$ and $B(n|n,v)$ r will take the values 0 and n respectively; in both cases we spend 1 time slot and then face the same expected time $T_v(n)$ before one station acquires the wire. With probability $B(1|n,v)$ this attempt is successful and the time spent is 0. Summarizing, we have for $T_v(n)$ the relation

(2:2.12)
$$T_v(n)=(B(0|n,v)+B(n|n,v))(1+T_v(n)) + \sum_{r=2}^{n-1} B(r|n,v)(1+\min\{T(r),T(n-r)\})$$

Now (2:2.11) is obtained by bringing $T_v(n)$ to the left-hand side and minimizing over v. Note that the "boundary conditions" are obvious; the solution is now proven by induction.

(2:2.13) Table. The following table gives for entries n the optimal probability with which the stations in the set under consideration should have green light (i.e. the argument of the minimization over v in (2:2.11)) and the minimum expected cost. The elements in the table have the format

$$\hat{v};$$
$$T(n)$$

n=2	n=3	n=4	n=5	n=6	n=7	n=8	n=9
0.50; 1.0	0.41; 0.8	0.31; 1.1	0.24; 1.2	0.19; 1.3	0.17; 1.3	0.15; 1.3	0.13; 1.3

It can be seen from tables (2:2.9) and (2:2.13) that T(N,n) approaches T(n) as N gets larger. Indeed we have here what was defined in section (2:1) as asymptotic optimality of the symmetric solution. This will be stated in theorem (2:2.19). The proof of the theorem is based on the two lemmas given below and the fact that the hypergeometric distribution approaches the binomial distribution as the population-size $N \to \infty$.

(2:2.14) Lemma. Let T(N,n) be the minimum expected time needed before one station has sole access when n out of N need access, as given by (2:2.8). Then for fixed n, T(·,n) is non-decreasing or more precisely

$$T(N+1,n) \geq T(N,n) \quad \forall N \geq n$$

and the <u>limit for N → ∞ exists</u>, i.e. ∃ a finite number T_n such that

$$T_n = \lim_{N\to\infty} T(N,n)$$

<u>Proof.</u> To show that $T(\cdot,n)$ is non-decreasing consider the situation where there are N stations out of which n need access. Applying the optimal strategy the expected time before one station has sole access is T(N,n). Another --but probably suboptimal-- way of solving the same access problem is by adding one (imaginary) station which does not need access and applying the optimal strategy for the case where n out of N+1 need access. Thus we ignore the knowledge that the additional station belongs to the set that doesn't need access, and the expected time before a single station has access is T(N+1,n). By definition we can not improve on the optimal strategy, certainly not by this particular procedure. Therefore

$$T(N+1,n) \geq T(N,n).$$

The existence of the limit now follows directly from the fact that T(N,n) is bounded by T(n):

$$T(N,n) \leq T(n) \quad \forall N \geq n$$

Each side of the inequality is the cost achieved by an optimal access strategy, but in the case corresponding to the left-hand side the strategy is based on more information, which implies the inequality.

(2:2.16) Lemma. Let $\hat{u}$ be the optimal number of stations with green light, as given by the minimization in (2:2.8). Then for fixed n, $\hat{u} \to \infty$ and $N-\hat{u} \to \infty$ as $N \to \infty$. More precisely,

$$\forall Z\ \exists M \text{ such that } N>M \Rightarrow \hat{u}>Z \text{ and } N-\hat{u}>Z$$

Proof. Suppose the lemma is not true. Then $\exists Z$ such that $\forall M$, no matter how large, $\exists N>M$ with the corresponding $\hat{u}<Z$. Given any small $\theta>0$ $\exists M_\theta$ such that $\forall N>M_\theta$

(2:2.17) $$H(0|N,n,\hat{u})>1-\theta \qquad \text{for } \hat{u}<Z$$

i.e. when the total number of stations is made large enough and the number of stations with green light is known to be less than Z then the probability of 0 stations accessing the wire can be made arbitrary close to 1. From the existence of $\lim_{N\to\infty} T(N,n)$ we know that for the same θ, $\exists M_\theta$ such that for $N>M_\theta$

(2:2.18) $$T(N,n)-\theta<T(N-\hat{u},n) \qquad \text{for } \hat{u}<Z$$

Now let $M=\max\{M_\theta, M_\theta\}$ and take $N>M$ such that the corresponding $\hat{u}<Z$. From (2:2.8) and the definition of $\hat{u}$ we get directly that

$$T(N,n) \geq H(0|N,n,\hat{u})(1+T(N-\hat{u},n))$$

Using the inequalities (2:2.17) and (2:2.18) this gives

$$T(N,n)>(1-\theta)(1+T(N,n)-\theta)$$

This leads to a contradiction since θ could be chosen arbitrarily small, which proves the $\hat{u} \to \infty$ -part of the lemma. To proof the $N-\hat{u} \to \infty$ -part we can repeat the same

argument with N-û and û mutually interchanged (except where û appears as an argument of the hypergeometric distribution H(·)) and with H(n|N,n,û) instead of H(0|N,n,û) i.e. when the total number of stations is made large enough and the number of stations with green light is known to differ less than Z from N then the probability of all n stations accessing the wire can be made arbitrary close to 1.

(2:2.19) Theorem. The <u>symmetric solution</u> (2:2.10) is <u>asymptotically optimal</u> in the sense that for fixed n

$$\lim_{N\to\infty} T(N,n)=T(n)$$

i.e. the minimum cost of the unrestricted solution (2:2.7) approaches the minimum cost of the symmetric solution as the number of stations $N \to \infty$.

<u>Proof.</u> First assume the theorem already proven for n'=1,...,n-1. In lemma (2:2.14) we defined $\lim_{N\to\infty} T(N,n)$ as T_n and from the proof of the lemma it is obvious that $T_n \leq T(n)$. So it is sufficient to show that $T_n \geq T(n)$. Let ϵ be an arbitrary small positive number. As a direct consequence of lemma (2:2.14) $\exists Z$ (for this ϵ) such that $u>Z \Rightarrow T(u,n')>T_{n'}-\epsilon$ for n'=1,...,n. From a limit theorem for the hypergeometric distribution (see [9] p.59) it follows that $\exists M_\epsilon$ such that $N>M_\epsilon \Rightarrow H(r|N-n,u)>B(r|n,\frac{u}{N})-\epsilon$, where B(·) stands for binomial distribution. By lemma (2:2.16) $\exists M_Z$ such that for $N>M_Z$ the argument of the minimization in (2:2.8) is guaranteed to lie between Z and N-Z. Therefore we can derive from (2:2.8) that for $N>\max\{M_\epsilon, M_Z\}$

$$T_n \geq T(N,n)$$

$$= \min_{Z<u<N-Z} \{ \sum_{r=0}^{n} H(r|N,n,u)(L(r)+\min\{T(u,r),T(N-u,n-r)\})\}$$

$$> \min_{v} \{ \sum_{r=0}^{n} (B(r|n,v)-\epsilon)(L(r)+\min\{T_r,T_{n-r}\}-\epsilon)\}$$

Since this can be shown for any $\epsilon>0$ we may conclude that

$$T_n \geq \min_{v} \{ \sum_{r=0}^{n} (B(r|n,v))(L(r)+\min\{T_r,T_{n-r}\})\}$$

This inequality can equivalently be formulated as

$$T_n \geq \{ \sum_{r=0}^{n} (B(r|n,\hat{v}))(L(r)+\min\{T_r,T_{n-r}\})\} \quad \text{for some } \hat{v}$$

Now T_n can be brought to the left-hand side, and using the assumption at the beginning of the proof and the facts that $L(1)=T(1)$, $T(0)=\infty$ we find

$$T_n \geq \frac{B(0|n,\hat{v})+B(n|n,\hat{v})+\sum_{r=2}^{n-1} B(r|n,\hat{v})(1+\min\{T(r),T(n-r)\})}{1-B(0|n,\hat{v})-B(n|n,\hat{v})} \quad \text{for some } \hat{v}$$

Which implies

$$T_n \geq \min_{v} \{ \frac{B(0|n,v)+B(n|n,v)+\sum_{r=2}^{n-1} B(r|n,v)(1+\min\{T(r),T(n-r)\})}{1-B(0|n,v)-B(n|n,v)} \}$$

The right-hand side is the same as in equation (2:2.11), thus

$$T_n \geq T(n)$$

To complete the induction argument observe that for n=1 the inequality is trivial because $T(N,1)=T_1=T(1)=0$. Q.E.D.

Finally we shall consider the case where there is only 1 bit of on line information on which to base decisions, besides the knowledge of the initial conditions (n_0,N_0). This cost will give us an upperbound on the minimum

cost for many other cases that are not treated here, such as the case where the stations can determine whether there is a collision but not how many stations are involved in the collision (cf. remark (2:2.6)). This solution is also mentioned in [10], where it is recognized that n_0 may also not be available information. Therefore, in the paper just mentioned, the solution is combined with a simple heuristic estimation procedure for n_0.

(2:2.20) Solution. (Case (m)). When the on-line information tells only whether in the last time slot one station was successful in acquiring the channel (in which case the problem ceases) or not, then the minimum time before one station acquires the channel is

(2:2.21) $$T_{n_0}^{(m)} = (1-1/n_0)^{n_0-1} - 1$$

and the optimal strategy is that each station attempts each time slot with probability $1/n_0$ to access the wire as long as no station has so far been successful in acquiring the channel.

Proof. Let n be the number of stations that need access and suppose that in the first time slot all stations have green light with probability v. Then this attempt will be successful with probability $p_1 \hat{=} nv(1-v)^{n-1}$ and the cost (i.e. number of time slots before one station acquired the channel) is 0. But with probability $1-p_1$ the attempt will be unsuccessful in which case we incur cost 1 and we face the

second time slot the same problem as we did initially. Therefore the minimum cost must satisfy the following equation

$$T_n^{(m)} = \min_v \{(1-p_1)(1+T_n^{(m)})\}$$

(2:2.22)
$$\Rightarrow T_n^{(m)} = \min_v \{\frac{1-p_1}{p_1}\} \qquad p_1 = nv(1-v)^{n-1}$$

by straightforward calculus we find that the minimum is attained for v=1/n. Substituting this value of v in the equation above gives us immediately (2:2.21).

(2:2.23) Table. The following table gives for entries n the optimal probability with which the stations should have green light (i.e. the argument of the minimization over v in (2:2.22)) and the minimum expected cost. The elements in the table have the format

$$\hat{v};$$
$$T_n^{(m)}$$

n=2	n=3	n=4	n=5	n=6	n=7	n=8	n=9
0.50; 1.0	0.33; 1.3	0.25; 1.4	0.20; 1.4	0.17; 1.5	0.14; 1.5	0.13; 1.5	0.11; 1.6

(2:2.24) Conclusion. Comparing tables (2:2.9), (2:2.13) and (2:2.23) we see that the cost of the symmetric solution is close to the cost of the unrestricted solution for large values of N_0 --say, N_0=50 or larger-- and small values of

n_0. It is only important to consider small values of n_0, because for a wire which is not overloaded $n_0=2$ is the most likely initial value and each next higher value is an order of magnitude less likely than its preceding value. When the wire is overloaded other controls than the ones considered in this section should be employed. Note that the solutions of case (s) and case (m) are independent of N_0. This independence enhances the properties of simplicity and robustness. When some additional cost is added to the derived cost of case (u) to account for greater complexity and less robustness, then the solution of case (s) will always be superior the solution of case (u) for N_0 large enough. The numerical results suggest that this value need not be unpractically large. If simplicity is a factor with great weight, then even case (m) could have the best solution.

CHAPTER 3: DECENTRALIZED CONTROL IN PACKET SWITCHED SATELLITE COMMUNICATION

3:1 Introduction and Model.

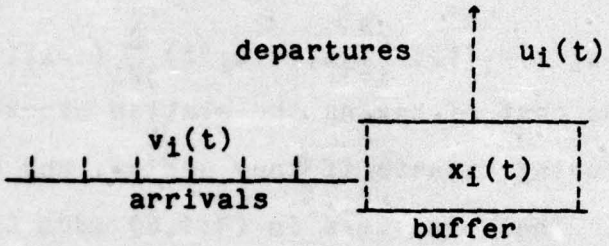
The packet switched satellite multi-access broadcast channel is becoming an ever more important medium for digital communication. Finding control schemes that ensure an efficient use of the channel is a non-trivial problem, mainly because any station that requires access to the channel at a given time can only determine with a fixed, significant delay whether another station accesses the channel at the same time, where the delay is the propagation delay for electromagnetic waves from a ground station to the satellite and back to the stations plus the transmission delay for putting one packet on the channel. When two or more stations access the channel at the same time a "collision" occurs, in which case the packets involved need to be retransmitted. We have here a control problem with many decision makers (the stations) each having partial information on the state of channel and the states of the stations.

(3:1.1) Remark. In the following model we are dealing with "new" packets. Transmitted packets are usually stored at a station: for comparison with the echo from the satellite to

determine whether a collision occurred and to have them available for retransmission when there was a collision. Retransmissions enter only in the present formulation through the cost of a collision K.

(3:1.2) Model. (Description.) Each station has a single buffer that can hold one packet. Packets are of fixed length and it takes one unit of time to take in or to send out one packet. The arrivals into the buffer and departures from the buffer are slotted, i.e. they start at integer instants of time. The arrivals at one station are independent from those at all other stations and form a Bernoulli process with constant parameter p, which is the same for all stations. This means that, at each station, a time slot will bring a packet with probability p. The departures are controlled by the stations on a decentralized basis. There will be a packet leaving a station, up to the satellite, when at the beginning of a time slot the buffer of that station is full and the station has "green light on". The buffer may be filled with a new packet at the same time that the previous packet goes out. Arrivals that occur when the buffer is full are refused. If two or more stations transmit a packet at the same time, then these transmissions will be unsuccessful; it is said that a collision occurred.

(3:1.3) Model. (Diagram.) for each station i we have:



The variables x,u and v for station i take values in {0,1} according to

$x_i(t)=0$: buffer is empty
$=1$: buffer is full

$u_i(t)=0$: red light
$=1$: green light

$v_i(t)=0$: no packet arrives
$=1$: packet arrives

$x_i(t)$ is the state of station i, $u_i(t)$ is the control of station i and $\{v_i(t) \mid t=1,2,...\}$ is a Bernoulli process with parameter p.

(3:1.4) Expressions. Although the variables involved are basically Boolean, we shall write the expressions for the dynamics and cost as real algebraic expressions for later convenience. The next state of station i is given by

(3:1.5) $$x_i(t+1)=v_i(t)+(1-u_i(t))(1-v_i(t))x_i(t)$$

We define units of cost by stating that one packet which waits one time slot (because $u_i(t)=0$) represents one unit of delay. Let the controls of all stations at all time slots under consideration $\{u_i(t) \mid i=1...,N,\ t=1,...,T\}$ be given. Then the expected cost for the whole system expressed in units of delay is given by the somewhat clumsy expression

(3:1.6) $$C(u)=E\{\sum_{t=1}^{T}\{\sum_{i=1}^{N}(1+B)x_i(t)(1-u_i(t))$$
$$+K[1-\prod_{i=1}^{N}(1-x_i(t)u_i(t))-\sum_{i=1}^{N}x_i(t)u_i(t)\prod_{j\neq i}^{N}(1-x_i(t)u_i(t))]\}\}$$

where B is the cost of having the station blocked, i.e. the cost of refusing packets if they arrive, and K is the cost of collision. The first term in (3:1.6) adds (1+B) units of delay whenever a buffer is full and no transmission takes place. In the second term the factor which multiplies K takes the value 1 if at time t there are two or more stations which have green light and their buffers full; otherwise, its value is 0.

The objective, of course, is to minimize cost. However at this point this statement is not well-defined. The reason is that the expression (3:1.6) gives the expected cost for given values of $\{u_i(t)\mid i=1,\ldots,N,\ t=1,\ldots,T\}$, but these values will typically not be predetermined. Instead they will be a function of the information available to the stations at every instant of time, which by the nature of our model is a random variable and cannot be known a priori. Now we shall define the information on which station i at time t could base its decision $u_i(t)$.

(3:1.7) Definition. When there is a team of N decision makers, each one observing a component $x_i(t)$ of the state vector $\underline{x}(t)$, and if each decision maker can inform all others about his observation with delay d (integer units of

time) then we call this a <u>delayed sharing of state information structure.</u> It is assumed that each decision maker has no limitation with respect to storing information. This definition is a variant of a more general definition of delayed sharing patterns [11].

