THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 50, No. 8, October, 1971
Printed in U.S.A.

Heavy Traffic Characteristics of a Circular Data Network

By B. AVI-ITZHAK

(Manuscript received April 26, 1971)

Traffic behavior in the Pierce loop for data transmission is studied under assumptions of heavy loading. A deterministic mathematical model for describing traffic flows is developed and analyzed. The mathematical problem is of a linear complementarity type which has not been dealt with in the literature of mathematical programming. An effective procedure, the load-and-shift algorithm, for determining traffic flows is proposed. The procedure yields all feasible solutions for traffic flows and reveals the possibility of stations grouping into dominating classes and preventing other stations from using the system. This property, which can be eliminated by exercising appropriate control, also may affect the stochastic behavior of the system when heavy traffic conditions do not prevail and therefore deserves careful investigation. The paper includes two numerical examples illustrating use of the load-and-shift algorithm and numerical results from a simulation showing some of the effects of dominating classes when heavy traffic conditions do not prevail.

I. INTRODUCTION

The concept of a loop network for data transmission has been proposed recently by J. R. Pierce.[1] In such a network the stations are connected to a closed loop main line on which one-way traffic is allowed. A message to be delivered from one station to another is arranged, at the sending station, into standard packets each carrying the address of the receiving station. These packets are then delivered onto the main line, one at a time, where they flow around in the allowed traffic direction. The address of each packet is checked at each station on the way until it reaches the receiving station where it is removed from the main line. Traffic on the main line cannot be delayed; therefore, a station can deliver a packet onto the main circular line only when permitted by the existence of a gap in traffic or when receiving a packet from the main line. Principal features of the system may be explained with the aid of the four-station network shown schematically in Fig. 1.

The four-armed structure revolves around the central axis and stops briefly every time the four packet-carrying compartments at the ends of the arms are aligned with the four stations. During such a stop each station is able to check the content of the aligned compartment. If the compartment is empty, the station can load it with a packet. If there is a packet in the compartment, it will not be removed unless it is addressed to the said station, in which case the station is permitted to load the compartment again after unloading it. A Pierce loop can be represented by the mechanical analog structure shown in Fig. 1; however, the number of revolving arms in the structure is not necessarily equal to the number of stations in the loop. Rather, the number of arms is determined by the "loop time" of the system (the time needed for a bit to complete one round on the loop). Significance of the loop time is discussed in more detail in Section V.

In the Pierce loop it is currently assumed that no outside control is applied and each station strives to send its messages at the earliest. Therefore, a station will never miss the opportunity to load a compartment unless there are no messages waiting for delivery at the station.

This paper presents a study of the flow characteristics of such a system in heavy traffic, i.e., when the system is not able to deliver all messages, and infinite queues build up at some stations. In Section II the notion of "stable solutions" for the traffic flows is introduced and

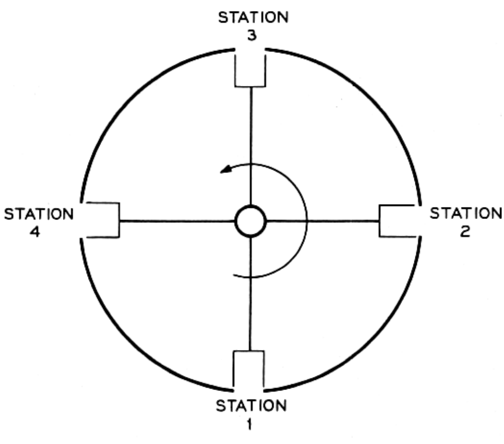

Fig. 1—A schematic description of a four-station Pierce loop.

formulated mathematically. The case of a totally saturated system is analyzed in Section III where the basis is laid for the mathematical development of the load-and-shift algorithm given in Section IV. A detailed description of a Pierce loop for data transmission is given in Section V where the alternating priorities effect due to dominating classes of stations is studied by simulation. This section also discusses the important aspect of the order of stations in the loop. Readers not interested in the mathematical elaborations may skip Section IV.

II. MATHEMATICAL FORMULATION

We assume that the flow direction on the main circular line is counterclockwise. There are n stations, connected to the main line, numbered from 1 to n in counterclockwise increasing order. The segment of main line between the ith and the $(i + 1)$th station, $i = 1, 2, \cdots, n - 1$, is called the ith branch. Similarly, the nth branch is the segment between station n and station 1. Let p_{ij} be the proportion of flow (packets) emerging from station i and destined for station j.

$$\sum_{j=1}^{n} p_{ij} = 1. \tag{1}$$

The $n \times n$ square matrix $P = \{p_{ij}\}$ possesses all the properties of a stochastic matrix. Note, however, that the elements of P are not necessarily probabilities.

The demand at station i is given by λ_i which is the average amount of flow (packets per time unit) generated at the station. The capacity of each branch equals 1, that is, each branch is capable of carrying a maximum flow of one unit. In the schematic description given in Fig. 1 assume that one full revolution takes four time units (in general it will take the number of time units equal to the number of arms) The capacity of each branch will then be one packet per time unit.

The average flow emerging from station i will be denoted by x_i and the average flow in branch i by ρ_i. Clearly

$$0 \leqq x_i \leqq \lambda_i, \quad i = 1, 2, \cdots, n \tag{2}$$

and

$$0 \leqq \rho_i \leqq 1, \quad i = 1, 2, \cdots, n. \tag{3}$$

Let a_{ij} be the proportion of flow emerging from station j and flowing through branch i, then

$$a_{ij} = \sum_{k \in S_{ij}} p_{jk}, \tag{4}$$

where

$$S_{ij} = \begin{cases} (i+1, i+2, \cdots, j) & \text{if} \quad j > i, \\ (i+1, i+2, \cdots, n, 1, 2, \cdots, j) & \text{if} \quad j \leqq i. \end{cases} \tag{5}$$

Note that $a_{i-1j} - a_{ij} = p_{ji} \geqq 0$ for all $j \neq i$ where $i-1$ is defined to equal n when $i = 1$. For $j = i$ we have $a_{ii} = 1$ and $a_{i-1i} = p_{ii} \leqq 1$. In most reasonable applications $p_{ii} = 0$.

The average flow in branch i may now be expressed as a linear function of $X = (x_1, x_2, \cdots, x_n)$.

$$\rho_i = \sum_{j=1}^{n} a_{ij} x_j \leqq 1. \tag{6}$$

Every X which is a feasible solution for the average flows must satisfy relations (2) and (6). The set of all feasible solutions is therefore contained in a convex polyhedral set. A central control could select a particular solution from this set to suit a given objective.

In the circular network suggested by Pierce, however, there is no central control. Rather each station is striving to maximize its own flow onto the main line. For this case we define a stable solution as a solution from which the system will not depart without outside intervention. Suppose then that X^* is a stable solution. Clearly X^* must satisfy relations (2) and (6) and the additional condition that if $\rho_i = \sum_{j=1}^{n} a_{ij} x_j^* < 1$ then $x_i^* = \lambda_i$. To show that this is a necessary condition assume that $\rho_i < 1$ and $x_i^* < \lambda_i$. However, station i strives to maximize its flow and can increase it as long as $x_i^* < \lambda_i$ and $\rho_i < 1$. Therefore, it is not possible that $\rho_i < 1$ and $x_i^* < \lambda_i$. This additional condition is not generally sufficient for assuring that x^* is a stable solution.