(3:1.8) Definition. An <u>admissible decision rule</u> or <u>admissible control law</u> is a collection of maps $g=\{g_i(t;\cdot)\mid i=1,\ldots,N,\ t=1,\ldots,T\}$ which determine i's control at time t as a function of i's information at time t. That means we can write

$$\textbf{(3:1.9)}\qquad u_i(t)=g_i\left(t;\begin{matrix}x_1(t-d),\ldots,x_1(1)\\ \vdots\qquad\qquad\ \vdots\\ x_i(t-d),\ldots,x_i(1)\\ \vdots\qquad\qquad\ \vdots\\ x_N(t-d),\ldots,x_N(1)\end{matrix}\right)$$

in the case of delayed sharing of state information. We shall denote the class of admissible control laws by G. An admissible control law $g=\{g_i(t;\cdot)\mid i=i,\ldots,N,\ t=1,\ldots,T\}$ is said to be open loop if all the functions $g_i(t;\cdot)$ are constant maps, i.e. the decision of station i at time t is independent of its information at time t. The subclass of G which contains all open loop control laws will be denoted by G_o.

(3:1.10) Remark. Besides the information which is mentioned in definition (3:1.7) and written out in (3:1.9) explicitly, it is also assumed that <u>implicit information</u> is available to the stations, namely: the underlying model,*) the value of the parameter p of the arrival processes and the control law which is in effect. One consequence of this assumption is that station i not only knows the state history of the other stations with delay d, but it also knows their control history with the same delay. Another consequence is that unknown quantities such as $u_j(s)$, $x_j(s)$ $j \neq i, s > t-d$ are well defined random variables from the viewpoint of station i.

*) The knowledge of station i that it <u>is</u> number i, whence its control law can be a function of the number assigned to it, is here considered implicit in knowing the underlying model. Thus, in terms of chapter 2, symmetric solutions are not considered. The motivation is that, compared with a wire communication system, there are in a satellite communication system not a large number of stations; a-priori agreement on which station has which number should be assumed.

(3:1.11) Problem Formulation. Now we have defined the class of admissible control laws, we can state the problem is we are trying to solve. First note that for a given control law g, the quantities that appear in the expression for cost (c.f.(3:1.6)) are well defined random variables. Let's denote the expected cost for given g by C(g). Then the problem is to find g which minimizes C(g), i.e.

$$\min_{g \in G} \{C(g)\}$$

3:2 Optimal Control Law for New Packets.

We start with a theorem which reduces the complexity of our problem tremendously by stating that, with a reasonable assumption, the optimal control law for our single buffer model, with delayed sharing of state information, will be an open loop control law.

(3:2.1) Theorem. Assume that each station i has green light at least once within any interval of d consecutive time slots:

$$\forall t\ \exists r,\ t \le r < t+d \text{ such that } u_i(r)=1$$

Then the minimum cost will be attained within the subclass G_o:

$$\min_{g \in G_o}\{C(g)\} = \min_{g \in G}\{C(g)\}$$

Proof. It is sufficient to show that at any instant any station i cannot improve its decision by using its information instead of no information. If $x_i(t)=0$ then the cost incurred at stage t as well as the next state $x_i(t+1)$ is independent of $u_i(t)$ (c.f. (3:1.5) and (3:1.6)). Therefore $u_i(t)$ can be taken the same as when $x_i(t)=1$, which implies that $g_i(t;\cdot)$ can be a constant map with respect to $x_i(t)$. Furthermore, note that since $x_i(t-1),\ldots,x_i(t-d+1)$ are unknown to the other stations and since the arrival processes at the stations are uncorrelated, they give no information on the present state of the other stations,

which implies that $g_i(t;\cdot)$ also can be a constant map with respect to $x_i(t-1),\ldots,x_i(t-d+1)$. At this point the only possibly relevant arguments of the function $g_i(t;\cdot)$ are $\{x_j(s) \mid j=1,\ldots,N,\ s=t-d,t-d-1,\ldots,2,1\}$, which we call the delayed state history. However, because of the assumption of the theorem, we know that all stations have had $u_j(r)=1$ for at least one value of r in the interval $t-1,\ldots,t-d$. If $u_j(r)=1$ then $x_j(r+1)$ is independent of the value of $x_j(r)$, and as a consequence the present state of the system is independent of the delayed state information and therefore $g_i(t;\cdot)$ can be a constant map with respect to this information as well. Altogether it means that $g_i(t;\cdot)$ can be restricted to be a constant map without increasing the value of the minimum cost.

The next theorem states that there is no reason to enlarge the solution space to include randomized decisions (however, cf. previous footnote). For clarity, we shall consider only randomizing over the set of open loop control laws G_o. However a similar argument could be given if we were to randomize over the set G. Thus theorem (3:2.3) is in fact independent of theorem (3:2.1)

(3:2.2) Definition. A <u>randomized decision rule</u> or <u>randomized control law</u> is given by a probability measure Pm on the set of control laws G_o, which specifies for each

control law, the probability that it will be in effect. E.g., Let N=2 and T=2. Then G_o has 2^4 elements. The randomized decision rule corresponding to 'each station has each time slot green light with probability .5 and these events are independent' is given by the probability measure that assigns to each of the elements in G_o probability 2^{-4}.

(3:2.3) Theorem. Let G_r denote the space of randomized control laws and G_o, as before, the space of open loop control laws then

$$\min_{u \in G_r}\{C(u)\} = \min_{u \in G_o}\{C(u)\}$$

i.e., the minimum cost is achieved within the class of open loop laws.

Proof. Suppose we number the elements of G_o such that $G_o = \{u_1, u_2, \ldots, u_c\}$, where c is the number of elements in G_o. Then any randomized control law is given in an obvious way by a sequence of probabilities $(p_1, p_2, \ldots, p_c)$, such that $\sum_{i=1}^{c} p_i = 1$. The expected cost for such a randomized control law is $\sum_{i=1}^{c} p_i C(u_i)$. Now let $\hat{i} = \arg\{\min_i C(u_i)\}$, then the expected cost $C(u_{\hat{i}})$ for the open loop control law $u_{\hat{i}}$ is clearly not larger then the expected cost for that randomized control law. Thus

$$\min_{u \in G_r}\{C(u)\} \geq \min_{u \in G_o}\{C(u)\}$$

But from the fact that G_o is a subset of G_r we know that

$$\min_{u \in G_r}\{C(u)\} \leq \min_{u \in G_o}\{C(u)\}$$

This proves the theorem, which is well known in Bayesian decision theory (Cf. [12], Ch. 8).

(3:2.4) Model. The variable $x_i(t)$ which describes the state of the buffer of station i at time t takes only the values 0 and 1. Therefore the probability of the buffer being full is

$$\Pr\{x_i(t)=1\} \triangleq \hat{x}_i(t) = E\{x_i(t)\}$$

Under an open loop control law, given by a predetermined set of values of the control variables $u \triangleq \{u_i(t) \mid i=1,\ldots,N, t=1,\ldots,T\}$, the <u>evolution</u> of $\hat{x}_i(t)$ is deterministic and can be found from (3:1.5) by taking expectations on both sides of the equation resulting in

(3:2.5) $$\hat{x}_i(t+1) = p+(1-u_i(t))(1-p)\hat{x}_i(t)$$

Similarly the <u>expected cost</u> C(u) for an open loop control law represented by u can be found from (3:1.6) by taking expectations using the fact that for given u:

$$E\{x_i(t)x_j(t)\} = E\{x_i(t)\}E\{x_j(t)\} = \hat{x}_i(t)\hat{x}_j(t) \quad (i\neq j)$$

because $x_i(t)$ and $x_j(t)$ are independent random variables. This gives

(3:2.6) $$C(u)=\sum_{i=1}^{T}\{\sum_{i=1}^{N}(1+B)\hat{x}_i(t)(1-u_i(t))+$$

$$K[1-\prod_{i=1}^{N}(1-u_i(t)\hat{x}_i(t))-\sum_{i=1}^{N}u_i(t)\hat{x}_i(t)\prod_{j\neq i}^{N}(1-u_j(t)\hat{x}_j(t))]\}$$

(3:2.7) Problem. In the new model formulation above, which was based on the knowledge that the control law would be open loop we can interpret the vector $\underline{\hat{x}}(t) \triangleq (\hat{x}_1(t),\ldots,\hat{x}_N(t))'$ as the new (but now deterministic) state of the system. From this viewpoint the problem

$$\min_{u \in G_o} \{C(u)\}$$

with the cost C(u) given by (3:2.6) and the dynamics by (3:2.5) constitutes a dynamic programming problem.

(3:2.8) Remark. Although there are well-known solution techniques for dynamic programming problems, none of those will allow us to solve our problem for, say, more than 3 or 4 stations, because then the state space, which is the N dimensional hyper-cube $[0,1]^N$, becomes "too big". This limitation is known as "the curse of dimensionality".

The conclusion is that we need to restrict even further our solution space if we are to find the optimal decision rule. The next restriction serves this purpose.

(3:2.9) Restriction. Let $q_o(t)$ be the probability that at time t the channel is silent, that is: no packet leaves any station. Then we shall restrict ourselves to solutions for which $q_o(t)$ is constant over time.

This restriction seems not unreasonable because we want to spread out transmissions as evenly as possible in order to minimize the chance of collision for a given traffic intensity, which should result in a constant probability of no traffic. In fact we could conjecture that this is not a restriction but a property of the optimal solution. Either way, the real importance lies in the fact

that it enables us to prove theorem (3:2.21). This theorem is powerful through the simplicity of its conclusion. It states that the optimal control law has either green light for all stations all the time (All Together) or green light for each station in turn (Round Robin). Before we embark upon theorem (3:2.21) we need some definitions and lemmas.

(3:2.10) Definitions. A station is said to be of <u>age</u> b at the beginning of a time slot if the start of its last time slot with green light was b units of time ago (one time slot lasts one unit of time). If station i is of age b at time t then this implies that $\hat{x}_i(t)=1-(1-p)^b$. The <u>total age</u> at time t : $b_T(t)$ is the sum of the ages over all stations. The <u>total age sending</u> at t : $b_S(t)$ is defined as the sum of the ages over all stations that have $u_i(t)=1$ (green light).

(3:2.11) Lemma. Restriction (3:2.9) implies that the total age sending is constant and equals N.

<u>Proof.</u> Suppose that at time t there are n stations with green light having ages $b_1(t),\ldots,b_n(t)$. Then the probability of no packet leaving the system in the t-th time slot is equal to the product of the probabilities of the sending stations having an empty buffer:

$$q_o(t)=\prod_{j=1}^{n}(1-p)^{b_j(t)}=(1-p)^{b_S(t)}$$

Since $q_o(t)$ is constant, $b_S(t)$ must be constant as well: $b_S(t)=b_S$. The evolution of the total age of the system can be seen to be

$$b_T(t+1)=b_T(t)+N-b_S$$

The total age of the system is bounded from below by N and from above by N*d, where the latter bound comes from the assumption made in theorem (3:2.1). The only way that $b_T(t)$, t=1,...,T can stay within those bounds for large T, is for $b_S = N$, Q.E.D.

(3:2.12) Lemma. The cost function (3:2.6) can alternatively be written as

(3:2.13) $$C(u)=\sum_{t=1}^{T} c(t)$$

in which c(t) is given by

$$c(t)=(1+B)\sum_{j=1}^{n}[b_j-\frac{1-(1-p)^{b_j}}{p}]$$

$$+K[1-(1-p)^N-\sum_{j=1}^{n}(1-(1-p)^{b_j})(1-p)^{N-b_j}]$$

where it is assumed that at time t there are n stations having green light with ages $b_1,...,b_n$ respectively.

Proof. Both in (3:2.6) and in (3:2.13) the cost consists of two terms, the first one is the cost of having a packet and not sending: in short, the cost of waiting --this term has the factor (1+B). The second term is the cost of a single collision K, multiplied by the probability of collision: in short, the cost of collision. The difference between (3:2.6) and (3:2.13) is a matter of accounting: in (3:2.6) the cost is counted for each station in each time slot; in (3:2.13) we count in a time slot the cost only for the stations which have green light in that time slot, including

their cost of waiting during previous time slots that was not accounted for. A second difference is that in (3:2.13) $\hat{x}_i(t)$ is calculated explicitly from (3:2.5). So the cost of waiting for the j-th station which has green light while at age b_j is

$$(1+B)[(1-(1-p))+\ldots+(1-(1-p)^{b_j-1}]$$
$$=(1+B)[b_j-\frac{1-(1-p)^{b_j}}{p}]$$

The second term in (3:2.13) is easily recognized by noting that the probability that none of the stations with green light has a packet is

$$(1-p)^{b_S}=(1-p)^N$$

and the probability that the j-th station with green light is the only one which has a packet is

$$(1-(1-p)^{b_j})(1-p)^{N-b_j}$$

(3:2.14) Lemma. Among all age distributions $\{b_1,\ldots,b_n\}$ of the total age sending $b_S=N$ the only two that can possibly minimize the cost (3:2.13) are those with

$$b_1=b_2=\ldots.=b_N=1$$

or

$$b_1=N$$

Proof. Consider any other age distribution say

$$b_1,\ldots,b_n.$$

For it to be different from the two mentioned before we must have $1<n<N$ and for at least one j: $b_j>1$, say, without loss of generality, $b_1>1$. We shall construct two variants of the age distribution under consideration, of which always one

will give strictly lower cost, thereby showing that the age distribution that we considered can not be optimal.

Variant 1: instead of 1 station of age b_1 and 1 of age b_2 we have b_1+b_2 stations of age 1 :

$$1,1,\ldots,1,b_3,\ldots,b_n$$

Variant 2: instead of 1 station of age b_1 and one station of age b_2 we have 1 station of age b_1+b_2:

$$b_1+b_2,b_3,\ldots,b_n$$

First consider variant 1. Since a station of age 1 has no cost waiting we find that the decrease in the cost of waiting is:

$$(1+B)[b_1-\frac{1-(1-p)^{b_1}}{p}+b_2-\frac{1-(1-p)^{b_2}}{p}]$$

$$(3:2.15)\qquad =\frac{1+B}{p}[(1-p)^{b_1}+b_1p-1+(1-p)^{b_2}+b_2p-1]$$

Also from (3:2.13) we find that for variant 1 the increase in the cost of collision is

$$K[(1-(1-p)^{b_1})(1-p)^{N-b_1}$$
$$+(1-(1-p)^{b_2})(1-p)^{N-b_2}-(b_1+b_2)p(1-p)^{N-1}]$$

after some manipulation this is

$$=K(1-p)^{N-b_1-b_2}[(1-p)^{b_2}(1-(1-p)^{b_1}-b_1p(1-p)^{b_1-1})$$
$$+(1-p)^{b_1}(1-(1-p)^{b_2}-b_2p(1-p)^{b_2-1})]$$

Now use the following inequality (for $0<p<1$ and $a\geq 1$):

$$1-(1-p)^a-ap(1-p)^{a-1}\leq(1-p)^a+ap-1$$

which can be proven by showing that both sides are 0 for $p=0$ but the slope of the right-hand side is consistently greater on the interval (0,1). Then the increase in cost of collision is at most

$$K(1-p)^{N-b_1-b_2}[(1-p)^{b_2}((1-p)^{b_1}+b_1p-1)$$
$$+(1-p)^{b_1}((1-p)^{b_2}+b_2p-1)]$$

which is in turn strictly less then

(3:2.16) $K(1-p)^{N-b_1-b_2}[(1-p)^{b_1}+b_1p-1+(1-p)^{b_2}+b_2p-1]$.

By comparing (3:2.15) and (3:2.16) we see that there will be improvement if

(3:2.17) $$\frac{1+B}{p} \geq K(1-p)^{N-b_1-b_2}$$

Now consider variant 2. The decrease in the cost of collision is

$$K[(1-(1-p)^{b_1+b_2})(1-p)^{N-b_1-b_2}$$
$$-(1-(1-p)^{b_1})(1-p)^{N-b_1}-(1-(1-p)^{b_2})(1-p)^{N-b_2}]$$

(3:2.18) $$=K(1-p)^{N-b_1-b_2}[(1-(1-p)^{b_1})(1-(1-p)^{b_2})]$$

the increase in cost of waiting is

$$(1+B)[(1-(1-p)^{b_1})+\ldots+(1-(1-p)^{b_1+b_2-1}$$
$$-(1-(1-p)-\ldots-(1-(1-p)^{b_2-1})]$$

(3:2.19) $$= \frac{1+B}{p}[(1-(1-p)^{b_1})(1-(1-p)^{b_2})]$$

comparing (3:2.18) and (3:2.19) we see that there will be improvement if

(3:2.20) $$\frac{1+B}{p} < K(1-p)^{N-b_1-b_2}$$

It follows immediately from (3:2.17) and (3:2.20) that always at least one of the variants gives lower cost, which now proves the lemma.