Insufficiency is best demonstrated by a simple numerical example. Suppose $n = 4$, $\lambda_1 = \lambda_2 = \lambda_3 = \lambda_4 = 2$ and

$$P = \begin{vmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{vmatrix}.$$

It is easy to verify that $X^0 = (0, 0, 1, 1)$ and $X^+ = (1, 1, 0, 0)$ both satisfy relations (2), (6), and the additional necessary condition as does $X = \alpha X^0 + (1 - \alpha) X^+$ for $0 \leqq \alpha \leqq 1$. For simplicity we assume that the flows generated at the stations are deterministic in nature and each station generates exactly two packets each time unit.

Suppose now that the system (see Fig. 1) has one rotating arm only. In such a case there are only two stable solutions, namely X^0, ($\alpha = 1$) and X^+, ($\alpha = 0$). If the system has two rotating arms, an additional stable solution, $\alpha = \frac{1}{2}$, is added. If the system has k arms, there exist $k + 1$ stable solutions, $\alpha = m/k$, $m = 0, 1, \cdots, k$. The system shall settle for the stable solution $\alpha = m/k$ if at time zero m compartments contain packets from stations 3 and 4 and $k - m$ compartments contain packets from stations 1 and 2. The necessary condition for stability is also a sufficient condition if the flows generated by the stations are continuous (as in the case of nonmixable fluids or small particles such as vehicles and a loop consisting of a pipe or a road). In such a case the set of feasible solutions is identical to the convex polyhedral set given by relations (2) and (6).

We wish to find all solutions which satisfy relations (2), (6), and the necessary stability condition (these will be all the stable solutions in the case of a system with continuous flows). Our problem can be redefined as one of finding all feasible solutions to the following set of equations:

$$x_i + u_i = \lambda_i ,$$

$$\sum_{j=1}^{n} a_{ij}x_j + z_i = 1, \qquad i = 1, 2, \cdots, n. \tag{7}$$

$$u_i z_i = 0, \qquad x_i \geqq 0, \qquad u_i \geqq 0, \qquad z_i \geqq 0.$$

This form resembles a linear complementarity problem,[2] where u_i and z_i are slack variables. A feasible solution to the set of equations (7) is a basic feasible solution to the set of equations (2) and (6) since at least n of the variables must equal zero.

III. COMPLETE SATURATION

For sufficiently large values of λ_i (for example $\lambda_i = 1, i = 1, 2, \cdots, n$), all the branches are saturated and $\rho_i = 1$, $i = 1, 2, \cdots, n$. The set of equations (7) takes the form

$$\sum_{j=1}^{n} a_{ij}x_j = 1, \qquad i = 1, 2, \cdots, n. \tag{8}$$

This can be shown to be equivalent to the set of $(n + 1)$ linear equations.

$$x_i - \sum_{j=1}^{n} p_{ji} x_j = 0, \qquad i = 1, 2, \cdots, n, \tag{9a}$$

$$\sum_{j=1}^{n} x_j \, d_j = b. \tag{9b}$$

The ith equation in (9a) is obtainable by subtracting the ith equation of set (8) from the $(i + 1)$th equation of set (8). Equation (9b) may be selected as any linear combination of equations (8). Note that one equation in (9a) is redundent since $\sum_{j=1}^{n} p_{ij} = 1$ for all i.

Remark: The physical interpretation of equations (9) is that when in complete saturation a station is able to deliver only when receiving. The flow emerging from a given station must equal the flow entering the station from the main line. This equilibrium relation is expressed by the n equations of set (9a). The $(n + 1)$th equation, (9b), expresses the capacity limitation of the branches. If we select $d_j = \sum_{i=1}^{n} a_{ij}$ we have

$$\sum_{j=1}^{n} x_j \, d_j = n, \tag{10}$$

where d_j is the average number of branches (distance) traveled by a packet emerging from station j. Equation (10) states that the average work (in terms of packets times distance) demanded from the system per time unit must equal n, since all branches are saturated and each traverses one packet per time unit. In matrix form we have

$$X^T P = X^T,$$

$$\sum_{i=1}^{n} x_j \, d_j = b, \tag{11}$$

where X^T is the transpose of X, (note that all vectors are defined to be column vectors).

P is a stochastic matrix and, therefore, the problem represented by equations (11) strongly resembles one of determining the steady state probabilities of a finite state space Markov chain. The difference is that in the Markov chain problem $d_j = b$ for all j while in our problem this is not necessarily so. In the following we shall make use of this resemblance.

Definitions:

(*i*) S_i shall be used to abbreviate "station i."
(*ii*) S_j is said to be accessible from S_i, $S_i \rightarrow S_j$, if there exists a

sequence of elements of P such that $P_{ik_1}P_{k_1k_2} \cdots P_{k_{r-1}k_r}P_{k_rj} > 0$. In such a case it is also customary to say that S_i leads to S_j.

(*iii*) If $S_i \rightarrow S_j$ and $S_j \rightarrow S_i$ both stations are said to communicate ($S_i \leftrightarrow S_j$). Clearly if $S_i \leftrightarrow S_j$ and $S_k \leftrightarrow S_j$ then $S_i \leftrightarrow S_k$.

(*iv*) Let $C(i) = \{S_j : S_j \leftrightarrow S_i\}$. Clearly if $S_j \in C(i)$ then $C(j) = C(i)$.

(*v*) A nonempty class of stations, C, is called a communicating class if for some station $S_i \in C$, $C = C(i)$. It follows that two communicating classes are either identical or disjoint.

(*vi*) A communicating class is closed if no station outside the class is accessible from a station in the class.

(*vii*) A closed communicating class shall be called a dominating class or a class of dominating stations. A station not belonging to any dominating class is a dominated station.

From the theory of Markov chains we know that there exists at least one class of dominating stations in a given Pierce loop. However, it is possible that there will be no dominated stations. In such a case all the stations are dominating and may form into one or more dominating classes. It may be shown that the number of dominating classes is one if and only if there exists a station accessible from all other stations in the loop. In the case that dominated stations exist each must lead to at least one dominating station.

Each dominating class of stations is represented by a principal submatrix of P (a dominating submatrix). This submatrix is obtainable by deleting all rows and columns of P corresponding to stations not belonging to the particular dominating class. Similarly, all dominated stations may be represented by one submatrix of P.

Theorem 1. Let B be a $k \times k$ submatrix of P representing a dominating class of stations, then all vectors Y satisfying the equation,

$$Y^T B = Y^T \tag{11a}$$

form a linear space of one dimension. Furthermore, all the elements of Y must have the same sign, i.e., either all positive, or all negative, or all zero.

Proof. Theory of finite Markov chains (e.g., Kemeny and Snell[3]).

Corollary. Assume that the dominating class represented by B is $C = \{S_{i_1}, S_{i_2}, \cdots, S_{i_k}\}$ then there exists a unique vector $Y^* = (y^*_{i_1}, y^*_{i_2}, \cdots, y^*_{i_k})$ satisfying equation (11a) and the scaling equation

$$\sum_{j=1}^{k} y^*_{i_j} d_{i_j} = b. \tag{12}$$

Clearly $Y^* > 0$ and any solution to equation (11a) may be obtained by multiplying Y^* by some real number. We define a vector $X = (x_1, x_2, \cdots, x_n)$ where $x_i = 0$ if $S_i \notin C$, and $x_i = y^*_{i_j}$ if $i = i_j$. The vector X is called a dominating solution to equations (11) corresponding to the matrix B and the class C.

Theorem 2. *Let Q be a $k \times k$ submatrix of P representing dominated stations, then the only solution to $Y^T Q = Y^T$ is $Y = 0$.*

Proof. Theory of finite Markov chains.

Theorem 3. *Suppose that the matrix P contains exactly m dominating submatrices $B_1, B_2, \cdots, B_m$ representing dominating classes $C_1, C_2, \cdots, C_m$ and suppose that X_i is the dominating solution corresponding to B_i, $i = 1, 2, \cdots, m$. X is a solution to equations (8) if and only if*

$$X = \sum_{i=1}^{m} \alpha_i X_i ,$$

and

$$\sum_{i=1}^{m} \alpha_i = 1, \tag{13}$$

where α_i, $i = 1, 2, \cdots, m$ are real numbers.

Proof. This is an immediate result of Theorems 1 and 2.

We conclude that in the case of complete saturation, ($\rho_i = 1$, $i = 1, 2, \cdots, n$), there always exists a nonnegative solution to our problem. If there exists only one dominating class of stations in the loop, there exists a unique solution to equations (8). If there is more than one dominating class in the loop, then there exist infinitely many nonnegative solutions obtainable as convex combinations of the dominating solutions.

IV. THE LOAD-AND-SHIFT ALGORITHM

Returning to the more general case, we describe in this section the load-and-shift procedure for solving the set of equations (7) when $\rho_i \leqq 1$, $i = 1, 2, \cdots, n$. The algorithm is based on the results obtained in the foregoing analysis. Assuming that the capacity of the branches, ϵ, may be varied between 0 and 1, we start with $\epsilon = 0$ and increase it until $\epsilon = 1$. While so doing we simultaneously load the system and obtain the feasible solutions for any given value of $0 \leqq \epsilon \leqq 1$.

We shall start by outlining the procedure for finding just one solution.

Start: $r: = 0$, $\epsilon: = 0$, $P^{(1)}: = P$, $\lambda_i^{(1)}: = \lambda_i$.

Step 1: Increase r by 1. In $P^{(r)}$ find a square submatrix $B^{(r)}$ representing a dominating class of stations $C^{(r)}$, $C^{(r)} = \{S_{i_1}, S_{i_2}, \cdots, S_{i_k}\}$.†

Step 2: Find the unique positive vector $Y^{(r)} = \{y_{i_1}^{(r)}, y_{i_2}^{(r)}, \cdots, y_{i_k}^{(r)}\}$ satisfying the set of equations

$$(Y^{(r)})^T B^{(r)} = (Y^{(r)})^T,$$

$$\sum_{m=1}^{k} y_{i_m}^{(r)} d_{i_m} = 1.$$

To determine the value of d_{i_m} we select an i such that $S_i \varepsilon C^{(r)}$ and let $d_{i_m} = a_{i,i_m}$.

Enlarge $Y^{(r)}$ to the form $X^{(r)} = (x_1^{(r)}, x_2^{(r)}, \cdots, x_n^{(r)})$, where $x_j^{(r)} = 0$ if $S_j \not\varepsilon C^{(r)}$ and $x_j^{(r)} = y_{i_m}^{(r)}$ if $j = i_m$.

Step 3 (Load): Find a number $\Delta^{(r)}$ such that

$$\Delta^{(r)} = \text{Min}\left\{\frac{\lambda_i^{(r)}}{x_i^{(r)}}\right\} = \frac{\lambda_j^{(r)}}{x_j^{(r)}},$$

where the minimization is over all i such that $S_i \varepsilon C^{(r)}$.

If $\Delta^{(r)} \geqq 1 - \epsilon$, set $\Delta^{(r)} = 1 - \epsilon$, set $N = r$, and go to "Last Step." Otherwise increase ϵ by $\Delta^{(r)}$ and continue.

Step 4 (Shift): If $r = n$, set $N = n$ and go to "Last Step." Otherwise construct the $(n - r + 1) \times (n - r + 1)$ square matrix $P^{(r+1)}$ by adding the jth column of $P^{(r)}$ to its $(j + 1)$th column (if j is the last column, it is added to the first one) and then deleting the jth column and the jth row. $\lambda_i^{(r+1)}: = \lambda_i^{(r)} - \Delta^{(r)} x_i^{(r)}$, $i = 1, 2, \cdots, n$. End of rth iteration. Go back to "Step 1."

Last Step: The solution is

$$X = \sum_{r=1}^{N} \Delta^{(r)} X^{(r)}. \tag{14}$$

STOP.

Theorem 4: The procedure described above will always yield a vector $0 \leqq X \leqq \lambda$ in at most n iterations.

Proof: All possible matrices $P^{(r)}$ are stochastic. Therefore there always exists at least one dominating submatrix of $P^{(r)}$ denoted by $B^{(r)}$. Since $X^{(r)} \geqq 0$ and $\Delta^{(r)} \geqq 0$ then $X = \sum_{r=1}^{N} \Delta^{(r)} X^{(r)} \geqq 0$. From the

† A procedure for determining $B^{(r)}$ is described in the Appendix.

algorithm we have $\lambda = \sum_{r=1}^{N} \Delta^{(r)} X^{(r)} + \lambda^{(N)} = X + \lambda^{(N)}$. The algorithm also ensures that $\lambda^{(r)} \geqq 0$, $r = 1, 2, \cdots, N$, and therefore $X \leqq \lambda$. Since P is $n \times n$ the number of iterations cannot exceed n.

Theorem 5 (Existence): X obtained by the Load-and-Shift procedure is a feasible solution to equations (7).

Proof: We enlarge $Y^{(r)}$ by adding zero elements corresponding to columns of $P^{(r)}$ not included in $B^{(r)}$. The enlarged vector, denoted by $Y^{0(r)}$ is a dominating solution to

$$Y^T P^{(r)} = Y^T$$

$$\sum_{j \varepsilon R_r} y_j\, d_j = 1, \tag{15}$$

where R_r is the set of indices of columns included in $P^{(r)}$. These equations are equivalent [see equations (8) and (9)] to

$$\sum_{j \varepsilon R_r} a_{ij}^{(r)} y_j = 1, \qquad i \varepsilon R_r . \tag{16}$$

It is easily verified, by the use of equation (4), that

$$a_{ij}^{(r)} = a_{ij} . \tag{17}$$

The vector $X^{(r)}$ is obtained by adding to $Y^{0(r)}$ zero elements corresponding to columns of P not included in $P^{(r)}$. Therefore

$$\Delta^{(r)} \sum_{j=1}^{n} a_{ij} x_j^{(r)} = \Delta^{(r)}, \quad \text{for all} \quad i \varepsilon R_r . \tag{18}$$

From the definition of a_{ij}, equations (4) and (5), we know that $a_{i-1j} \geqq a_{ij}$ for all j except $j = i$, (where $i - 1 = n$ if $i = 1$). If $i \not\varepsilon R_r$ then $x_i^{(r)} = 0$. It follows that

$$\Delta^{(r)} \sum_{j=1}^{n} a_{ij} x_j^{(r)} \leqq \Delta^{(r)}, \quad \text{for all} \quad i \not\varepsilon R_r . \tag{19}$$

Summing equations (18) and (19) with respect to r and then substituting equation (14) we obtain (for $N < n$)

$$\sum_{j=1}^{n} a_{ij} x_j = \sum_{r=1}^{N} \Delta^{(r)} = 1, \quad \text{if} \quad i \varepsilon R_r \quad \text{for all} \quad r = 1, 2, \cdots, N,$$

$$\sum_{j=1}^{n} a_{ij} x_j \leqq \sum_{r=1}^{N} \Delta^{(r)} = 1, \quad \text{otherwise.} \tag{20}$$

From the algorithm we know that if $i \varepsilon R_r$ for all $r = 1, 2, \cdots, N$ then $x_i \leqq \lambda_i$. Otherwise $x_i = \lambda_i$. This completes the proof for $N < n$.