(3:2.21) Theorem. The optimal control law for 'new' packets, depending on the value of p, is to have green light for all stations all the time:

$u_i(t)=1 \quad \forall i,t \quad$ if $p<p_o \qquad$ (All Together)

or to have green light for each station in turn:

$u_i(t)=1 \quad$ for $\quad t_{\text{mod } N}=i-1 \quad$ if $p\geq p_o$ (Round Robin)

where p_o is the unique solution on the interval (0,1) of

(3:2.22) $(1+B)[N-\frac{(1-(1-p)^N)}{p}]=K[1-(1-p)^N-Np(1-p)^{N-1}]$

Proof. From theorems (3:2.1) and (3:2.3) we know that the optimal control law is a predetermined pattern of 0's and 1's. From lemma (3:2.14) we know that we should have (if feasible) in each time slot either N stations of age 1 with green light or 1 station of age N. Using lemma (3:2.12), the cost per stage is then respectively

$$K[1-(1-p)^N-Np(1-p)^{N-1}]$$

or

$$(1+B)[N-\frac{1-(1-p)^N}{p}]$$

The theorem states that we should take whichever is the cheapest of those two, for given p, K, and B. This choice is feasible if we choose the proper initial conditions, namely at t=1 station i is of age 1 $\forall i$ if $p<p_o$, or at t=1 station i is of age N+1-i if $p\geq p_o$. There will be little objection to choosing initial conditions at convenience, if we note that the initial conditions are 'erased' as soon as every station has had green light once and that we are interested in solutions for large T.

(3:2.23) Remark. An approximate solution to (3:2.22) which will be a better approximation as the resulting p_o becomes smaller and N larger, is given by

(3:2.24) $$p_o \cong \frac{1+B}{K}$$

This value for p_o can be explained as follows: Consider a station that has a packet and suppose the stations operate in All Together mode; then the station will send the packet right away. It estimates the probability that another station has a packet to be Np, hence with probability Np there will be a collision of cost K (assuming that the other stations operate in the same mode and neglecting the probability of collisions with 3 or more packets involved). Therefore the expected cost of collision per station-with-packet is KNp/2. Now suppose that the stations operate in Round Robin mode, then the station with a packet has to wait on average N/2 time slots. Therefore the average cost of waiting is (1+B)N/2. So p_o is just at the crossover point where the "cost of collision becomes greater than the cost of waiting".

3:3 Optimal Control Law for Collided Packets.

In this section the problem of what to do after a collision is solved as a separate problem. This corresponds to assuming that as soon as the echo from a collision comes back, all stations are blocked for new packets until the collision is resolved in a sense to be made precise below. Imposing such a separation between new packets and collided packets enables us to solve the overall problem in a nice way, however it may introduce sub-optimality at the same time. Specifically, sub-optimality may arise from the build up of a backlog of new packets during the time the channel is used for retransmissions. When the system returns to All Together mode, the probability of a new collision is higher then was accounted for. For very small values of p, this sub-optimality will be negligible. For larger values of p, the system operates in Round Robin mode --therefore no retransmissions of collided packets take place and there is no sub-optimality. It is left for further research to solve the integrated problem for intermediate values of p. Possibly it can be shown that for these intermediate values the optimal mode is Round Robin to.

(3:3.1) Model. Suppose in time slot t_c (i.e. the time slot that started at $t=t_c$) the set of stations which were transmitting a packet $\{i_1, i_2, \ldots i_r\}$ has two or more

elements. Then in time slot t_c+d-1 the echo of a collision comes back to all stations. In response to this collision all stations refuse a possible arrival during time slot t_c+d and remain blocked until the collision is resolved, the stations that were involved in the collision place their collided packet in their buffer (it is assumed that all stations keep somewhere copies of transmitted packets until the transmission is known to be successful). If we 'reset' time such that $t=t_c+d$ becomes $t=1$ then we have the following <u>initial conditions</u>:

$$x_i(1)=1 \quad \text{if } i\in\{i_1,i_2,\ldots,i_r\}$$
$$=0 \quad \text{otherwise}$$

where $\{i_1,i_2,\ldots,i_r\}$ is the set of success-indices of a sequence of N Bernoulli trials with probability of success p. Note that the initial conditions form a random vector of which each station only knows its own component. Since there are no arrivals the <u>dynamics</u> are simply given by

(3:3.2) $$x_i(t+1)=(1-u_i(t))x_i(t)$$

and the <u>cost</u>, for given controls $u_i(t)$ $i=1,\ldots,N$, $t=1,\ldots,T_2$, similar to (3:1.6) is

(3:3.3) $$K=E\{\sum_{t=1}^{T_2}\{\sum_{i=1}^{N}x_i(t)(1-u_i(t))+NB$$
$$+K_2[1-\prod_{i=1}^{N}(1-x_i(t)u_i(t))-\sum_{i=1}^{N}x_i(t)u_i(t)\prod_{j\neq i}^{N}(1-x_i(t)u_i(t))]\}\}$$

where K_2 is the cost of creating a second collision. Clearly as soon as every station has had $u_i(t_i)=1$ once for some $t_i<t$ then $x_i(t)=0$ $\forall i$ and the collision is resolved

(there may be a second collision but at this point we assume that that is taken care of at cost K_2). Therefore T_2, the time needed for a second transmission, is defined by

$$T_2 = \max\{t \mid x_i(t)=1 \text{ for some } i\}$$

The problem, of course, is to choose the controls $u_i(t)$ such as to minimize the cost. The approach to the problem for collided packets parallels the approach in the preceding sections. First we note that the problem is not fully defined until we specify what the information is on which controls can be based. The following theorem shows that the little information available, is of no use.

(3:3.4) Theorem. The controls $u_i(t)$ that minimize the cost are predetermined (open loop): they will be independent of the information that each station has about the random vector which determines the initial conditions.

Proof. Besides implicit information such as the probability distribution of the initial condition, the only information available to station i is $x_i(1)$. However if $x_i(1)=0$ then the next states (of station i) as well as the cost are independent of the controls $u_i(t)$, $t=1,\ldots,T_2$. Therefore the controls can be taken the same as when $x_i(1)=1$, which implies that $u_i(t)$, $t=1,\ldots,T_2$ will be independent of station's involvement in the collision.

(3:3.5) Remark. We shall not consider randomized controls. They are excluded here too for very much the same reasons as given in theorem (3:2.3).

In model (3:3.1) there is still one point that has not been fully described, namely the cost for a second collision K_2. It was assumed that K_2 was a given quantity, but to determine K_2 we would have to solve a separate problem to find a parameter K_3: the cost for a third collision. This could lead to a long chain $K, K_2, K_3, \ldots$. The following restriction is introduced to break the chain right after K_2.

(3:3.6) Restriction. The controls must be chosen such that for any packet there will be no third collision.
The main motivation is that this restriction enables us to find, without too much complexity, an expression for the cost of collision K. But there are other benefits from this restriction, namely that it reduces the variance in delay and the maximum delay. Of course by doing more work one could have lesser restrictions of no fourth collision, no fifth collision, and so on up to no N-th collision (beyond that, the restriction is meaningless), however this does not seem worth the effort.

(3:3.7) Problem. From the dynamics (3:3.2) it is clear that for the second transmission each station needs only one time

slot with green light. From theorem (3:3.4) it follows that the assignment of a station to one of the time slots $1,2,\ldots,T_2$ can be predetermined and from the 'equality' of all stations it follows that only the number of stations assigned to a time slot matters. Restriction (3:3.6) dictates that whenever a second collision occurs among a group of stations which were assigned to one time slot then for the third transmission that group must do 'Round Robin'. So at this point the problem can be formulated as: <u>find the grouping</u>

$$m(1),m(2),\ldots,m(T_2)$$
$$\text{such that } \sum_{t=1}^{T_2} m(t)=N$$

<u>that minimizes the cost</u>

$$K=\sum_{t=1}^{T_2} k(m(t),t)$$

where $m(t)$ is the number of stations with green light during time slot t (of the second transmission) and K is the total cost, measured in units of delay, incurred by the system as a result of a collision in a first transmission. K can be calculated per group, with the cost of a group $k(m(t),t)$ as given in the next lemma.

(3:3.8) Lemma. Suppose a collision has occurred and let a group of m stations be assigned to the t-th time slot of the second transmissions. Then the <u>cost of collision</u> that can be <u>attributed to that group</u> is (for $m>0$)

$$k(m,t)=(d+t-1)\frac{mp(1-(1-p)^{N-1})}{1-(1-p)^N-Np(1-p)^{N-1}}+NB$$

$$+(d+\frac{m-1}{2})\frac{mp(1-(1-p)^{m-1})}{1-(1-p)^N-Np(1-p)^{N-1}}$$
$$+mNB\frac{1-(1-p)^m-mp(1-p)^{m-1}}{1-(1-p)^N-Np(1-p)^{N-1}}$$

Proof. According to the model (3:3.1) it is determined by a sequence of N Bernoulli trials which station has a packet (success) and which has not. The assumption that a collision occurred means that all probabilities are conditioned on 'at least two successes', abbreviated as 'coll.'. Define n as the number of stations, within the group of m stations. That have a packet, then the collision cost for the group can be written as

$$k(m,t)=(d+t-1)E\{n|\text{coll.}\}+NB$$
$$+(d+\frac{m-1}{2})E\{n|n\geq 2\}Pr\{n\geq 2|\text{coll.}\}$$
$$+mNB\ Pr\{n\geq 2|\text{coll.}\}$$

and explained, term by term, as follows. The first term gives the expected amount of waiting in the group from the first (unsuccessful) transmission to the second transmission for that group. During second transmissions all N stations are blocked for new packets, this adds a term NB per group. With probability $Pr\{n\geq 2|\text{coll.}\}$ the group has two or more stations with a packet in which case the second transmission is unsuccessful. Then there is more waiting: per packet this is one round trip time d plus waiting for its turn when doing Round Robin among the group members (on average $\frac{m-1}{2}$) and there is more blocking: during m time slots of the

third transmissions all N stations are blocked. By elementary probability theory we find

$$E\{n|\text{coll.}\} = \sum_{j=1}^{m} jPr\{n=j|\text{coll.}\}$$

$$= Pr\{n=1|\text{coll.}\} + \sum_{j=2}^{m} j\frac{Pr\{n=j\}}{Pr\{\text{coll.}\}}$$

$$= \frac{Pr\{n=1\&\text{coll.}\}}{Pr\{\text{coll.}\}} + \sum_{j=1}^{m}\frac{jPr\{n=j\}}{Pr\{\text{coll.}\}} - \frac{Pr\{n=1\}}{Pr\{\text{coll.}\}}$$

$$= \frac{mp(1-p)^{m-1}(1-(1-p)^{N-m})+mp-mp(1-p)^{m-1}}{Pr\{\text{coll.}\}}$$

$$= mp\frac{1-(1-p)^{N-1}}{1-(1-p)^{N}+Np(1-p)^{N-1}}$$

and

$$E\{n|n\geq 2\} = \sum_{j=2}^{m} j\frac{Pr\{n=j\}}{Pr\{n\geq 2\}}$$

$$= \sum_{j=1}^{m}\frac{jPr\{n=j\}}{Pr\{n\geq 2\}} - \frac{Pr\{n-1\}}{Pr\{n\geq 2\}}$$

$$= \frac{mp(1-(1-p)^{m-1})}{1-(1-p)^{m}-mp(1-p)^{m-1}}$$

and

$$Pr\{n\geq 2|\text{coll.}\} = \frac{Pr\{n\geq 2\}}{Pr\{\text{coll.}\}}$$

$$= \frac{1-(1-p)^{m}-mp(1-p)^{m-1}}{1-(1-p)^{N}-Np(1-p)^{N-1}}$$

(3:3.9) Remark. In the preceding lemma it was implicitly assumed that there is <u>no interference between retransmissions</u>. E.g., in the (unlikely) case that a collision of 4 or more new packets occurred and that in each of two groups (say, the groups with m(1) and m(2) members respectively) there are 2 or more stations that have a packet. Then we neglect the interference which arises in the 2 overlapping intervals of third transmissions:

$[t_c+2d, t_c+2d+m(1)-1]$ and $[t_c+2d+1, t_c+2d+m(2)]$.

(3:3.10) Algorithm. The solution to problem (3:3.7) can now easily be given in an algorithmic form with the approach of dynamic programming [13]. The algorithm explains itself, with the following definition in mind. Define J(n,t) to be the cost when n stations are assigned optimally to the first t time slots of second transmissions. (Note that the meaning of the symbol n does not correspond to its previous meaning.) Further, define $k(0,t)=0$ $\forall$,t.

```
========initialize========
for n=1 to N
          J(n,1)=k(n,1)
next n
for all t
          M(N,t)≠0
next t
t=1
========recursion=========
for t=t+1 until M(N,t)=0
          for n=1 to N
                    J(n,t)=min_m{k(m,t)+J(n-m,t-1)}
                    M(n,t)=arg{min_m{k(m,t)+J(n-m,t-1)}}
          next n
next t
========results===========
T2=t-1
K=J(N,T2)
m=N
for t=T2 to 1 step -1
          m(t)=M(m,t)
          m=m-m(t)
next t
```

(3:3.11) Conclusions. In the preceding sections we looked at the problem of access control to a packet switched satellite

broadcast channel in the framework of non-classical control theory. This viewpoint, with the simple model presented, enabled us to derive a nice solution that differs interestingly from other approaches to the same problem [14], [15]. In particular it is shown, by a result of Bayesian decision theory, that randomized decisions will not be necessary. Further we found that, under appropriate assumptions, for 'new' packets only two modes of operation of the channel could be optimal, namely: 'All Together' (every station sends whenever it has a packet) or 'Round Robin' (every station sends whenever it is its turn and it has a packet), and an equation for the crossover arrival probability p_o at which the channel would switch from All Together ($p<p_o$) to Round Robin ($p\geq p_o$) is given. For collided packets it was found that it is optimal to have predetermined assignments of groups of stations to time slots for retransmissions, and a dynamic programming algorithm for determining the optimal grouping is given.

Of course to get these results a number of assumptions had to be made. Some of the most restrictive assumptions are: a.) all stations have an equal arrival probability p, b.) The arrival probabilities stay constant over time, c.) The stations have only a single buffer for packets that are ready to be transmitted. Nevertheless we believe the results of this chapter to be useful, also in more general

situations that do not comply with the restrictions mentioned. It is currently being investigated as to how (maybe in a heuristic way) the results of this chapter can be applied to more general situations. Specifically we have the following in mind: a.) When stations have different arrival rates then they could be divided in two classes; one All Together class for stations with small arrival probability, and one Round Robin class for stations with high arrival probability, with time slots divided between the two classes. b.) When arrival probabilities vary over time, it will be reasonable to assume that the time between variations of the arrival probabilities is much longer then one round trip time to the satellite. In this case the stations could report periodically what their arrival probability is and on the basis of those reports the stations could agree on what presently the proper mode of operation will be. c.) When in each station, besides the single buffer, there are auxiliary buffers in which packets that would otherwise be refused are placed, then additional delay is incurred by packets in the auxiliary buffer. We purposely carried the factor B (cost of the single buffer being blocked) through all of the analyses, because in B one could summarize all the cost of waiting that is incurred in the auxiliary buffer.

Other points for further investigation are the performance and stability of the channel.

CHAPTER 4: INFORMATION POLICIES FOR ROUTING IN A COMPUTER NETWORK

4:1 Introduction and Definitions.

Packet switched computer networks have already proven to be a useful combination of computer systems and communication facilities, and their role in electronic data processing will become more and more important [16], [17]. Problems in the technical operation of a computer network range from of reliable transmission over the communication line (modulation, demodulation, hardware error correction) to the communication between computers that use the communication network (host-to-host communication protocol). At an intermediate level lies the problem of how to route a packet of digital information with a given destination through the network. The decisions as to what route will be taken by the packet at a node in the network can be made by the communication computer at that node. This is known as distributed routing [18]. The nodes are geographically separated and the information which is exchanged between the communication computers for the purpose of making better routing decisions travels on the same network that is to be controlled. These facts make the routing problem fit within the framework of non-classical control theory (i.e. more than one decision maker). The routing problem has already

been considered within the framework of control theory by a number of authors (e.g. [19] or [20]). The approach taken in this chapter is somewhat different in the sense that we will concentrate on the underlying problem of what is the "best" information on which to base the decisions. With regard to the question of what are the "best" decisions we shall see later in this section that --given a suitable objective-- there is an obvious answer. First we shall specify the routing problem more precisely in the following paragraphs.