In the case $N = n$ it is possible that $\sum_{r=1}^{N} \Delta^{(r)} < 1$ (nonheavy traffic). The theorem still holds since $x_i = \lambda_i$, $i = 1, 2, \cdots, n$. In the following we illustrate the use of the algorithm in solving for the flows in a 5-station loop:

Numerical Example 1

$$P = \begin{bmatrix} 0 & 1/2 & 1/2 & 0 & 0 \\ 2/3 & 0 & 1/3 & 0 & 0 \\ 1/4 & 3/4 & 0 & 0 & 0 \\ 1/4 & 1/4 & 1/4 & 0 & 1/4 \\ 1/4 & 0 & 1/2 & 1/4 & 0 \end{bmatrix} \qquad \lambda = \begin{bmatrix} 0.5 \\ 0.4 \\ 0.5 \\ 1.0 \\ 0.8 \end{bmatrix}$$

Table of Results

r	$C^{(r)}$-dominating Class	$x_1^{(r)}$	$x_2^{(r)}$	$x_3^{(r)}$	$x_4^{(r)}$	$x_5^{(r)}$	$\Delta^{(r)}$	ϵ
1	S_1, S_2, S_3.	0.6000	0.7000	0.5444	0	0	0.5714	0.5714
2	S_1, S_3.	0.2500	0	1.0000	0	0	0.1953	0.7667
3	S_1, S_4, S_5.	0.3123	0	0	1.0000	0.2500	0.2333	1.0000

The procedure terminated in three iterations yielding a solution:

$$x_1 = 0.4646$$

$$x_2 = 0.4000 = \lambda_2$$

$$x_3 = 0.5000 = \lambda_3$$

$$x_4 = 0.2333$$

$$x_5 = 0.0508$$

Actually the algorithm finds the value of one variable in each iteration. When all variables equal to their respective λ_i's have been determined, the algorithm finds in one iteration the values of all the remaining variables. It is therefore advisable to test the possible solution $X = \lambda$ beforehand.

We note that in the example there is a single dominating class of stations in each iteration (i.e., each matrix $P^{(r)}$ has only one dominating submatrix $B^{(r)}$). It will be shown later that if P has only one dominating submatrix then there exists a unique solution to equations (7). In the general case, however, $P^{(r)}$ may have more than one dominating submatrix. This may give rise to the existence of infinitely many solutions. We would now like to improve the algorithm in order to be able to determine all feasible solutions.

Theorem 6: If $P^{(r)}$ has exactly one dominating submatrix, say $B^{(r)}$, then $P^{(r+1)}$ will also have exactly one dominating submatrix.

Proof: Suppose i is such that $S_i \varepsilon C^{(r)}$, then $S_j \rightarrow S_i$ for all j. We apply the shift operation from i to k, (adding column i to column k and then deleting row and column i), thus creating $P^{(r+1)}$. Clearly now $S_j \rightarrow S_k$ for all j. However, if there exists a station accessible from all other stations there exists exactly one dominating class in the loop. Furthermore $C^{(r+1)} = C(k)$.

Corollary: If P has one dominating submatrix so will $P^{(r)}$, $r = 1, 2, \cdots, N$.

Theorem 7: If $P^{(r)}$ has $m > 1$ dominating submatrices, $P^{(r+1)}$ will have either $(m - 1)$ or m dominating submatrices.

Proof (outline): Let $B_1^{(r)}, B_2^{(r)}, \cdots, B_m^{(r)}$ be the m dominating submatrices of $P^{(r)}$ and let $C_1^{(r)}, C_2^{(r)}, \cdots, C_m^{(r)}$ be the corresponding classes of dominating stations. $Q^{(r)}$ is the submatrix representing dominated stations. We apply the shift operation from i to k to obtain $P^{(r+1)}$. Without loss of generality we assume that $S_i \varepsilon C_1^{(r)}$. For S_k one of three alternatives must be true:

(*i*) $S_k \varepsilon C_1^{(r)}$. In this case a dominating class will be formed, containing some or all the remaining stations of $C_1^{(r)}$. Note that S_k must be in the newly formed dominating class since it is accessible from all remaining stations of $C_1^{(r)}$. Those stations of $C_1^{(r)}$ which are not included in the newly formed dominating class turn into dominated stations. The matrix $P^{(r+1)}$ will then have m dominating submatrices, namely $B_2^{(r)}, B_3^{(r)}, \cdots, B_m^{(r)}$ and a newly formed one.

(*ii*) $S_k \varepsilon C_l^{(r)}$, $l \neq 1$. In this case every station in $C_l^{(r)}$ is accessible from any remaining station of $C_1^{(r)}$. Therefore all remaining stations of $C_1^{(r)}$ become nondominating, and $P^{(r+1)}$ will have $m - 1$ dominating matrices, namely $B_2^{(r)}, B_3^{(r)}, \cdots, B_m^{(r)}$.

(*iii*) S_k is nondominating. S_k must lead to at least one dominating class. If it leads to any dominating class other than $B_1^{(r)}$ then this case becomes the same as (*ii*). If S_k leads to $B_1^{(r)}$ only, then a new dominating class is formed. This class includes all stations accessible from S_k. Remaining stations of $B_1^{(r)}$ which are not accessible from S_k

become nondominating. $P^{(r+1)}$ will have m dominating submatrices, namely $B_2^{(r)}$, $B_3^{(r)}$, $\cdots$, $B_m^{(r)}$, and a newly formed one.

This completes the outline of the proof. It is important to note that a dominating submatrix of $P^{(r)}$, say $B_l^{(r)}$, will be a dominating submatrix of $P^{(r+1)}$ if $S_i \notin C_l^{(r)}$.

An Outline of the Complete Version of the Load-and-Shift Algorithm

In the procedure for finding just one solution, outlined in the preceding, we apply one load and one shift operation in each iteration. In the complete version of the algorithm we need to apply several such operations in each iteration. To eliminate possible confusion the superscript denoting the iteration number will be placed at the upper left side. Thus, for example, ${}^{(r)}P$ is the stochastic matrix remaining by the beginning of the rth iteration.

Definitions: Suppose ${}^{(r)}P$ has m_r dominating submatrices, ${}^{(r)}B_1$, ${}^{(r)}B_2$, $\cdots$, ${}^{(r)}B_{m_r}$, representing classes ${}^{(r)}C_1$, ${}^{(r)}C_2$, $\cdots$, ${}^{(r)}C_{m_r}$. The matrix ${}^{(r)}P$ was obtained from ${}^{(1)}P = P$ by executing some sequence of load and shift operations. In order to keep track of these operations we relate to ${}^{(r)}B_i$ a set ${}^{(r)}E_i$ containing labels. If the label ${}^{(k)}\alpha_j \in {}^{(r)}E_i$ we know that ${}^{(r)}B_i$ cannot be obtained unless the appropriate shift operation is applied to ${}^{(k)}B_j$. Thus, when ${}^{(r)}E_i$, $i = 1, 2, \cdots, m_r$, are given, the exact sequence of load and shift operations that led us to ${}^{(r)}P$ is known.

The n dimensional vector ${}^{(r)}\lambda$ denotes the remaining unutilized flows at the n stations of the loop. The m_r dimensional vector ${}^{(r)}\epsilon = ({}^{(r)}\epsilon_1, {}^{(r)}\epsilon_2, \cdots, {}^{(r)}\epsilon_{m_r})$ describes the amount of branch capacity utilization. Thus ${}^{(r)}\epsilon_i$ is the amount of branch capacity utilized by the sequence of load and shift operations resulting in ${}^{(r)}B_i$.

For each ${}^{(r)}B_i$ it is possible to determine the dominating solution ${}^{(r)}X_i$ and the quantity ${}^{(r)}\Delta_i$ in the manner described in steps 2 and 3 of the procedure outlined previously.

We say that ${}^{(r)}B_i \to {}^{(r)}B_j$ if when applying a shift operation on the appropriate column of ${}^{(r)}B_i$ all remaining stations of ${}^{(r)}C_i$ lead to stations of ${}^{(r)}C_j$. If ${}^{(r)}C_i$ consists of a single station then ${}^{(r)}B_i \to {}^{(r)}B_j$ if the shift is from this station to a station in ${}^{(r)}C_j$ or to a dominated station leading to ${}^{(r)}C_j$. The submatrices ${}^{(r)}B_i$, $i = 1, 2, \cdots, m_r$, divide into three types: transient, terminal, and ring members.

Ring: The set $R = \{{}^{(r)}B_{i_1}, {}^{(r)}B_{i_2}, \cdots, {}^{(r)}B_{i_k}\}$ is called a ring if it satisfies three conditions:

(*i*) ${}^{(r)}B_{i_1} \rightarrow {}^{(r)}B_{i_2} \rightarrow \cdots \rightarrow {}^{(r)}B_{i_k} \rightarrow {}^{(r)}B_{i_1}$,

(*ii*) ${}^{(r)}B_{i_j}$, $j = 1, 2, \cdots, k$, does not lead to any dominating submatrix outside of the same ring,

(*iii*) $1 - \sum_{j=1}^{k} ({}^{(r)}\epsilon_{i_j} - {}^{(r)}\Delta_{i_j}) > 0$.

Terminal: (*i*) If conditions *i* and *ii* above are satisfied while condition *iii* is violated the matrices ${}^{(r)}B_{i_1}, {}^{(r)}B_{i_2}, \cdots, {}^{(r)}B_{i_k}$ are called terminal matrices.

(*ii*) If ${}^{(r)}\Delta_i \geqq 1 - {}^{(r)}\epsilon_i$ then ${}^{(r)}B_i$ is terminal.

(*iii*) If ${}^{(r)}B_i$ is terminal and ${}^{(r)}B_j \rightarrow {}^{(r)}B_i$ then ${}^{(r)}B_j$ is terminal too.

Transient: ${}^{(r)}B_i$ is called transient if it does not belong to a ring and is not terminal.

There is a strong similarity between the notion of a ring and the notion of a dominating class. To identify rings one may use essentially the same technique proposed in the Appendix for identification of dominating classes.

It is important to observe that when applying the appropriate shift operations to all members of R, what remains of R will form one dominating submatrix and possibly some columns and rows corresponding to dominated stations will be left out. This property follows from Theorem 7.

The following is an outline of the algorithm:

Start: $r: = 0$, ${}^{(1)}P: = P$, ${}^{(1)}\lambda: = \lambda$, ${}^{(1)}\epsilon: = 0$, ${}^{(1)}E_i : = \phi$.

Step 1: $r: = r + 1$.

Find all the dominating submatrices of ${}^{(r)}P$ denoted by ${}^{(r)}B_1$, ${}^{(r)}B_2, \cdots, {}^{(r)}B_{m_r}$. Calculate the numerical values of ${}^{(r)}\Delta_i$ and ${}^{(r)}X_i$, corresponding to ${}^{(r)}B_i$, ${}^{(r)}\lambda$, and ${}^{(r)}\epsilon_i$, $i = 1, 2, \cdots, m_r$. Find all the rings of ${}^{(r)}P$ denoted by ${}^{(r)}R_1, {}^{(r)}R_2, \cdots, {}^{(r)}R_{k_r}$. Identify the terminal and transient dominating submatrices of ${}^{(r)}P$. If all dominating submatrices of ${}^{(r)}P$ are terminal go to LAST STEP. If the number of columns of ${}^{(r)}P$ equals the number of its dominating submatrices who form into one ring, go to LAST STEP 1. Otherwise ${}^{(r)}\Delta_i : = 0$ for all ${}^{(r)}B_i$ not belonging to a ring.

Step 2: Execute the appropriate shift operations to all ${}^{(r)}B_i$ belonging to rings and obtain ${}^{(r+1)}P$. Note that in ${}^{(r+1)}P$ there will be one dominating submatrix corresponding to each ring of ${}^{(r)}P$ and, in addition, all terminal and transient dominating submatrices of ${}^{(r)}P$ will be dominating in ${}^{(r+1)}P$.

Step 3:

$${}^{(r+1)}\lambda := {}^{(r)}\lambda - \sum_{i=1}^{m_r} {}^{(r)}\Delta_i\, {}^{(r)}X_i\,.$$

$${}^{(r+1)}\epsilon_i := \sum_{\{j:{}^{(r)}B_j \varepsilon {}^{(r)}R_i\}} ({}^{(r)}\epsilon_j + {}^{(r)}\Delta_j), \qquad i = 1, 2, \cdots, k_r\,.$$

${}^{(r+1)}\epsilon_i := {}^{(r)}\epsilon_j$ where ${}^{(r+1)}B_i = {}^{(r)}B_j\,, i = k_r + 1, k_r + 2, \cdots, m_{r+1}\,.$

$${}^{(r+1)}E_i := \bigcup_{\{j:{}^{(r)}B_j \varepsilon {}^{(r)}R_i\}} ({}^{(r)}E_j \cup {}^{(r)}\alpha_j), \qquad i = 1, 2, \cdots, k_r\,.$$

${}^{(r+1)}E_i := {}^{(r)}E_j$ where ${}^{(r+1)}B_i = {}^{(r)}B_j\,, i = k_r + 1, k_r + 2, \cdots, m_{r+1}$. GO TO STEP 1.

Last Step 1: The unique solution is $X = \lambda$. *STOP.*

Last Step: $N := r$.

All the solutions are given by

$$X = \sum_{r=1}^{N} \sum_{i=1}^{m_r} {}^{(r)}\alpha_i\, {}^{(r)}X_i\,, \tag{21}$$

$$\sum_{r=1}^{N} \sum_{i=1}^{m_r} {}^{(r)}\alpha_i = 1, \qquad 0 \leqq {}^{(r)}\alpha_i \leqq {}^{(r)}\Delta_i\,, \qquad {}^{(r)}\alpha_i = 0, \qquad r \geqq 2,$$

$$\text{unless} \quad {}^{(k)}\alpha_j = {}^{(k)}\Delta_j \quad \text{for all} \quad {}^{(k)}\alpha_j \,\varepsilon\, {}^{(r)}E_i\,. \quad \textit{STOP}.$$

Note that it is advisable to test the possible solution $X = \lambda$ beforehand.

Theorem 8: The load-and-shift procedure will yield vectors $0 \leqq X \leqq \lambda$ in at most n iterations. Each such vector, given by equation (21), is a feasible solution to set of equations (7).

Proof: The proof is practically identical to the proofs of Theorems 4 and 5.

The use of the algorithm is illustrated by a 10-station numerical example.