(4:1.1) Definitions. A network is a set of nodes

$$\tilde{N}=\{1,2,\ldots,N\}$$

together with a set of links between ordered pairs of nodes

$$A=\{(i_1,j_1),\ldots,(i_L,j_L)\mid i_n,j_n\in\tilde{N}\}$$

The node i_n is called the begin point of the link (i_n,j_n), and correspondingly node j_n is the end point of that link. At each node i there is a source that generates packets with destinations $j\in\tilde{N}$ and there is a sink that absorbs all packets with destination i. Packets can travel along links. The first part of this definition is the usual definition of a directed graph. We assume that $(j,i)\in A$, whenever $(i,j)\in A$, that is the lines are full duplex. This assumption is not essential for the subsequent analysis but is made for simplicity.

A routing decision made by node i for a packet with destination $k \neq i$ is an assignment of packets to one of the links that have i as begin point. The routing decision will depend on the state of the network. A criterion for deciding which decision is "best" is the time needed for the packet to travel from source to destination. The time needed is obviously the sum of the delays on each of the "hops" lying on the path of the the packet (the packet flows through one hop when it flows through one node and one link). The state of the network is the vector D^1 of all 1-hop delays in the network (one element for each link). The one-hop delay from node i to node j, d^1_{ij}, is the sum of the following terms:

(4:1.2) $\tilde{d}^1_{ij}$ = processing time for the packet at node i

$$+ \frac{\text{physical distance between i and j}}{\text{speed of light along the path}}$$

$$+ \frac{\text{packet length}}{\text{channel capacity}}$$

$$+ \frac{\Sigma \text{ packet lengths of predecessors in queue}_{ij}}{\text{channel capacity}}$$

In the definition of $\tilde{d}^1_{ij}$ it is assumed that the packet always gets to node j. In practice this is not true, sometimes the packet is lost due to transmission errors or cannot be received. One scheme to deal with such events is to acknowledge each well received packet. If no acknowledgement comes back, the packet has to be retransmitted (this is the ARPA Network scheme). It is not

hard to include the expected delay due to possible retransmissions. Let p be the probability that a retransmission is necessary and let r be the maximum time that a node waits for an acknowledgement from a neighbor node. The expected number of retransmissions is

$$\sum_{k=1}^{\infty} kp^k(1-p) = \frac{p}{1-p},$$

and this adds a term

$$(\tilde{d}^1_{ij}+r)\frac{p}{1-p}$$

to the delay, assuming that r starts after $\tilde{d}^1_{ij}$ is elapsed and that the retransmission packet experiences the same delay, $\tilde{d}^1_{ij}$, as a new packet. From now on we shall include the correction due to possible retransmissions in the one-hop delay. Thus the <u>one-hop delay</u> is redefined as

(4:1.3) $$d^1_{ij} = \tilde{d}^1_{ij}+(\tilde{d}^1_{ij}+r_{ij})\frac{p}{1-p} = (\frac{1}{1-p})\tilde{d}^1_{ij}+(\frac{p}{1-p})r_{ij}$$

where $\tilde{d}^1_{ij}$ is as given in equation (4:1.2), p is the probability that a retransmission is necessary (may depend on ij) and r_{ij} is the time waited before retransmitting.

(4:1.4) Remarks. In the definition of the one-hop delay we considered the output queues of packets that were waiting at node i to go on the link to i's neighbor j, but not explicitly the <u>input queue</u> at node i of packets that are waiting for a routing decision to be made and for other processing required for newly arrived packets. This queue can be modeled in the same way as the output queues will be

modeled, without any additional difficulty. The delay in the input queue can be added to the delays of each of the output queues. For the sake of simplicity this term will be left out of our subsequent analysis.

The one-hop delay as given in (4:1.2)-(4:1.3) contains a number of terms which depend on the particular link (i,j) and may vary with time or with every packet. These are r_{ij}, p_{ij}, packet length β, channel capacity c_{ij} and processing time t_{ij}. Again for ease of exposition, we restrict ourselves for most of this paper to the case where the only variable is q_{ij}, the number of seconds of queueing delay in $queue_{ij}$, which implies fixed packet length β and constant r_{ij}, p_{ij}, c_{ij} and t_{ij} for each link. This means that d^1_{ij} can be thought of as

(4:1.5) $$d^1_{ij} = (k_{ij}+q_{ij})k'_{ij}$$

where k_{ij} and k'_{ij} are constants. However in remark (4:2.5) we give formulas to include variable packet length when the distribution over the different possible packet lengths is given. And in remark (4:2.15) we indicate how a node can update from time to time its estimated values for the distribution of packet length, r_{ij}, p_{ij}, c_{ij}, and possibly t_{ij} under the assumption that they change at a much slower rate than q_{ij}.

(4:1.6) Definitions. For conceptual and notational convenience we can arrange the state vector of one-hop

delays into a matrix of one-hop delays, which we denote by the same symbol D^1, in which the element (i,j) gives the one-hop delay from i to j. The diagonal elements of this matrix are defined to be 0 and the pairs $(i,j) \notin A$ have corresponding entry ∞.

The min-sum C of two matrices A and B of compatible dimensions, denoted by $A \hat{} B$ is given by

$$(C)_{ik} = \min_j \{(A)_{ij} + (B)_{jk}\}$$

It can be seen directly from this definition that when B is a matrix of minimum (p-1) - hop delays and A is a matrix of 1-hop delays then C will be the matrix of minimum p-hop delays.

The matrix of minimum delays is given by $D = D^M$ where D^M is defined recursively through $D^p = D^1 \hat{} D^{p-1}$ and M is the maximum hop distance in the network. Note that although D^M contains M-hop delays only, because of the zeros on the diagonal of D^1, less-than-M-hop delays are essentially included.

So when a node knows the state vector D^1 precisely it can compute the matrix of minimum delays D. However the matrix D tells a node only what the minimum delay is, not along which path the minimum delay is attained. A node needs another matrix on which to base its decisions: the node delay table.

The node delay table of node i, denoted D_i, has as element (j,k) the minimum delay from i to k via j where j is a neighbor of i (i.e., $(i,j) \in A$). The node delay table can be computed from D^1 and D namely $(D_i)_{jk} = (D^1)_{ij} + (D)_{jk}$.

The definitions above lead up to an obvious decision rule, which is given in the next assertion, provided we assume that the node delay table is accurate and that the objective is to minimize delay for an individual packet.

(4:1.7) Assertion. The optimal decision for minimizing delay for an individual packet, under the assumption of perfect information, is to send the packet along the minimum delay path as given by the node delay table. We call this the minimum delay decision rule.

(4:1.8) Remark. The assumption of perfect information is rather strong. It requires that the decision maker, say node i, knows the state of the network D^1 precisely not only at the time that the decision is made but also ahead of time for the duration of the flight of the packet to its destination. This means that we must either assume that everything in the network remains constant, except for node i's decision, or that node i can anticipate all the changes in the delays, which will be experienced by the packet, due to decisions of other nodes. Obviously neither assumption is ever going to be satisfied. In the next section this strong assumption will be relaxed with our choice to let the the information be not the actual values of the one-hop delays but the expected values of the delays. Basing the minimum

delay decision rule on the latter information changes the objective from minimizing delay to minimizing expected delay. Nevertheless, it is worthwhile to consider first the assumption of perfect information because it suggests the simple minimum delay decision rule, which we shall assume to be in effect for the rest of our analysis. Our primary concern is to determine the information on which to base the decisions.

4:2 Choosing an Information Policy.

In designing the information policy, there is a sequence of decisions that must be made about the nature of the information to be collected, and the structure of the communication policy for distributing the information. We shall describe each decision in turn, indicating which choice we make in the design presented here.

As pointed out in remark (4:1.8) the assumption of perfect information is rather strong and cannot be satisfied in practice because the delays at the nodes change much faster than information can travel through the network. (Note that this is fundamental to packet switched networks since the control information flows on the same channel as the data and should take only a fraction of the channel. This is not the case in other applications such as vehicle traffic.) A particular delay will here be modeled as a constant plus the queueing delay in a queue with Poisson arrivals of intensity $\lambda(t)$. The parameter of the Poisson process, $\lambda(t)$, will be modeled as a jump process, where the time between successive jumps is comparable with the delays in the network. The parameters of the jump process will be considered constant. On the basis of this cascade of stochastic processes we can define different classes of information policies. The choice among these classes represents our first major design choice.

(4:2.1) Definition. An information policy in our context is said to be of Class 0 if the information that the nodes exchange are the actual delays, either the actual delay itself or the queue length from which the the corresponding one-hop delay can be determined according to (4:1.2)-(4:1.3). An information policy is of Class I if the information exchanged is the parameters of the Poisson processes or the corresponding expected delays. An information policy is said to be of Class II if the information is the parameters of the jump processes or the corresponding long term expected delay.

(4:2.2) Choice. (Information class.) In section 4:3 we shall discuss why we believe Class I to be the best choice and we continue this section under that assumption.

The next design choice to make is how to model the stochastic process at a node which is responsible for the delay. As indicated in remark (4:1.4) we shall pursue in depth the case in which variations in delay are caused only by variations in the number of packets in the corresponding queue. Extensions to the case of variable packet length as given in remark (4:2.5) and to the case of variable probability of retransmission, p, retransmission time, r, and other terms contributing to the delay as given in remark (4:2.15) are straightforward and do not alter the

information policy significantly. This justifies choosing a queueing model for the delay. This model has the additional advantage that it has as a natural description the Poisson model, which requires estimating only a single parameter, from which delay can be predicted.

(4:2.3) Choice. (Queueing model.) At each node there are several queues - one for each outgoing link and one input queue (cf. remark (4:1.4)). These queues can be modeled individually as a set of independent queues or as a much more complex system of dependent queues. We choose the former approach for simplicity. The arrivals in an individual queue are assumed to form a Poisson process with intensity $\lambda(t)$. This choice is made because: a) packets arriving in the queue come from a (possibly large) number of independent sources, b) the collection of arrival patterns that can be modeled as a Poisson process with time varying parameter is very rich. The assumption of fixed packet length implies a deterministic service time for each packet -- say β seconds (i.e., we measure packet length in seconds needed to go on the line, rather than bits). The model we have now is that of a M/D/1 (Poisson arrivals/ deterministic service time/ 1 server) queue. Let $Q(t)$ denote the queueing delay in seconds at time t. For $\lambda(t)$ constant the steady state expected queueing delay is given by

(4:2.4)
$$\lim_{t\to\infty} E\{Q(t) \mid \lambda(t)=\lambda\} = \frac{\lambda\beta^2}{2(1-\lambda\beta)} \qquad \text{for } \lambda\beta<1$$

AD-A039 222 HARVARD UNIV CAMBRIDGE MASS DIV OF ENGINEERING AND --ETC F/G 17/2
DECENTRALIZED CONTROL IN COMPUTER COMMUNICATION.(U)
APR 77 F C SCHOUTE N00014-75-C-0648
UNCLASSIFIED TR-667 NL

2 OF 2
AD A039222



END

DATE FILMED
5-77

which is a special case of the next formula (4:2.6). Equation (4:2.4) enables us to translate the parameter of the Poisson process into expected delay, which gives us an obvious way of applying the minimum delay decision rule on Class I information. Note that since (4:2.4) gives the steady state value it is "predictive", which may compensate for information delays.

(4:2.5) Remark. Suppose that the packet length is variable and takes the value l_k with probability p_k ($k=1,2,\ldots,K$), independent of any previous packet length, where length is measured in time needed to put a packet on a outgoing line. Let $\lambda(t)$ again be the intensity of the Poisson process of arrivals. Then the expected steady state queueing delay is given by:

(4:2.6)
$$\lim_{t\to\infty} E\{Q(t) \mid \lambda(t)=\lambda\} = \frac{\lambda\beta_2}{2(1-\lambda\beta)} \qquad \text{for } \lambda\beta<1$$

where
$$\beta = \sum_{k=1}^{K} p_k l_k \quad \text{(average packet length)}$$

and
$$\beta_2 = \sum_{k=1}^{K} p_k l_k^2 \quad \text{(second moment)}$$

This is a standard result for the M/G/1 queue (Poisson arrivals/ General distribution of service time/ 1 server) and can be found in e.g. [21], Ch.4.

(4:2.7) Remark. It is interesting to compare (4:2.4) and (4:2.6) with the expected steady state delay in a queue with Poisson arrivals and negative exponential service time. This

is the M/M/1 queue, which is used by most authors who apply queueing theory to computer networks. For a M/M/1 queue we have:

(4:2.8) $$\lim_{t\to\infty} E\{Q(t)|\ \lambda(t)=\lambda\} = \frac{\lambda 2\beta^2}{2(1-\lambda\beta)} \qquad \text{for } \lambda\beta<1$$

Formulas (4:2.4), (4:2.6) and (4:2.8) give successively more conservative figures for the expected queueing delay which can be seen from

$$\lambda\beta^2 \le \lambda\beta_2 \le \lambda 2\beta^2 \qquad \text{for } 0\le\lambda\beta<1$$

The next step in the cascade of stochastic processes, which generates the sequence of Class 0, Class I, ... information policies, is to model the time varying parameter of the Poisson process as a stochastic process.

(4:2.9) Choice. (Model for the parameter of the Poisson process.) The parameter of the Poisson process is assumed to be a jump process which can take only a finite number of values in the state space $\{\lambda_1,\lambda_2,..\ ,\lambda_L\}$. Jumps within the state space occur only at equidistant points in time and form a Markov chain with transition probabilities given by the stochastic matrix Π. We choose this model for the following reasons: a) only a finite number of states are needed because we cannot distinguish between λ and $\lambda+\theta$ with statistical confidence on the basis of a limited number of observations for a whole range of θ's, b) the assumption of changes in the state only at equidistant points in time is

made to simplify the estimation of the state of the jump process; this limitation can be made arbitrarily small by choosing the distance between the time-points sufficiently small, c) the Markov chain assumption is made because of convenience; it corresponds to the assumption that the probability of jumping to a given next state is a function only of the present state.

(4:2.10) Remark. Choice (4:2.9) still leaves two parameters for which a value has to be chosen, namely the number of elements in the state space of the jump process, L, and the time interval between possible jumps, h. The choice of L and h is a tradeoff between high precision of the model (large L, small h) on the one hand and low line bandwidth and node bandwidth requirements (small L, large h) on the other hand.

The final decision in the design process of the random variable that represents delay, is how to model the parameters of the jump process.

(4:2.11) Choice. (Model for the parameters of the jump process.) The matrix Π of transition probabilities of the jump process is considered constant. That is, it changes so slowly that some centralized off-line procedure may be implemented to adjust Π. The motivation for this choice is that it probably takes a couple of hours of data collection

before one can decide with statistical confidence that the parameters of the jump process are not what they were thought to be. In appendix 4:A a simple numerical example is considered to support this statement.

(4:2.12) Assertion. (Estimating the state of the jump process.) Let the vector $p(k^-)$ give the probability distribution over the states of the of the jump process $\{\lambda_1,\ldots,\lambda_L\}$ just before a possible jump at $t=t_k$ takes place. The matrix of (constant) transition probabilities Π has as its $(l,m)^{th}$ element the probability of jumping to λ_l given that the present state is λ_m. Then by elementary probability theory the probability distribution over $\{\lambda_1,\ldots,\lambda_L\}$ just after t_k is given by $p(k^+) = \Pi\, p(k^-)$. Suppose that the number of arrivals in the queue in the time interval t_k to $t_{k+1}=t_k+h$ is n, then by Bayes Theorem the probability of being in state λ_l just before $t=t_{k+1}$, given the n arrivals, is:

$$\Pr\{\text{state at } t_{k+1}^- = \lambda_l \mid n \text{ arrivals}\} = \frac{\Pr\{n \text{ arrivals} \mid \lambda_l\}p_l(k^+)}{\Pr\{n \text{ arrivals}\}}$$

where

$$\Pr\{n \text{ arrivals} \mid \lambda_m\} = \frac{(h\lambda_m)^n}{n!}e^{-h\lambda_m}$$

and

$$\Pr\{n \text{ arrivals}\} = \sum_{m=1}^{L}\Pr\{n \text{ arrivals} \mid \lambda_m\}$$

Let $\hat{l}$ be such that $p_l(k+1^-)\leq p_{\hat{l}}(k+1^-)$ for $l=1,\ldots,L$, then the a-posteriori most likely estimate of the state of the jump process just before $t=t_{k+1}$ is $\lambda_{\hat{l}}$.

Up to this point we have modeled three classes of parameters that describe the state of the network; from Class 0 that gives the instantaneous state and changes vary rapidly, to Class II that gives long term statistics about the state of the network and can be considered constant. We also decided that the information exchanged between nodes should be of Class I, for reasons to be explained in section 4:3. Still more choices have to be made, namely within Class I we must decide what sort of data to send between the nodes, whether or not to perform the computation of minimum delays in a distributed way (this affects the data that is exchanged between nodes) and when to send the data.

(4:2.13) Choice. (Sending Poisson parameter vs. expected delay values.) First we must decide what data to send. Two possibilities are the Poisson parameter from which can be derived the expected queueing delay, or the actual expected value of the delay itself. The former has the advantage that it is a very concise representation, and transmitting it is economical of line bandwidth. On the other hand, it may require a significant amount of node processing and node storage to convert this parameter into expected delay. The node receiving the parameter $\lambda(ij)$ must perform the following calculation to obtain the expected delay:

(4:2.14) $$d^1_{ij} = (k_{ij} + \frac{\lambda_{ij}\beta^2}{2(1-\lambda_{ij}\beta)})k^1_{ij}$$

where k_{ij} and k^1_{ij} are "constants" that depend on t_{ij}, c_{ij}, p_{ij}, r_{ij} (cf. (4:1.5)). Under our assumptions of constant β, t_{ij}, c_{ij}, p_{ij}, r_{ij}, all the nodes must duplicate the performance of the calculation (4:2.14) and have tables of k_{ij} and k^1_{ij}. If the values of t_{ij}, c_{ij}, p_{ij}, r_{ij} are changing with time, additional communication is needed to update k_{ij} and k^1_{ij}. Therefore, we choose to design the information policy to send the actual value of the expected delay in favor of the Poisson parameter. We note in passing that it may still be possible to use a very compact representation of the delay in order to minimize the use of bandwidth, at the expense of increased node bandwidth requirements. For instance, a floating point format or other simplified version of the same concept can be used since there are relatively few significant bits of information about the value of the expected delay, whereas there may be a large range of values.

(4:2.15) Remark. The choice of sending expected delay values has the additional advantage that it is now simple to accommodate the case where t_{ij}, c_{ij}, p_{ij}, r_{ij} are changing in time if we assume that they change at a rate slower than or equal to the rate at which the Poisson parameter λ_{ij} is changing. In addition to the estimation of λ_{ij} according to

(4:2.13) node i can perform (probably less sophisticated) procedures for estimating t_{ij}, c_{ij}, p_{ij}, r_{ij} and use those values for computing d^1_{ij}. And of course variable packet length can also be included in the computation of d^1_{ij} (cf. remark (4:2.5)).

(4:2.16) Choice. (Replicated vs. distributed computation of minimum delays.) An important design decision is whether all the nodes perform independent, replicated computations with the same input data, or whether they cooperate in distributing the computation among the nodes, and share the output data. In the replicated computation, after all the nodes have exchanged their values of the expected delay d^1_{ij}, each node performs its computation of the matrix of minimum expected delays D, as indicated in definition (4:1.6) and from that finds the neighbor which lies on the quickest path. Some simple calculations show that this approach requires unrealistically high node bandwidth. Each node receiving a new one-hop distance must perform a new quickest path computation which is quadratic in the number of nodes, and each node receives such updates from all other nodes, which makes the process cubic. The computation of delay vectors from one-hop delay in a replicative fashion is therefore discarded in favor of a distributed approach.

We now present one way of distributing the computation of the matrix D among the nodes, which we shall use in our

comparative analysis in section 4:3. Let node i compute row i of $D^p = D^1 \hat{} D^{p-1}$, and send the result to its neighbors; this constitutes one iteration. Note that D^1 is a sparse matrix: row i has only finite elements in columns $j_1, j_2, \ldots, j_n$ and i, where $j_1, j_2, \ldots, j_n$ are the neighbors of i (typically n is 5 or less). This means that in computing row i of D^p node i needs to have only rows $j_1, j_2, \ldots, j_n$ of D^{p-1}. This is exactly what node i received from its neighbors in the previous iteration. The total number of iterations needed is equal to the maximum hop distance M in the network. However, partial results in the computation of D (these are D^p, $p<M$) can often be used for routing decisions. For example, after only three iterations, node i has available to it the minimum 3-hop delays from its neighbors to all destinations (some of the values may be "infinite" meaning that that destination is not reachable in 3 hops from the neighbor). Thus, the node can decide for all destinations that are reachable in 4 hops or less (these may account for a large portion of the traffic) via which neighbor the minimum 4-hop or less delay is attained. This will lead to a sub-optimality only in the case that there is a path of more than 4 hops to some node which has lower delay than the 4-hop or less path. It is assumed that the nodes determine their estimates of expected 1-hop delay to their neighbors just before a new sequence of M iterations is started and do

not alter this value until the next sequence of M iterations. This is to help prevent the situation in which node i thinks that the quickest path to destination k is via node j, and node j thinks that the the quickest path to destination k is via node i.

With this model of the distributed computation we have also determined the form of the data that is exchanged between the nodes: row-vectors of minimum p-hop expected delays of one node to all destination.

(4:2.17) Remark. The ARPA Network scheme of distributed computation differs from the scheme given in (4:2.16). In the ARPA Network scheme, node i send to its neighbors a row-vector of minimum delays from i to all destinations and gets the same information from its neighbors. Node i then recomputes its new row vector of minimum delay from its present values of one hop delays to its neighbors and the information just received. This scheme also converges to the actual minimum delays when all the one-hop delays remain constant, but in an undetermined number of iterations. In practice, of course, the nodes must update their estimates of the one-hop delays from time to time, sometimes within a sequence of update iterations. This can in some cases lead to the ping-ponging alluded to in (4:2.16), where node i sends traffic for k via j and j sends traffic for k via i. There are ad hoc solutions to prevent this ping-ponging but

then it is no longer simple to analyse how up-to-date the routing information is. We have therefore chosen the scheme in (4:2.16) for further analysis in section 4:3.

The final choice we have to make determines the frequency with which the nodes exchange control data.

(4:2.18) Choice. (When to exchange data.) A sequence of iterations as described in (4:2.16) can be started periodically without regard to changes in the one-hop delays or solely in response to such changes, or a combination of both. The last approach has the merits of both; when information changes, the local node can start a new sequence of iterations. Some aspects of periodic updating must be maintained to ensure that the process does not occur with too low a frequency (bad for reliability) or too high a frequency (bad for line bandwidth). In fact, in a large network, it is not possible to have the ideal of event triggered routing information flooding the network, since too many new events occur and the paths are so long. Then this approach degenerates to periodic updating.

(4:2.19) Summary. In this section we have presented a set of design choices leading to a fully specified routing algorithm of Class I. As shown in figure 4:2.a these choices can be seen as defining a particular algorithm which can then be analysed and compared against other methods. For the

purposes of a comparative analysis in section 4:3, we define an analogous set of decisions for Class 0 and Class II, as illustrated in the figure. It is these three algorithms, the results of choices shown, which are analysed in section 4:3.

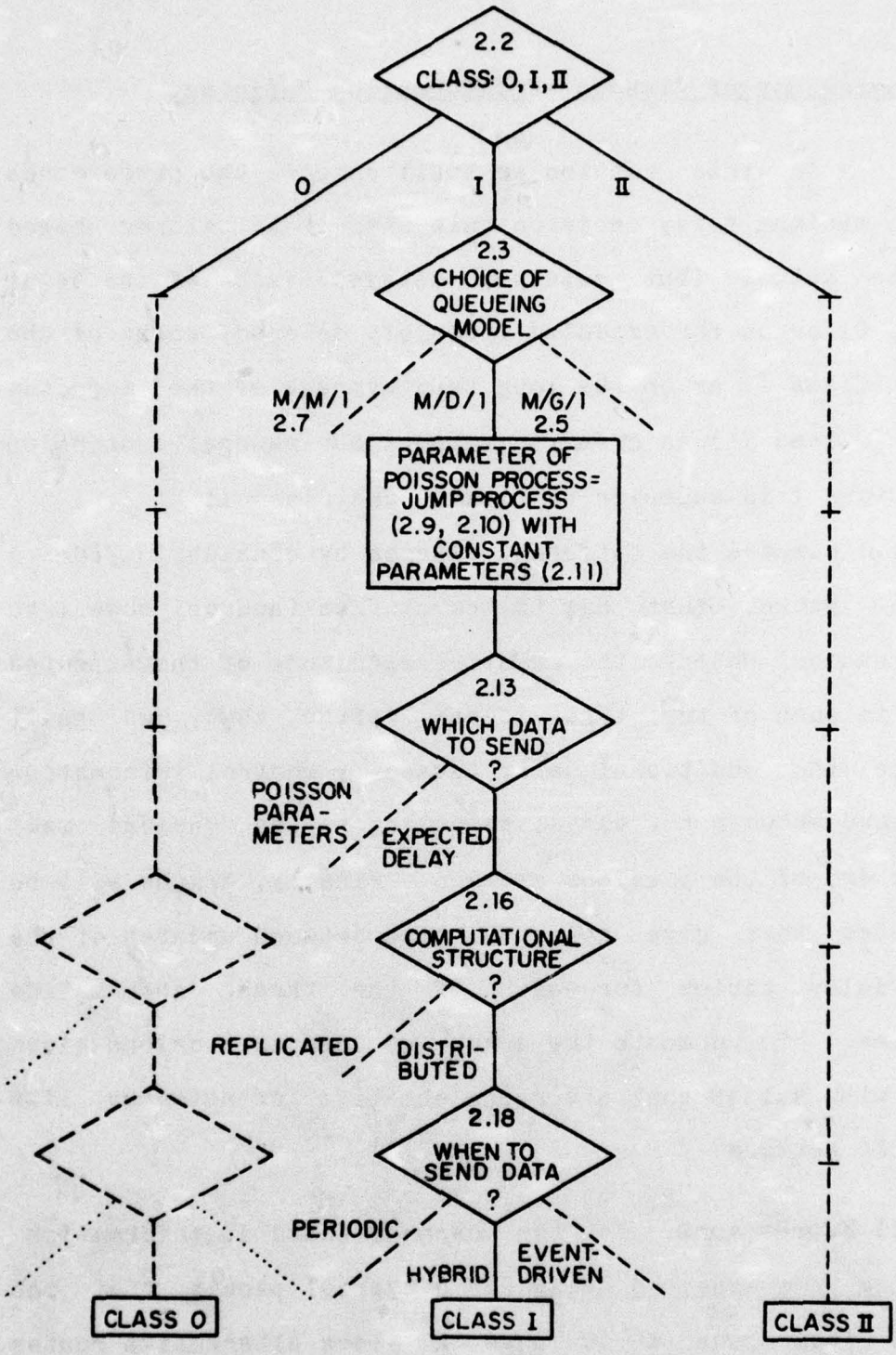


Figure 4:2.a. Design choices for an information policy.

4:3 Comparison of Classes of Information Policies.

In this section we shall compare the performance of the minimum delay decision rule when it is either based on the actual (but possibly delayed) value of the delay (Class 0) or on the expected (possibly delayed) value of the delay (Class I) or on the long term average of the expected delay (Class II) in order to support our general contention that Class I is superior to Class 0 and Class II.

We shall compare the different classes by considering for a typical packet that has to travel from (source) node i to (destination) node k, the relative magnitude of the expected delay in each of the three cases. After that, we shall include the additional delay caused by control information exchanged between the nodes, according to the choices made at the end of the previous section. Finally, graphs will be presented that give delay vs. time between updates of the node delay tables for each of the three information policies. To compute the graphs we used expressions given below with values that are representative for networks like the ARPA Network.

(4:3.1) Expressions. In the case of Class II information, the long term expected delay for a typical packet that can travel from node i to node k along alternative routes 1,...,R is

(4:3.2) $E_{II} = \min_{r=1,\ldots,R}\{\text{long term expected delay along route } r\}$

For Class I and Class 0 we have the following expressions

(4:3.3) $$E_I = PW_I+(1-P)\tilde{W}_I$$

(4:3.4) $$E_0 = PW_0+(1-P)\tilde{W}_0$$

P is the probability that the expected values of the delays along the alternative routes have not changed in the time interval between collecting data for the routing decision and transmitting the packet. In other words, P is the probability that the traffic pattern has not changed between the computation of the node delay table and the time that the packet travels from node i to node k. W_i and W_0 are the long term expected delays under Class I and Class 0 information given that the traffic pattern has not changed. $\tilde{W}_I$ and $\tilde{W}_0$ are the corresponding quantities, given that the traffic pattern has changed. The quantities W_I, $\tilde{W}_I$, W_0, $\tilde{W}_0$ depend on the probability distributions of the expected delay along the different routes and the (conditional) probability distributions of the actual delay given the expected delay. In Appendix 4:B expressions are derived for those quantities in terms of the probability densities.

We still need to assess the probability P that the values of the expected delays are still the same at the time of transmission of the packet. Let T be the average time between changes of the values of the expected delays, and suppose the time between changes has a negative exponential

distribution. Further let U be the time between the starts of updates (the start of an update is considered to coincide with the time of collecting data), and let V be the time between the start of an update and the time the data are available. The time elapsed since the last change, t, has an negative exponential distribution with parameter T. The time since the last update that started longer then V ago, u, has an uniform distribution on (V,U+V). The probability P corresponds to the probability that t>u. For given u this is

$$\Pr\{t>u|\ u\} = \int_u^\infty \frac{1}{T} e^{-t/T} d$$

integrating over the uniform distribution gives

(4:3.5)
$$P = \frac{1}{U}\int_V^{U+V}\int_u^\infty \frac{1}{T} e^{-t/T} dt\ d$$
$$= \frac{T}{U}(e^{V/T} - e^{-(U+V)/T})$$

(4:3.6) Remark. Our choice of information class I can be "proved" to be optimal if we can show that $E_I \leq E_0$ and $E_I \leq E_{II}$ for a typical large packet switched computer network (see (4:3.9) for what is meant by "typical large"). One can see from (4:3.2)-(4:3.4) that in order to prove optimality in general it would be sufficient to show that a) $W(I) \leq E_{II}$, b) $W_I \leq W_0$, c) $\tilde{W}_I \leq E_{II}$, and d) $\tilde{W}_I \leq \tilde{W}_0$ if we do not consider the additional delay due to control information. Let us admit here that it is not possible to show that in general a), b), c), and d) are true. It can be shown that a) and b) are true in general but d) may be not true for some pathological

choices of the probability distribution for the expected delay and for the actual delay given the expected delay. Moreover, c) is in general not true. Therefore additional conditions on the probability densities, and conditions on the value of P that ensure what is gained in b) has more weight than what is lost in d) are needed. Rather than to try to find these precise conditions we will determine typical values of E_0, E_I, E_{II}, that are adjusted to include additional delay due to routing information and from which we can also infer how much the differences in expected delay are.

The final factor we have to take in account is the increase in delay experienced by data packets due to communication between nodes of delay values (control information). In a Class II information scheme, the delay values can be considered constant which means that there is no increase because of control information. For Class I and Class 0 information schemes, we assume that the delay values are exchanged between nodes by means of the decentralized computation of minimum delays according to (4:2.16). Although (4:2.16) was formulated in terms of Class I information, it also holds for Class 0 information merely by reading actual delay for expected delay.

(4:3.7) Expression. (Increase in delay due to control information.) Let U be the time between successive series of iterations according to (4:2.16). One iteration consists of the comparison of the minimum p-hop delay vectors of its neighbors and the exchange of the new computed (p+1)-hop delay vectors. Each vector travels as a packet of v bits. Let M be the total number of iterations per computation of minimum delays for the whole network. Then the average number of packets of control information per second that travels on each link is $\frac{M}{U}$ packets per second, which implies a capacity reduction on that link of $\frac{Mv}{U}$ bits/sec. This implies an increase in queueing delay, which would otherwise be $\frac{\beta^2}{2(1-\beta)}$. The service time β is related to the line capacity C by $\beta=\frac{b}{C}$ where b is the number of bits in a data packet. A little bit of calculus with the three expressions just mentioned gives for the relative increase of queueing delay

(4:3.8)
$$\frac{d+\Delta d}{d} = \frac{C^2-C_dC}{C_r^2-C_dC_r}$$

where

C = line capacity

$C_r = C-\frac{Mv}{U}$ = reduced line capacity

C_d = b = capacity needed for data packets.

Of course the factor should only multiply the portion of the delay which is in excess of the fixed delay $W_{1\ fix}$, respectively $W_{0\ fix}$, where the fixed delay is the delay along the path when all queues were empty.

Now typical values must be chosen for the parameters that appear in (4:3.2)-(4:3.5) and (4:3.8) before we can draw representative graphs that give long term expected delay versus frequency of updating, 1/U, for each of the three information policies.

(4:3.9) Values. (Typical values for a large network.) For the long term expected delay we distinguish between information class (II, I, 0), and whether the traffic pattern (t.p.) has changed or not. Also for Class I and 0 that portion of the delay that is fixed (i.e. when all queues are empty) is given.