Numerical Example 2

$$P = {}^{(1)}P = \begin{array}{c} \begin{array}{cccccccccc} S_1 & S_2 & S_3 & S_4 & S_5 & S_6 & S_7 & S_8 & S_9 & S_{10} \end{array} \\ \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/2 & 1/2 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1/4 & 0 & 0 & 3/4 & 0 & 0 & 0 & 0 & 0 \\ 0 & 3/4 & 0 & 1/4 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1/4 & 0 & 0 & 3/4 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1/4 & 0 & 0 & 3/4 \\ 0 & 0 & 0 & 0 & 0 & 1/4 & 1/4 & 0 & 1/2 & 0 \end{bmatrix} \end{array}$$

$$\lambda = {}^{(1)}\lambda = \begin{bmatrix} 0.10 \\ 0.46 \\ 0.20 \\ 0.20 \\ 0.65 \\ 0.50 \\ 0.45 \\ 0.60 \\ 0.50 \\ 1.00 \end{bmatrix}$$

We first observe that ${}^{(1)}P$ has $m_1 = 3$ dominating submatrices

$${}^{(1)}B_1 = \begin{array}{c} \begin{array}{cc} S_1 & S_3 \end{array} \\ \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \end{array}, \quad {}^{(1)}B_2 = \begin{array}{c} \begin{array}{ccc} S_2 & S_4 & S_5 \end{array} \\ \begin{bmatrix} 0 & 1/2 & 1/2 \\ 1/4 & 0 & 3/4 \\ 3/4 & 1/4 & 0 \end{bmatrix} \end{array}, \quad {}^{(1)}B_3 = \begin{array}{c} \begin{array}{cc} S_6 & S_8 \end{array} \\ \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \end{array}.$$

The calculated results for the first iteration are summarized in the following table.

Results for First Iteration

${}^{(1)}C_1 = \{S_1, S_3\}$	${}^{(1)}X_1 = (1\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0)$	${}^{(1)}\Delta_1 = 0.1$	${}^{(1)}B_1 \to {}^{(1)}B_2$	Tr	${}^{(1)}\epsilon_1 = 0$
${}^{(1)}C_2 = \{S_2, S_4, S_5\}$	${}^{(1)}X_2 = (0\ \frac{26}{33}\ 0\ \frac{20}{33}\ \frac{28}{33}\ 0\ 0\ 0\ 0\ 0)$	${}^{(1)}\Delta_2 = 0.33$	${}^{(1)}B_2 \to {}^{(1)}B_2$		${}^{(1)}\epsilon_2 = 0$
${}^{(1)}C_3 = \{S_6, S_8\}$	${}^{(1)}X_3 = (0\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 0)$	${}^{(1)}\Delta_3 = 0.50$	${}^{(1)}B_3 \to {}^{(1)}B_3$		${}^{(1)}\epsilon_3 = 0$

Tr—Transient; Te—Terminal.

$$ {}^{(1)}R_1 = \{{}^{(1)}B_2\}, \qquad {}^{(1)}R_2 = \{{}^{(1)}B_3\}, $$

$$ {}^{(1)}E_1 = \phi, \qquad {}^{(1)}E_2 = \phi, \qquad {}^{(1)}E_3 = \phi. $$

Note: ${}^{(1)}\Delta_1$ will be set to equal zero before continuing. Shift operations are performed from column 4 to 5 and from 6 to 7 in ${}^{(1)}P$. The resulting 8×8 matrix, ${}^{(2)}P$, has three dominating submatrices.

$$ {}^{(2)}B_1 = \begin{array}{cc} S_2 & S_5 \\ \end{array} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad {}^{(2)}B_2 = \begin{array}{ccc} S_1 & S_9 & S_{10} \end{array} \begin{bmatrix} 1/4 & 3/4 & 0 \\ 1/4 & 0 & 3/4 \\ 1/2 & 1/2 & 0 \end{bmatrix}, \quad {}^{(2)}B_3 = {}^{(1)}B_1 $$

$$ {}^{(2)}\lambda = (0.10\quad 0.20\quad 0.20\quad 0\quad 0.37\quad 0\quad 0.45\quad 0.10\quad 0.50\quad 1.00) $$

Results for Second Iteration

${}^{(2)}C_1 = \{S_2, S_5\}$	${}^{(2)}X_1 = (0\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0)$	${}^{(2)}\Delta_1 = 0.20$	${}^{(2)}B_1 \to {}^{(2)}B_3$	Te	${}^{(2)}\epsilon_1 = 0.33$
${}^{(2)}C_2 = \{S_7, S_9, S_{10}\}$	${}^{(2)}X_2 = (0\ 0\ 0\ 0\ 0\ 0\ \frac{20}{29}\ 0\ \frac{24}{29}\ \frac{18}{29})$	${}^{(2)}\Delta_2 = 0.50$	${}^{(2)}B_2 \to {}^{(2)}B_2$		${}^{(2)}\epsilon_2 = 0.50$
${}^{(2)}C_3 = \{S_1, S_3\}$	${}^{(2)}X_3 = (1\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0)$	${}^{(2)}\Delta_3 = 0.10$	${}^{(2)}B_3 \to {}^{(2)}B_1$		${}^{(2)}\epsilon_3 = 0$

$$^{(2)}R_1 = \{^{(2)}B_1 \quad ^{(2)}B_3\},$$

$$^{(2)}E_1 = \{^{(1)}\alpha_2\}, \qquad ^{(2)}E_2 = \{^{(1)}\alpha_3\}, \qquad ^{(2)}E_3 = \phi.$$

Note: $^{(2)}\Delta_2$ will be set to equal zero before continuing. Shift operations are performed from column 1 to 2 and from column 2 to 3 in $^{(2)}P$. The resulting 6 × 6 matrix, $^{(3)}P$, has two dominating submatrices.

$$\overset{S_3}{^{(3)}B_1 = [1]}, \qquad ^{(3)}B_2 = {^{(2)}B_2}.$$

$$^{(3)}\lambda = (0 \quad 0 \quad 0.10 \quad 0 \quad 0.17 \quad 0 \quad 0.45 \quad 0.10 \quad 0.50 \quad 1.00)$$

Results for Third Iteration

$^{(3)}C_1 = \{S_3\}$	$^{(3)}X_1 = (0 \; 0 \; 1 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0)$	$^{(3)}\Delta_1 = 0.10$	$^{(3)}B_1 \rightarrow {^{(3)}B_4}$	Te	$^{(3)}\epsilon_1 = 0.63$
$^{(3)}C_2 = \{S_7, S_9, S_{10}\}$	$^{(3)}X_2 = (0 \; 0 \; 0 \; 0 \; 0 \; 0 \; \frac{20}{29} \; 0 \; \frac{24}{29} \; \frac{18}{29})$	$^{(3)}\Delta_2 = 0.50$	$^{(3)}B_2 \rightarrow {^{(3)}B_2}$		$^{(3)}\epsilon_2 = 0.50$

$$^{(3)}R_1 = \{^{(3)}B_1\},$$

$$^{(3)}E_1 = \{^{(1)}\alpha_2, {^{(2)}\alpha_1}, {^{(2)}\alpha_3}\}, \qquad ^{(3)}E_2 = \{^{(1)}\alpha_3\}.$$

Note: $^{(3)}\Delta_2$ will be set to equal zero before continuing. Shift operation is performed from column 3 to 5 in $^{(3)}P$. The resulting 5 × 5 matrix $^{(4)}P$ has two dominating submatrices

$$\overset{S_5}{^{(4)}B_1 = [1]}, \qquad ^{(4)}B_2 = {^{(3)}B_2}.$$

$$^{(4)}\lambda = (0 \quad 0 \quad 0 \quad 0 \quad 0.17 \quad 0 \quad 0.45 \quad 0.10 \quad 0.50 \quad 1.00)$$