E_{II} = 447 msec : Class II

W_I = 350 msec : Class I, t.p. not changed

$\tilde{W}_I$ = 477 msec : Class I, t.p. changed

$W_{I\,fix}$ = 128 msec : Class I, fixed portion

W_0 = 376 msec : Class 0, t.p. not changed

$\tilde{W}_0$ = 481 msec : Class 0, t.p. changed

$W_{0\,fix}$ = 134 msec : Class 0, fixed portion

The parameters above are determined by the choices of probability densities, according to the formulas given in appendix 4:B. The probability densities that were actually used in the computation are given in figures (4:B.a)-(4:B.c). They were inspired by graphs for the distribution of delay of messages in the ARPA Network from the Network Measurement Centre.

Further we have

$T = 10$ sec : average time between changes in t.p.

$V = \max\{\frac{H-1}{M}U, V_{min}\}$: time between start of update and availability of new routing data. The expression for V is explained below.

$H = 5$ hops : number of hops in typical path between source node and destination node.

$M = 10$ hops : maximum hop distance in the network

The number of iterations in an update of the node delay tables is M but after H-1 iterations delay values to destinations that are H or less hops away may be used. Assuming that iterations are equally spaced in the interval between updates we find $V = \frac{H-1}{M}U$. However there is a minimum amount of time that is required for one iteration resulting in

$V_{min} = .2$ sec : time for 4 iterations

This conservative estimate (i.e. the maximum of the minimum time for 4 iterations) is obtained as follows: The time for one iteration is the maximum time it takes to get the delay vector to the neighbor node, this is .02 sec waiting for the current packet of 1000 bits going out @ 50K bits/sec, .022 sec for the delay vector of 600 bits @ 50K bits/sec + 3000

kilometer @ 300,000 km/sec (speed of light) plus .001 sec processing time: 180 comparisons and additions (n=3 neighbors per node, N=60 nodes) @ 5 μsec. Hence V_{min} = 4·(.02+.022+.001) = .2 sec.

C = 50K bits/sec : line capacity

$C_r = 50 - \frac{6}{U}K$ bits/sec : reduced line capacity because of routing information flowing

C_d = 25K bits/sec : capacity needed for data packets

With the given values we can compute the graphs below, of which the last one summarizes the long term expected delay as a function of the frequency of updating, for each of the three information policies.

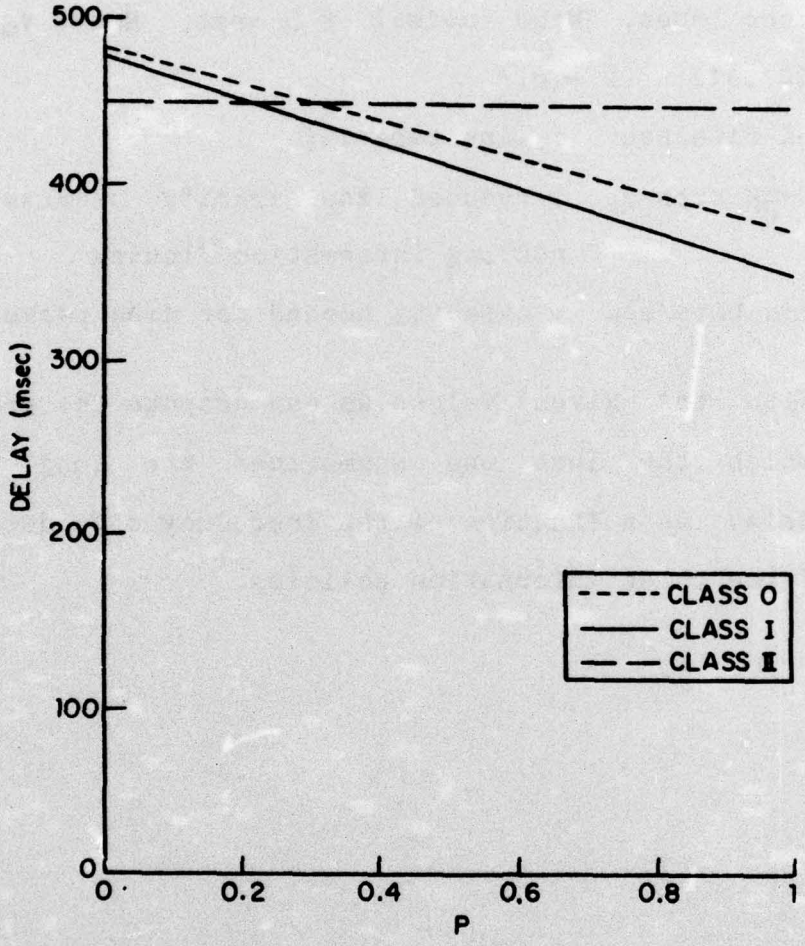


Figure 4:3.a. Delay vs. P. P is the probability that the traffic pattern has not changed since the last update. The long term expected delay for a typical packet here, does not include delay due to routing information.

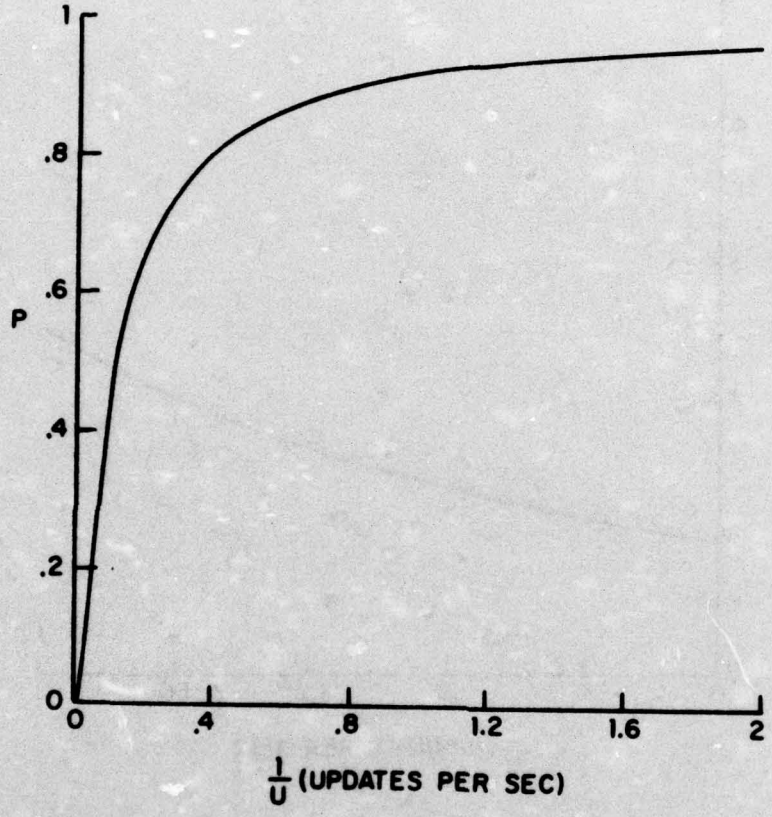


Figure 4:3.b. P vs. frequency of updating. U is the time between successive updates. The probability that the traffic pattern has not changed since the last update P is an increasing function of the frequency of updating $\frac{1}{U}$.

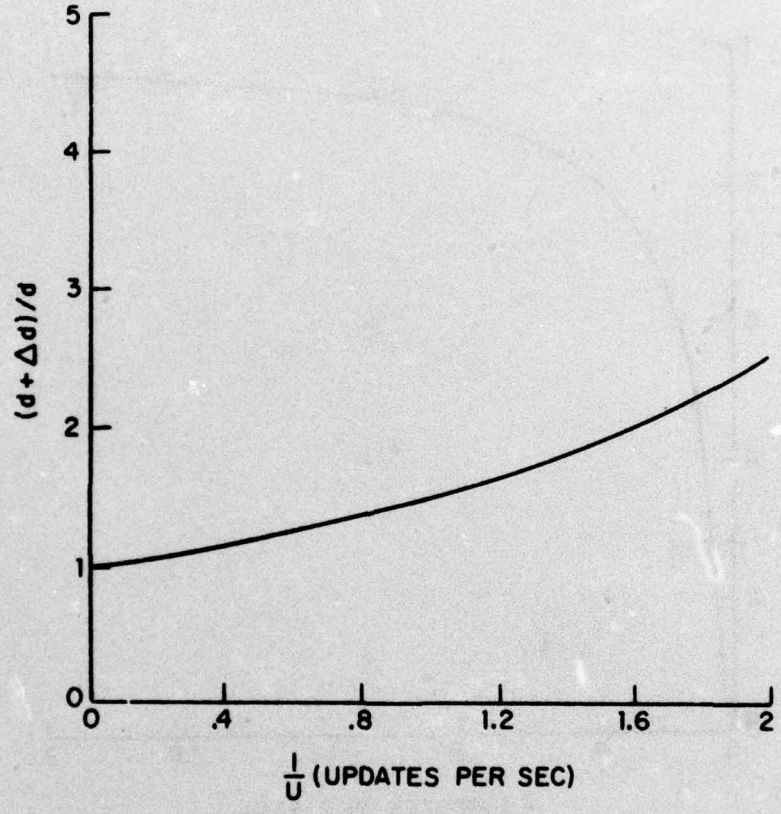


Figure 4:3.c. Relative increase in delay vs. frequency of updating. The relative increase in delay $(d+\Delta d)/d$ gives the factor by which the queueing delays increase due to routing information flowing through the network.

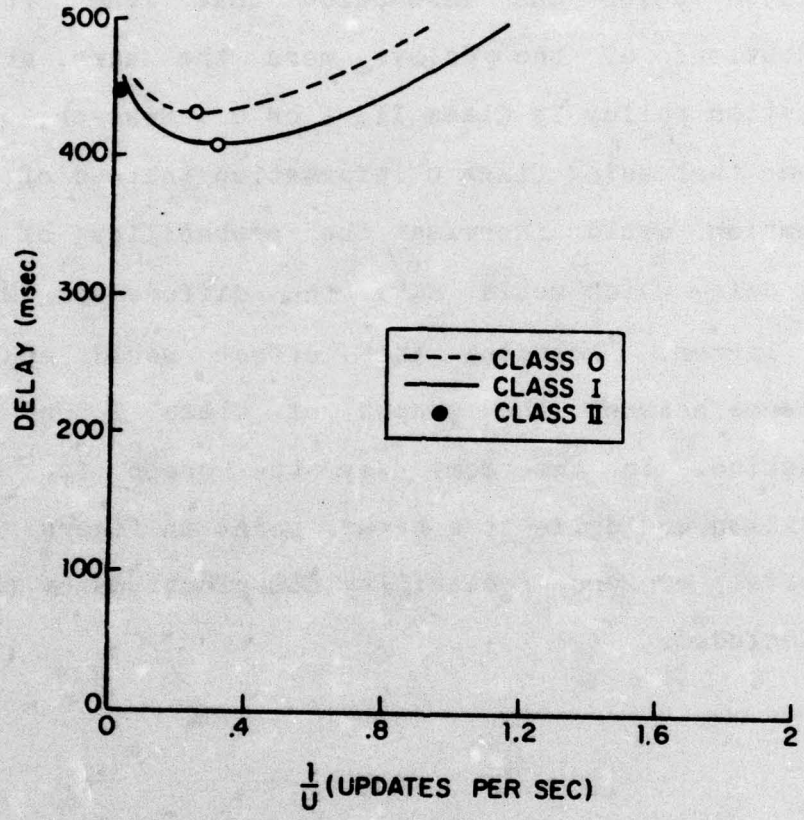


Figure 4:3.d. All-in delay vs. frequency of updating. The long term expected delay including the delay due to routing information has for Class I a minimum of 412 msec at .3 updates/sec, for Class 0 information there is a minimum of 433 msec at .25 updates/sec. Class II information implies fixed routing. For fixed routing the frequency of updating is essentially 0. The corresponding long term expected delay is 447 msec.

(4:3.10) Remark. Figure 4:3.d gives a slightly conservative view of the differences in long term expected delay under Class II, Class I and Class 0 information. The graphs were calculated under the assumption that the probability distributions of the delays were the same, whether the information policy is Class II, I or 0 . However, we may now conclude that using Class 0 information instead of Class I information would increase the probability of having a larger delay which would make the differences $W_I - W_0$ and $\tilde{W}_I - \tilde{W}_0$ larger. Including this effect would enhance the difference between the graphs of Class I and Class 0 information. In the same way the graph for class II information would lie at a higher point in figure 4:3.d if the effect of worse probability distributions of the delays were included.

4:A Appendix. Parameters of the Jump Process.

The time between jumps of the parameter of the Poisson process is typically a few seconds, say 4 seconds. Suppose the estimated parameters of the jump process Π_{est} are close to the true parameters Π_{true}, say $\sum_{i=1}^{L} |\Pi_{est}(i,j)-\Pi_{true}(i,j)| < .1$, $j=1,\ldots,L$, where L is the number of states of the jump process. Then the amount of data that has to be collected to be able to reject the hypothesis $\Pi=\Pi_{est}$ in favor of the hypothesis $\Pi=\Pi_{true}$ with a .05 probability of rejecting inappropriately is typically about 300 or more per state of the jump process. If we assume the number of states L=5, then this amounts to collecting data at least for 5x300x4= 6,000 sec = 1.6 hours which makes adjusting the values of Π well suited for a centralized off line procedure.

4:B Appendix. Derivation of the Long Term Expected Delay with Class 0, I and II Information.

A typical packet that has to travel from node i to node k can do so along alternative routes 1,...,R. Which alternative is chosen depends on the class of information policy and on the actual values of the information. As a direct consequence of the choices of models made in section 4:2, we can model the delay along each of the alternative routes in the following way. The expected delay along $\bar{d}_r$ along route r is a sample of the probability density $g_r(\cdot)$. The actual delay d_r along route r given the expected delay $\bar{d}_r=t$ is a sample of the probability density $f_r(\cdot|t)$. From this it follows that the unconditional delay along route r is a sample from the probability density $\tilde{f}_r(\cdot)$ where $\tilde{f}_r(\cdot)$ is defined by

$$\tilde{f}_r(s) = \int_0^\infty f_r(s|t)g_r(t)\,dt \qquad r=1,\ldots,R$$

The long term expected delay along route r is then given by

$$\tilde{d}_r = \int_0^\infty s\tilde{f}_r(s)\,ds \qquad r=1,\ldots,R$$

In figures 4:B.a-4:B.c, typical probability densities $g_r(\cdot)$, $f_r(\cdot|t)$ for some sample values t and $\tilde{f}_r(\cdot)$ are plotted. They represent the probability densities that were used in computing the values (4:3.9) on which the graphs at the end of section 4:3 were based.

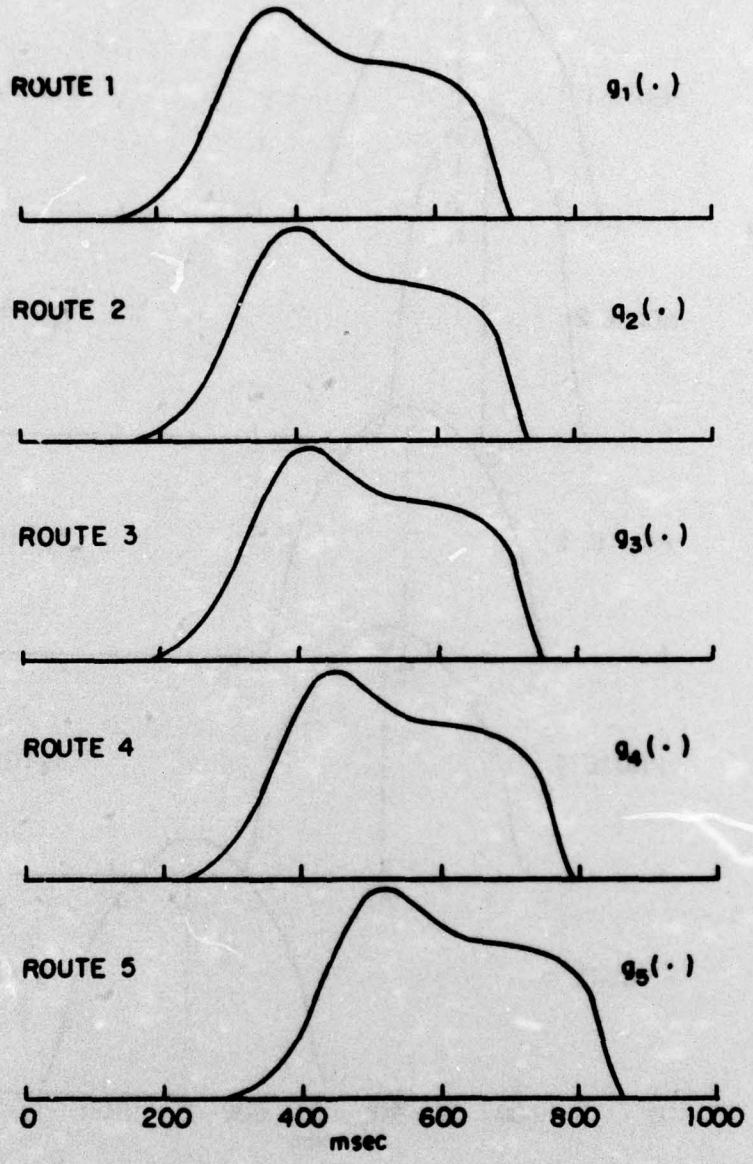


Figure 4:B.a. Probability densities of expected delay $g_r(\cdot)$. A set of independent samples from these densities constitutes a traffic pattern. According to our model the time between taking successive sets of samples, i.e. the time between changes of traffic pattern, has a negative exponential distribution with mean T.

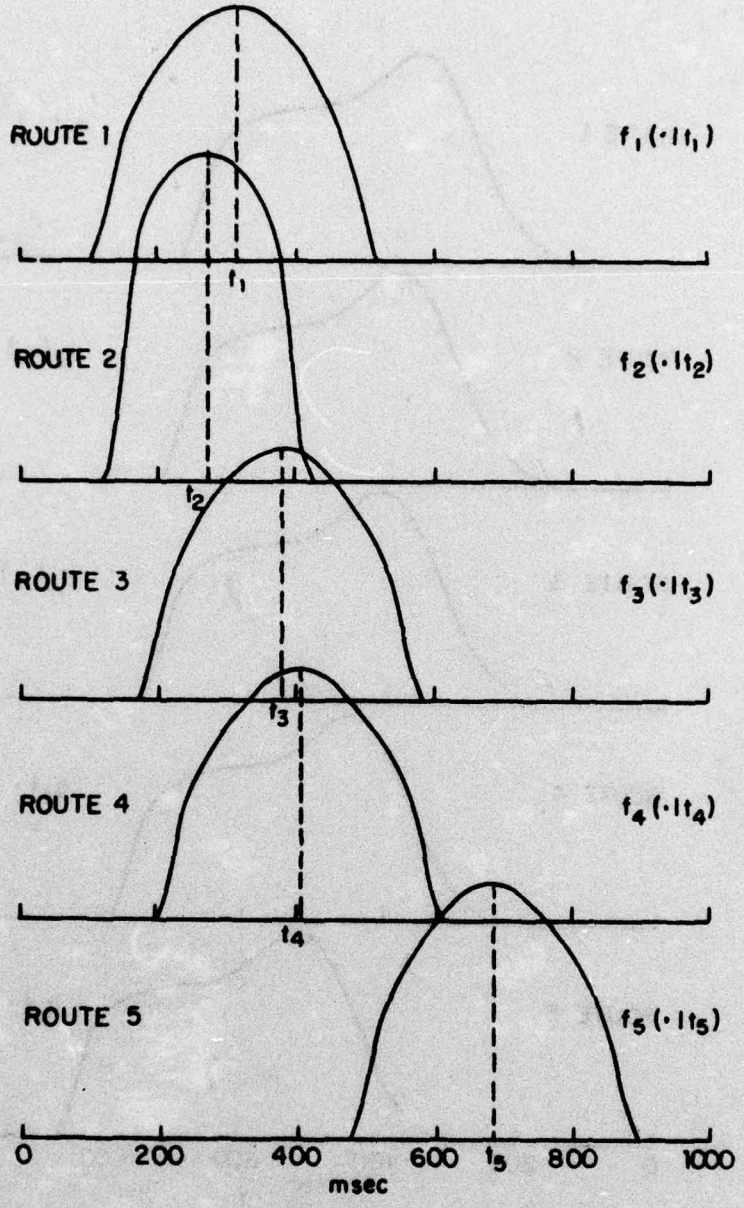


Figure 4:B.b. Probability densities of actual delay given the expected delay $f_r(\cdot|t_r)$. The values $t_1, t_2, \ldots, t_5$ are a set of independent samples from the densities $g_r(\cdot)$. The actual delay along a route r is such a rapidly changing quantity that, still when the traffic pattern has not changed, the reported actual delay and the actual delay delay for the packet are considered independent samples from $f_r(\cdot|t_r)$.

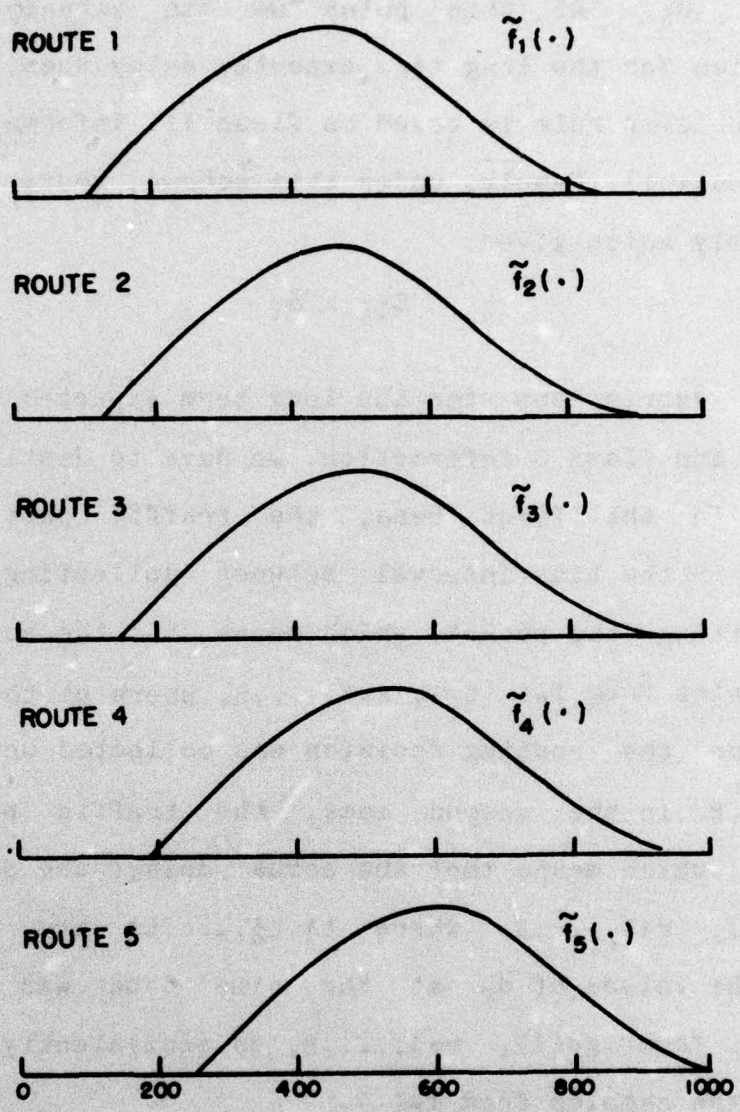


Figure 4:B.c. Unconditional probability densities of actual delay $\tilde{f}_r(\cdot)$. The actual delay along route r when the traffic pattern is not known (anymore) is an independent sample from $\tilde{f}_r(\cdot)$. Taking a sample from $\tilde{f}_r(\cdot)$ is equivalent to first taking a sample from $g_r(\cdot)$, say this is t_r, and then taking a sample from $f_r(\cdot|t_r)$.

We assume that the routes are numbered in such a way that $\bar{d}_1 \leq \bar{d}_2 \leq \ldots \leq \bar{d}_R$. At this point we can already give the expression for the long term expected delay when the minimum delay decision rule is based on Class II information (i.e. fixed routing). Namely, under this scheme, route 1 is chosen invariably which gives

(4:B.1) $$E_{II} = \bar{d}_1$$

To get expressions for the long term expected delay under Class I and Class 0 information, we have to distinguish two cases. In the first case, the traffic pattern has not changed in the time interval between collecting data and transmitting the packet, which means that the actual delays are samples from $f_r(\cdot|t_r)$, $r=1,\ldots,R$, where at the time the data for the routing decision was collected we had $\bar{d}_r=t_r$, $r=1,\ldots,R$. In the second case, the traffic pattern has changed, which means that the actual delays are samples from $f_r(\cdot|t_r')$, $r=1,\ldots,R$, where $t_1',t_2',\ldots,t_R'$ are independent (from the values of $\bar{d}_r$ at the time data was collected) samples from $g_r(\cdot)$, $r=1,\ldots,R$, or equivalently the actual delays are samples from $\bar{f}_r(\cdot)$.

Let r_I denote the route chosen by applying the minimum delay decision rule to Class I information:

(4:B.2) $$r_I = \arg\{\min_{r=1,\ldots,R}\{t_r\}\}$$

where t_r is the expected delay of route r at the time that data is collected. Let r_0 be the route chosen with Class 0 information, i.e.,

(4:B.3) $$r_I = \arg\{\min_{r=1,\ldots,R}\{s_r\}\}$$

where s_r is a sample from $f_r(\cdot|t_r)$, r=1,...,R. This sample is assumed to be always independent of the actual delay the packet experiences going from i to k, representing the fact that the actual delay is very rapidly changing compared to the delay in information.

In the first case (when expected delays have not changed) with Class I information, the long term expected delay given r_I=r weighted with the probability that r_I=r is

(4:B.4) $$\int_0^\infty t g_r(t) \prod_{p\neq r} (1-G_p(t))\, dt$$

where $G_p(\cdot)$ is the distribution function corresponding to $g_p(\cdot)$. This result (and also later results of the same character) is easily understood by noting that $g_r(t)dt$ represents the probability that a sample of $g_r(\cdot)$ has the value t and that $(1-G_p(t))$ is the probability that a sample from $g_p(\cdot)$ has a value >t. Summing (4:B.4) over all possible values of r, we get the long term expected delay under Class I information, given that the traffic pattern has not changed.

(4:B.5) $$W_I = \sum_{r=1}^{R} \int_0^\infty t g_r(t) \prod_{p\neq r} (1-G_p(t))\, dt$$

When expected delays have not changed and we have Class 0 information, the probability that r_0=r given that the expected delay along route r is

(4:B.6) $$\int_0^\infty f_r(s|t) \prod_{p\neq r} (1-\bar{F}_p(s))\, ds$$

where $\tilde{F}_p(\cdot)$ is the distribution function corresponding to $\tilde{f}_p(\cdot)$. Similar to (4:B.5), the long term expected delay under Class 0 information given that the traffic pattern has not changed is

(4:B.7) $$W_0 = \sum_{r=1}^{R} \int_0^\infty t g_r(t) \int_0^\infty f_r(s|t) \prod_{p\neq r} (1-\tilde{F}_p(s)) \, ds \, dt$$

Now we consider the second case where the expected values of the delays along the alternative routes at the time of transmission of the packet are independent of the values of the expected delays at the time data was collected for the routing decision. In this case, the long term expected delay given that route r is chosen is simply $\tilde{d}_r$, for both Class I and Class 0 information. The probability that route r is chosen is

$$\int_0^\infty g_r(t) \prod_{p\neq r} (1-G_p(t)) \, dt$$

for class I, and

$$\int_0^\infty \tilde{f}_r(t) \prod_{p\neq r} (1-\tilde{F}_p(t)) \, dt$$

for Class 0. Therefore, the long term expected delay under Class I information given that the traffic pattern has changed is

(4:B.8) $$\tilde{W}_I = \sum_{r=1}^{R} \tilde{d}_r \int_0^\infty g_r(t) \prod_{p\neq r} (1-G_p(t)) \, dt$$

and the corresponding quantity for class 0 information is

(4:B.9) $$\tilde{W}_0 = \sum_{r=1}^{R} \tilde{d}_r \int_0^\infty \tilde{f}_r(t) \prod_{p\neq r} (1-\tilde{F}_p(t)) \, dt$$

The probability densities $\tilde{f}_r(\cdot)$, r=1,...,R, also determine what portion of the delay is the fixed delay. By fixed delay

we mean the delay along a route when all queues are empty. According to the assumption that all variation in delay is a consequence of variations in queue length (see remark (4:1.4)) it follows that the fixed delay along route r is

$$d_{r\ fix} = \max(\ d\ )\{d|\ \bar{f}_r(d')=0\ \forall d' \leq d\}$$

The fixed portion of the delay under Class I or class 0 information is then the sum of the fixed delays along each of the routes, weighted with the probability that that route is chosen. Therefore we get similar to (4:B.8) and (4:B.9)

(4:B.10) $$\bar{W}_{I\ fix} = \sum_{r=1}^{R} d_{r\ fix} \int_0^\infty g_r(t) \prod_{p\neq r} (1-G_p(t))\ dt$$

and

(4:B.11) $$\bar{W}_{0\ fix} = \sum_{r=1}^{R} d_{r\ fix} \int_0^\infty \bar{f}_r(t) \prod_{p\neq r} (1-\bar{F}_p(t))\ dt$$

CHAPTER 5: CONCLUSIONS.

In the decentralized control of computer communication it is appropriate to consider the controllers as having the same objective: the efficient operations of the communication medium. In this sense decentralized control of computer communication fits in the area of multi-person decision making that is designated as team theory. Section 2:1 was devoted to aspects of team theory, specifically to the introduction of the notions of symmetric team problems and symmetric solutions. For the purpose of introduction and immediate application it was sufficient to consider only finite team problems in strategic form. For the purpose of making the theory more complete it is important to extend the definition and analysis of symmetric team problems and symmetric solutions to team problems with an infinite number of possible strategies and to consider the problem in extensive form.*)

Finding applications of the notions of symmetric team problems and symmetric solutions is not hard. By way of illustration we considered the corridor problem in section

---

*) In a team problem in strategic form, each decision maker chooses from his set of possible strategy-maps, a map which maps his (sequence of) information into his (sequence of) decisions. In extensive form a decision is determined for each decision maker at each point in time, given his information at that point in time [6].

2:1, and later in section 2:2 the access problem in wire communication was considered. The unrestricted and symmetric solutions to the access problem correspond to two different information structures: in the unrestricted case the information of what number is assigned to a station can be used to base decisions on, in the symmetric case it can not be used. A third information structure is considered for comparison purposes. It can be concluded that the information structure is an important parameter in finding and explaining different solutions to problems in decentralized control.

The multi-access satellite channel, which was considered in chapter 3, has as its characteristic feature that the ground stations can only share information with considerable delay. The model was chosen such that with some light assumptions it could be proven that the delayed information was as good as no information at all, or in other words, the optimal control law could be open-loop. In general for multi-person decision problem with delayed sharing of information such a strong result can not be proved. However it may be feasible and worthwhile to show that in general as the delay gets larger the shared information gets less valuable in terms of the objective function. From the result which states that the optimal control law is open loop, it should not be concluded that in

the operation of the satellite channel there is no need for feedback at all. The result only means that feedback should not be at the level of delayed reported values of the state. However when parameters, such as the arrival rate of packets at the stations p, are slowly (compared with the delay) changing, then feedback can be at the level of changing the open-loop control law whenever the current value of p is updated.

The suggested feedback in the preceding paragraph, corresponds directly to the main conclusion of chapter 4. In this chapter on information policies for routing in a computer network, different classes of information were defined on which a given decision rule could be based. Class 0 corresponds to delayed sharing of the rapidly changing value of the state of the network, Class I to the delayed sharing of the expected value of the state of the network, which is slowly changing and Class II corresponds to sharing a constant: the long term expected state of the network. The conclusion was that the decision rule should be based on Class I information, i.e. feedback on the level of a slowly changing parameter. The definition of different classes of information policies was based on modeling a stochastic process (here the value of the delay along a link) as a cascade of stochastic processes. At each next level of the cascade the (expected) time between changes of the value of

the stochastic process is larger than at the previous level. In chapter 4 this method of constructing different classes of information policies was plausible in view of the specific application, but the validity of this method is probably much more general. Therefore it would be useful to find generalizations that are well founded on principles of stochastic processes and mathematical statistics.

The conclusion of the conclusions could be stated as:

> In decision making, particularly in decentralized decision making, the question what is the information on which decisions are based is of prime importance. Next is the question what decisions are made.

REFERENCES

[1] N.R. Sandell Jr., P. Varaiya and M. Athans, "A Survey of Decentralized Control Methods for Large-Scale Systems", to appear in IEEE Trans. Aut. Control.

[2] Y.C. Ho and S.K. Mitter, Directions in Large-Scale Systems, Plenum Publishing, New York, 1976.

[3] R.M. Metcalfe, "Packet Communication", Ph.D. Dissertation, Harvard University, 1973.

[4] H. Witsenhausen, "Information Structures, Feedback and Causality", SIAM J. Control 9, 149-159, 1971.

[5] Y.C. Ho and K.C. Chu, "Team Decision Theory and Information Structures in Optimal Control Problems", IEEE Trans. Aut. Control AC-17, Feb. 1972.