Results for Fourth (Final) Iteration

${}^{(4)}C_1 = \{S_5\}$	${}^{(4)}X_1 = (0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0)$	${}^{(4)}\Delta_1 = 0.17$	${}^{(4)}B_1 \to {}^{(4)}B_2$	Te	${}^{(4)}\epsilon_1 = 0.73$
${}^{(4)}C_2 = \{S_7, S_9, S_{10}\}$	${}^{(4)}X_2 = (0\ 0\ 0\ 0\ 0\ 0\ \tfrac{20}{29}\ 0\ \tfrac{24}{29}\ \tfrac{18}{29})$	${}^{(4)}\Delta_2 = 0.50$	${}^{(4)}B_2 \to {}^{(4)}B_2$	Te	${}^{(4)}\epsilon_2 = 0.50$

$$ {}^{(4)}E_1 = \{{}^{(1)}\alpha_2\,,\ {}^{(2)}\alpha_1\,,\ {}^{(2)}\alpha_3\,,\ {}^{(3)}\alpha_1\}, \quad {}^{(4)}E_2 = \{{}^{(1)}\alpha_3\}. $$

The iterative procedure terminates after completing the fourth iteration since all dominating submatrices of ${}^{(4)}P$ are terminal.

The set of all feasible solutions to the 10-station problem is

$$ X = {}^{(4)}\alpha_1{}^{(4)}X_1 + {}^{(4)}\alpha_2{}^{(4)}X_2 + {}^{(3)}\alpha_1{}^{(3)}X_1 + {}^{(2)}\alpha_1{}^{(2)}X_1 + {}^{(2)}\alpha_3{}^{(2)}X_3 + {}^{(1)}\alpha_2{}^{(1)}X_2 + {}^{(1)}\alpha_3{}^{(1)}X_3\,, $$

where

$$ {}^{(4)}\alpha_1 + {}^{(4)}\alpha_2 + {}^{(3)}\alpha_1 + {}^{(2)}\alpha_1 + {}^{(2)}\alpha_3 + {}^{(1)}\alpha_2 + {}^{(1)}\alpha_3 = 1, $$

and

$$ 0 \leqq {}^{(4)}\alpha_1 \leqq 0.7, \quad {}^{(4)}\alpha_1 = 0 \text{ unless } {}^{(3)}\alpha_1 = 0.10. $$
$$ 0 \leqq {}^{(4)}\alpha_2 \leqq 0.50, \quad {}^{(4)}\alpha_2 = 0 \text{ unless } {}^{(1)}\alpha_3 = 0.50. $$
$$ 0 \leqq {}^{(3)}\alpha_1 \leqq 0.10, \quad {}^{(3)}\alpha_1 = 0 \text{ unless } {}^{(2)}\alpha_1 = 0.20 \text{ and } {}^{(2)}\alpha_3 = 0.10. $$
$$ 0 \leqq {}^{(2)}\alpha_1 \leqq 0.20, \quad {}^{(2)}\alpha_1 = 0 \text{ unless } {}^{(1)}\alpha_2 = 0.33. $$
$$ 0 \leqq {}^{(2)}\alpha_3 \leqq 0.10. $$
$$ 0 \leqq {}^{(1)}\alpha_2 \leqq 0.33. $$
$$ 0 \leqq {}^{(1)}\alpha_3 \leqq 0.50. $$

It remains to show that the load-and-shift algorithm will yield all the feasible solutions to equations (7).

Theorem 9: If $X^0 = (x_1^0, x_2^0, \cdots, x_n^0)$ is a feasible solution to equations (7), then

$$\lambda_i \geqq x_i^0 \geqq \text{Min} \left\{ \sum_{j=1}^{N} p_{ji} x_j^0 , \lambda_i \right\}. \tag{22}$$

Proof: The left-side inequality of (1) is part of equations (7). For the right-side inequality, suppose first that $x_i^0 < \lambda_i$, then

$$\sum_{j=1}^{n} a_{ij} x_j^0 = 1 \quad \text{and} \quad \sum_{j=1}^{n} a_{i-1j} x_j^0 \leqq 1.$$

Taking the difference we obtain

$$\sum_{j=1}^{n} (a_{ij} - a_{i-1j}) x_j^0 \geqq 0. \tag{23}$$

From the definition of a_{ij} we have that

$$a_{ij} - a_{i-1j} = -p_{ji}, \quad \text{for all} \quad j \neq i,$$

and

$$a_{ii} - a_{i-1i} = 1 - p_{ii}.$$

Therefore

$$x_i^0 \geqq \sum_{j=1}^{n} p_{ji} x_j^0 . \tag{24}$$

It follows then that $x_i^0 = \lambda_i$ if $\sum_{j=1}^{n} p_{ji} x_j^0 \geqq \lambda_i$ and therefore

$$\lambda_i \geqq x_i^0 \geqq \text{Min} \left\{ \sum_{j=1}^{n} p_{ji} x_j^0 , \lambda_i \right\}.$$

Theorem 10: Let X^0 be a solution to equations (7), and assume that $\lambda > 0$, then

(*i*) *If $x_j^0 = 0$, then $x_i^0 = 0$ for all i such that $S_i \rightarrow S_j$.*
(*ii*) *If $x_j^0 > 0$, then $x_i^0 > 0$ for all i such that $S_j \rightarrow S_i$.*

Proof: If $S_j \rightarrow S_i$ there exists a sequence $p_{j,k_1}, p_{k_1,k_2}, \cdots, p_{k_{r-1},k_r}, p_{k_r,i}$ whose product is positive. Suppose $x_j^0 > 0$, then min $(\sum_{m=1}^{n} p_{mk_1} x_m^0, \lambda_{k_1}) \geqq p_{jk_1} x_j^0 > 0$. From relation (22) it follows that $x_{k_1}^0 > 0$ and similarly $x_{k_2}^0 > 0, \cdots, x_i^0 > 0$. This proves *ii*. The proof of *i* is immediate.

Remark: The assumption $\lambda > 0$ is not restrictive. If $\lambda_i = 0$, we apply a shift operation to the *i*th column of P.

Theorem 11: Let $B^{(1)}$ be a dominating submatrix of P and assume, without loss of generality that $C^{(1)} = \{S_1, S_2, \cdots, S_k\}$ is the class of dominating stations represented by $B^{(1)}$. $\Delta^{(1)}$ and $X^{(1)}$ are then uniquely defined. Suppose X^0 is a feasible solution to set of equations (7) then

Either: (*i*) $x_i^0 = \alpha x_i^{(1)}, i = 1, 2, \cdots, k, 0 \leqq \alpha \leqq \Delta^{(1)}$.
or: (*ii*) $x_i^0 \geqq \Delta^{(1)} x_i^{(1)}, i = 1, 2, \cdots, k$, *where the inequality is strict for at least one value of i.*
(*iii*) *If i is the case then* $x_j^0 = 0$ *for all* $S_j \notin C^{(1)}$ *and leading to* $C^{(1)}$. *If there exists such an* S_j *and* $x_j^0 > 0$, *then ii is the case.*

Proof: Since $B^{(1)}$ is a dominating submatrix then $\sum_{j=1}^{k} p_{ij} = 1$ for $i = 1, 2, \cdots, k$.