[6] Y.C. Ho and K.C. Chu, "Information Structure in Dynamic Multi-Person Control Problems", Automatica, Vol.10, pp. 341-351, 1974.

[7] H.G. Tucker, A Graduate Course in Probability, Academic Press, New York 1967.

[8] L. Kleinrock and F.A. Tobagi, "Packet Switching in Radio Channels: Part I", IEEE Trans. on Commun., vol. COM-23, no.12, Dec 1975.

[9] W. Feller, An Introduction to Probability Theory and its Applications, vol. II, New York 1957.

[10] R.M. Metcalfe and D.R. Boggs, "Ethernet: Distributed Packet Switching for Local Computer Networks", Comm. of ACM, vol 7, no.19, July 1976.

[11] H.S. Witsenhausen, "Separation of Estimation and Control for Discrete Time Systems", proc. IEEE, vol. 59, pp 1557-1566, Nov 1971.

[12] M. DeGroot, Optimal Statistical Decision Theory, New York: McGraw Hill, 1970.

[13] R.E. Bellman and S.E. Dreyfus, Applied Dynamic Programming, Princeton University Press, 1962.

[14] S.S. Lam and L.Kleinrock, "Packet Switching in a Multi-access Broadcast Channel: Dynamic Control Procedures", IEEE Trans. Commun., vol. Com-23, pp 891-904, Sep 1975.

[15] G. Fayolle, E. Gelenbe, J. Labetoulle, "Stability and Control of Packet Switching Broadcast Channels," Institut de Recherche d'Informatique et d'Automatique, Rapport de Recherche no. 116, Apr 1975.

[16] R.P. Blanc and I.W. Cotton Eds., Computer Networking, IEEE Press N.Y., 1976

[17] P.K. Verma, Ed., Proceedings of the Third International Conference on Computer Communication, International Council for Computer Communication, Washington D.C., 1976.

[18] J.M. McQuillan, "Adaptive Routing Algorithms for Distributed Computer Networks", Ph.D. Dissertation, Harvard University, 1974.

[19] R.G. Gallagher, "An Optimal Routing Algorithm Using Distributed Computation", IEEE Trans. on Communications, 1977

[20] A. Segall, "The Modeling of Adaptive Routing in Data Communication Networks", IEEE Trans. on Communications, 1977

[21] J.W. Cohen, The Single Server Queue, North Holland Publishing Company, 1969

JSEP REPORTS DISTRIBUTION LIST

N00014-75-C-0648

Defense Documentation Center
Attn: DDC-TCA (Mrs. V. Caponio)
Cameron Station
Alexandria, Virginia 22314

Asst. Director
Electronics and Computer Sciences
Office of Director of Defense Research and Engineering
The Pentagon
Washington, DC 20315

Office of the Director of Defense Research and Engineering
Information Office Library Branch
The Pentagon
Washington, DC 20301

GSBB and E Advisory Group on Electron Devices
201 Varick Street
New York, New York 10014

Chief, R and D Division (360)
Defense Communications Agency
Washington, DC 20301

Director, National Security Agency
Fort George G. Meade, Maryland 20755
Attn: Dr. T. J. Beahn

Institute for Defense Analysis
Science and Technology Division
400 Army-Navy Drive
Arlington, Virginia 22202

Dr. Stichley
Defense Advanced Research Projects Agency
Attn: Technical Library
1400 Wilson Boulevard
Arlington, Virginia 22209

Dr. R. Reynolds
Defense Advanced Research Projects Agency
Attn: Technical Library
1400 Wilson Boulevard
Arlington, Virginia 22209

AF/RDPS
The Pentagon
Washington, DC 20330

AFSC (Lt/Sr. Irving B. Mirman)
Andrews AFB
Washington, DC 20334

Directorate of Electronics and Weapons
HQ AFSC/DLC
Andrews AFB, Maryland 20334

Directorate of Science
HQ AFSC/DLS
Andrews AFB
Washington, DC 20334

LTC J. W. Gregory
AF Member, TAC
Air Force Office of Scientific Research
Bolling AFB
Washington, DC 20332

Mr. Carl Blaise
AFCRL/LZ
Hanscom AFB, MA 01731

Dr. Richard Picard
AFCRL/OPL
Hanscom AFB, MA 01731

Mr. Robert Barrett
AFCRL/LQ
Hanscom AFB, MA 01731

Dr. John H. Howard
AFCRL (CA)
Hanscom AFB, MA 01731

Dr. Richard D. Hisch
AFCRL/LZB
Hanscom AFB, MA 01731

Documents Library (TSLD)
Rome Air Development Center
Griffiss AFB, New York 13441

Mr. H. E. Webb, Jr. (ESCP)
Rome Air Development Center
Griffiss AFB, New York 13441

Mr. Murray Kesselman (ESCA)
Rome Air Development Center
Griffiss AFB, New York 13441

Mr. W. Edwards
AFAL/TE
Wright-Patterson AFB, Ohio 45433

Mr. R. D. Larsen
AFAL/WRI
Wright-Patterson AFB, Ohio 45433

Chief Scientist
AFAL/CA
Wright-Patterson AFB, Ohio 45433

HQ ESD (DRA/Maj Ho)
Hanscom AFB, MA 01731

Professor R. E. Fontana
Head, Dept. of Electrical Eng.
AFIT/ENE
Wright-Patterson AFB, Ohio 45433

Mr. John Mathematik (SSCT)
HQ ESD (AFSC)
Hanscom AFB, MA 01731

LTC Richard J. Gowen
Professor
Dept. of Electrical Engineering
USAF Academy, Colorado 80840

AUL/LSE-9663
Maxwell AFB, Alabama 36112

AFETR Technical Library
P.O. Box 4608, MU 1650
Patrick AFB, Florida 32542

ADTC (DLOSL)
Eglin AFB, Florida 32542

HQ AMD (RDR/COL Geddes)
Brooks AFB, Texas 78235

USAF European Office of Aerospace Research
Technical Information Office
Box 14, FPO, New York 09510

Dr. Carl E. Baum
AFWL (ES)
Kirtland AFB, New Mexico 87117

ASAFSAM/RAL
Brooks AFB, Texas

NODA (DAMA-ARZ-A)
Washington, DC 20310

Commander
US Army Security Agency
Attn: IAAD-T
Arlington Hall Station
Arlington, VA 22212

Commander
US Army Material Development and Readiness Command
Attn: Technical Library, Rm 7S 35
5001 Eisenhower Avenue
Alexandria, VA 22333

Commander
US Army Ballistics Research Laboratory
Attn: DRXBR-BAD
Aberdeen Proving Ground
Aberdeen, MD 21005

Commander
Picatinny Arsenal
Attn: SMUPA-TS-T-S
Dover, NJ 07801

US Army Research Office
Attn: Dr. Hermann Robl
P.O. Box 12211
Research Triangle Park, NC 27709

US Army Research Office
Attn: Mr. Richard O. Ulsh
P.O. Box 12211
Research Triangle Park, NC 27709

US Army Research Office
Attn: Dr. Jimmie R. Suttle
P.O. Box 12211
Research Triangle Park, NC 27709

US Army Research Office
Attn: Dr. Horst Wittmann
P.O. Box 12211
Research Triangle Park, NC 27709

Commander
Frankford Arsenal
Attn: Mr. George, C. White, Jr.
Deputy Director, Pitman-Dunn Lab.
Philadelphia, PA 19137

Commander
US Army Missile Command
Attn: Chief, Document Section
Redstone Arsenal, AL 35809

Commander
US Army Missile Command
Attn: DRSMI-RR
Redstone Arsenal, AL 35809

Commander
US Army Materials and Mechanics Research Center
Attn: Chief, Materials Science Division
Watertown, MA 02172

Commander
Harry Diamond Laboratories
Attn: Mr. John E. Rosenberg
2800 Powder Mill Road
Adelphi, MD 20783

Commandant
US Army Air Defense School
Attn: ATSAD-T-CSM
Fort Bliss, TX 79916

Commandant
US Army Command and General Staff College
Attn: Acquisitions, Lib. Div.
Fort Leavenworth, Kansas 66027

Dr. Hans K. Ziegler
Army Member, TAC/JSEP
US Army Electronics Command
(DRSEL-TL-D)
Fort Monmouth, NJ 07703

Mr. J. E. Toth
Executive Secretary, TAC/JSEP
US Army Electronics Command
(DRSEL-TL-DT)
Fort Monmouth, NJ 07703

Director
Night Vision Laboratory, ECOM
Attn: DRSEL-NV-D
Fort Belvoir, VA 22060

Commander/Director
Atmospheric Sciences Laboratory
(ECOM)
Attn: DRSEL-SA-SD
White Sands Missile Range, NM 88002

Director
Electronic Warfare Laboratory
(ECOM)
Attn: DRSEL-WL-MT
White Sands Missile Range, NM 88002

Commander
US Army Armament Command
Attn: DRSAR-RD
Rock Island, IL 61201

Director, Div. of Neuropsychiatry
Walter Reed Army Institute of Research
Washington, DC 20012

Commander
USASATCOM
Fort Monmouth, NJ 07703

Commander
US Army R and D Group (Far East)
APO San Francisco, CA 96343

Commander
US Army Communications Command
Attn: Director, Advanced Concepts Office
Fort Huachuca, AZ 85613

Project Manager
AAFADS
EAI Building
West Long Branch, NJ 07764

Commander
US Army White Sands Missile Range
Attn: STEWS-ID-R
White Sands Missile Range, NM 88002

Director, TRI-TAC
Attn: TT-AD (Mrs. Bethler)
Fort Monmouth, NJ 07703

Commander
US Army Communication Command
Attn: CC-OPS-PD
Fort Huachuca, AZ 85613

COL Robert Noce
Senior Standardization Representative
US Army Standardization Group, Canada
Canadian Force Headquarters
Ottawa, Ontario, Canada K1A OK2

Commander
US Army Electronics Command
Attn: DRSEL-RD-O (Dr. W. S. McAfee)
CT-L (Dr. R. Buser)
NL-O (Dr. H. S. Bennett)
NL-T (Mr. R. Kulsrud)
TL-D
VL-D
WL-D
TL-MM (Mr. N. Lipsto)
NL-H (Dr. F. Schwering)
TL-E (Dr. S. Kronenberg)
TL-E (Dr. J. Kohn)
TL-I (Dr. C. Thornton)
NL-B (Dr. S. Amoroso)
Fort Monmouth, NJ 07703

Project Manager
Ballistic Missile Defense Program Office
Attn: DACS-BMP (Dr. A. Gold)
1300 Wilson Blvd.
Washington, DC 22209

Dr. Sidney Ross
Technical Director
SARFA-TD
Frankford Arsenal
Philadelphia, PA 19137

Department of Naval Research
Electronic and Solid State Sciences Program (Code 427)
800 N. Quincy
Arlington, Virginia 22217

Office of Naval Research
Code 200
Assistant Chief for Technology
800 N. Quincy
Arlington, Virginia 22217

Office of Naval Research
Information Sciences Program
(Code 437)
800 N. Quincy
Arlington, Virginia 22217

Naval Research Laboratory
4555 Overlook Avenue, S. W.
Washington, DC 20375
Attn: Codes 2627
4000
4105
5000
5200
5203
5210
5270
5300
5400
5460
5464
5900
5910
6400

Director
Office of Naval Research Branch Office
536 South Clark Street
Chicago, Illinois 60605

San Francisco Area Office
Office of Naval Research
760 Market Street, Room 447
San Francisco, CA 94102

Dr. A. Laufer
Chief Scientist
Office of Naval Research Branch Office
1030 East Green Street
Pasadena, CA 91101

Director
Office of Naval Research
Branch Office
715 Broadway, 5th Floor
New York, New York 10003

Mr. J. W. Sumner
Naval Electronics Systems Command
NC No. 1
2511 Jefferson Davis Highway
Arlington, VA 20360

Mr. B. Frankle
Naval Electronics Systems Command
NC No. 1
2511 Jefferson Davis Highway
Arlington, VA 20360

Mr. N. Butler
Naval Electronics Systems Command
NC No. 1
2511 Jefferson Davis Highway
Arlington, VA 20360

Dr. N. J. Moseller
Naval Air Systems Command
JP No. 1
1411 Jefferson Davis Highway
Arlington, VA 20360

Capt. R. B. Mesko
Naval Sea Systems Command
NC No. 3
2531 Jefferson Davis Highway
Arlington, VA 20362

Commander
Naval Surface Weapons Center
Attn: Technical Library
Silver Spring, Maryland 20910

Officer-in-Charge
Naval Surface Weapons Center
Dahlgren Laboratory
Dahlgren, Virginia 22448

Naval Air Development Center
Attn: Technical Library
Johnsville
Warminster, Penn. 18974

Commander
Naval Avionics Facility
Indianapolis, Indiana 46241
Attn: D/035 Technical Library

Naval Missile Center
Technical Library
Code 5632.2
Point Mugu, CA 93042

Naval Weapons Center
Attn: Technical Library
Code 533
China Lake, CA 93555

Naval Weapons Center
Attn: Code 6010
China Lake, CA 93555

Naval Weapons Center
Mathematics Division
China Lake, CA 93555

Naval Training Equipment Center
Technical Library
Orlando, Florida 32813

Naval Research Laboratory
Underwater Sound Reference Div.
Technical Library
P.O. Box 8337
Orlando, Florida 32806

Naval Undersea Sound Center
Technical Library
New London, Conn. 06320

U.S. Naval Oceanographic Office
Library - Code 1640
Washington, DC 20373

Commandant, Marine Corps
Scientific Advisor (Code AX)
Washington, DC 20380

Dr. Gernot M. R. Winkler
Director, Time Service
US Naval Observatory
Mass. Ave. at 34th St., N.W.
Washington, DC 20390

Naval Postgraduate School
Technical Library
Monterey, California 93940

Naval Electronics Laboratory Center
Technical Library
San Diego, California 92152

Naval Electronics Laboratory Center
Attn: Code 3600
San Diego, CA 92152

Naval Electronics Laboratory Center
San Diego, California 92152
Attn: Codes 2000
2200
2300
2400
2500
2520
2600
2700
2800
2900

Naval Undersea Center
Technical Library
San Diego, California 92152

Naval Ship Research and Development Center
David W. Taylor
Code 522.1
Bethesda, Maryland 20084

Office of Chief of Naval Operations
NAVCOM/MIS Planning Branch
OOP-916D, Pentagon
Washington, DC 20350

Mr. F. C. Schwenk, RD-T
National Aeronautics and Space Administration
Washington, DC 20546

Los Alamos Scientific Laboratory
Attn: Reports Library
P.O. Box 1663
Los Alamos, New Mexico 87544

M. Rose Thornton
Deputy Director Institute for Computer Sciences and Technology
National Bureau of Standards
Washington, DC 20234

Director, Office of Postal Technology (G and D)
US Postal Service
11711 Parklawn Drive
Rockville, Maryland 20852

NASA Lewis Research Center
Attn: Library
21000 Brookpark Road
Cleveland, Ohio 44135

Library - 851
Bureau of Standards
Acquisitions
Boulder, Colorado 80302

MIT Lincoln Laboratory
Attn: Library A-082
P.O. Box 73
Lexington, Mass. 02173

Dr. Jay Harris
Program Director
Devices and Waves Program MEF
1800 G Street
Washington, DC 20550

Dr. Howard W. Etzel
Deputy Director
Div. of Materials Research MEF
1800 G Street
Washington, DC 20550

Dr. Joan Mitchell
Program Director
Solid-State Physics
Div. of Materials Research
National Science Foundation
1800 G Street
Washington, DC 20550

Director
Research Laboratory of Electronics
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

Director
Microwave Research Institute
Polytechnic Institute of New York
Long Island Graduate Center, Route 110
Farmingdale, New York 11735

Asst. Director
Microwave Research Institute
Polytechnic Institute of New York
333 Jay Street
Brooklyn, New York 11201

Director
Columbia Radiation Laboratory
Dept. of Physics
Columbia University
538 West 120th Street
New York, New York 10027

Director
Coordinated Science Laboratory
University of Illinois
Urbana, Illinois 61801

Director
Stanford Electronics Laboratory
Stanford University
Stanford, California 94305

Director
Microwave Laboratory
Stanford University
Stanford, California 94305

Director
Electronics Research Laboratory
University of California
Berkeley, California 94720

Director
Electronics Science Laboratory
University of Southern California
Los Angeles, California 90007

Director
Electronics Research Center
The University of Texas at Austin
Engineering Science Bldg. 112
Austin, Texas 87812

Director of Laboratories
Div. of Engineering and Appl. Physics
Harvard University
Pierce Hall
Cambridge, Mass. 02138