(*i*) We assume that $x_i^0 < \lambda_i$ for $i = 1, 2, \cdots, k$ and then sum the first k inequalities (24) to obtain

$$\sum_{i=1}^{k} x_i^0 \geqq \sum_{i=1}^{k} x_i^0 + \sum_{j=k+1}^{n} x_j^0 \sum_{i=1}^{k} p_{ji} .$$

For this relation to hold it is necessary that $x_j^0 \sum_{i=1}^{k} p_{ji} = 0$ for all $j \geqq k + 1$. If $C^{(1)}$ is not accessible from S_j then $\sum_{i=1}^{k} p_{ji} = 0$, otherwise either $\sum_{i=1}^{k} p_{ji} > 0$ and therefore x_j^0 must equal zero, or $S_j \rightarrow S_m$, $m \geqq k + 1$, and $\sum_{i=1}^{k} p_{mi} > 0$ and therefore x_m^0 must equal zero. From Theorem 10 we know that in this event $x_j^0 = 0$. Concluding then that $x_j^0 = 0$ for all $\{j: S_j \notin C^{(1)}\}$ and $S_j \rightarrow C^{(1)}$ we obtain [in a manner similar to the one used in obtaining relation (24)]

$$x_i^0 = \sum_{j=1}^{k} p_{ji} x_j^0 , \qquad i = 1, 2, \cdots, k,$$

yielding

$$x_i^0 = \alpha x_i^{(1)}, \quad i = 1, 2, \cdots, k.$$

Since $0 \leqq x_i^0 < \lambda_i$ we have that $0 \leqq \alpha < \Delta^{(1)}$. It also follows that if there exists $S_j \notin C^{(1)}$ and leading to $C^{(1)}$ and $x_j^0 > 0$ then there exists at least one value of i, $i = 1, 2, \cdots, k$, for which $x_i^0 = \lambda_i$.

(*ii*) Suppose there exists a value of i, $i = 1, 2, \cdots, k$, such that $x_i^0 \geqq \Delta^{(1)} x_i^{(1)}$. From Theorem 10 it follows that $x_i^0 > 0$ for all $i = 1, 2, \cdots, k$. We select a number α, $\Delta^{(1)} > \alpha > 0$, such that

$$\operatorname*{Min}_{i=1,2,\cdots,k} \{x_i^0 - \alpha x_i^{(1)}\} = x_{i_m}^0 - \alpha x_{i_m}^{(1)} = 0.$$

Clearly if such an α exists then $x_{i_m}^0 < \Delta^{(1)} x_{i_m}^{(1)}$ and from relation (24) we have

$$x^0_{i_m} \geqq \sum_{j=1}^{n} p_{j i_m} x^0_j . \tag{25}$$

On the other hand we have

$$x^0_{i_m} = \alpha x^{(1)}_{i_m} = \sum_{j=1}^{k} p_{j i_m} \alpha x^{(1)}_j . \tag{26}$$

Subtracting equation (26) from (25) yields

$$0 = \sum_{j=1}^{k} p_{j i_m}(x^0_j - \alpha x^{(1)}_j) + \sum_{j=k+1}^{n} p_{j i_m} x^0_j .$$

Since $S_{i_m} \varepsilon C^{(1)}$ there exists a value of j, $j \neq i_m$ and $S_j \varepsilon C^{(1)}$, such that $p_{j i_m} > 0$. It follows then that $x^0_i = \alpha x^{(1)}_i$ for $i = 1, 2, \cdots, m$. This is a contradiction, which completes our proof.

Corollary: Let ${}^{(1)}B_1, {}^{(1)}B_2, \cdots, {}^{(1)}B_{m_1}$ *be all the dominating submatrices of* $P = {}^{(1)}P$, *and let* ${}^{(1)}C_1, {}^{(1)}C_2, \cdots, {}^{(1)}C_{m_1}$ *be the corresponding dominating classes. If* X^0 *is a solution to equations (7) then either*

$$X^0 = \sum_{i=1}^{m_1} {}^{(1)}\alpha_i \, {}^{(1)}X_i , \qquad \sum_{i=1}^{m_1} {}^{(1)}\alpha_i = 1, \qquad 0 \leqq {}^{(1)}\alpha_i \leqq {}^{(1)}\Delta_i , \tag{27}$$

or there exists at least one class of dominating stations, say ${}^{(1)}C_1$, *such that* $X^0 \geqq {}^{(1)}\Delta_i {}^{(1)}X_i$.

Theorem 12: If X^0 *is a feasible solution to equations (7) it is obtainable by the load-and-shift algorithm.*

Proof: If X^0 is given by equation (27), it is obviously obtainable after one iteration of the algorithm. Otherwise there exists a class of dominating stations, say ${}^{(1)}C_i$, such that $X^0 \geqq {}^{(1)}\Delta_i {}^{(1)}X_i$. We execute a shift operation involving the submatrix ${}^{(1)}B_i$ and obtain a stochastic matrix $P^{(2)}$ (note that $P^{(2)} \neq {}^{(2)}P$). We let $X^{0(2)} = X^0 - {}^{(1)}\Delta_i {}^{(1)}X_i$ and $\lambda^{(2)} = \lambda - {}^{(1)}\Delta_i {}^{(1)}X_i$. Clearly $X^{0(2)}$ is a feasible solution to equations (7) when using as parameters $P^{(2)}$ and $\lambda^{(2)}$ and replacing the right side by $1 - {}^{(1)}\Delta_i$. It is possible therefore to proceed with a sequence of load-and-shift operations until X^0 is obtained. Since the algorithm takes into account all possible sequences of load-and-shift operations, X^0 is contained in the set of solutions given by equation (21).

Corollary (uniqueness): If P has exactly one dominating matrix then there exists a unique feasible solution to equations (7). Note, however, *that this is not a necessary condition for uniqueness. It is possible that P will have several dominating submatrices and the solution will be unique.*

V. DISCUSSION

The heavy traffic assumption enables us to regard the system as a deterministic one. The analysis of the deterministic flows shows that stations tend to "band" into classes with the ability of dominating the system and preventing other stations from using it.

This undesirable property, which can be eliminated by exercising appropriate control, also may affect the stochastic behavior of the system when heavy traffic conditions do not exist. One can imagine two classes of stations competing for domination of the system. Since traffic is not heavy, all the stations in the system are able to deliver their messages in finite time. Nevertheless, when one dominating class controls the system, it will prevent other stations from using the belt line until the queue at one of the stations belonging to this class becomes empty. At that moment the competing dominating class is able to take over and prevent other stations from using the belt line. This may result in a situation of alternating priorities (see Ref. 4) where, while one class is served, the queues at the competing class build up. While average queue sizes may not be strongly affected, the strong fluctuations in queue lengths may be undesirable. This possibility has been explored numerically by the use of a digital simulator.[5] The operating principles of the simulated system will be explained with the aid of Fig. 2.

Each of the stations is represented by a B-box. A packet coming out of a station is first multiplexed on the line by the B-box, provided the line is free, and then is passed from B-box to B-box until its destination is reached. At each B-box on the way the address of the packet is examined. At the particular B-box of destination the packet is taken off the line. The main function of the A-box, shown in Fig. 2, is synchronization of the loop.

Assume that the packets are made of L bits each and the address is given in the first k bits of the packet. A time unit in this system is the time it takes to multiplex a bit on the main line by a B-box. Assume also that the traveling time from one B-box to the adjacent one is zero. Station i is allowed to start sending a packet at times $mL + ik$, $m = 0, 1, 2, \cdots$, providing that the main line is free. The A-box is a buffer. Bits coming out of the B-box of station n accumulate in the A-box. Suppose at time k B-box 1 starts sending a packet. At time

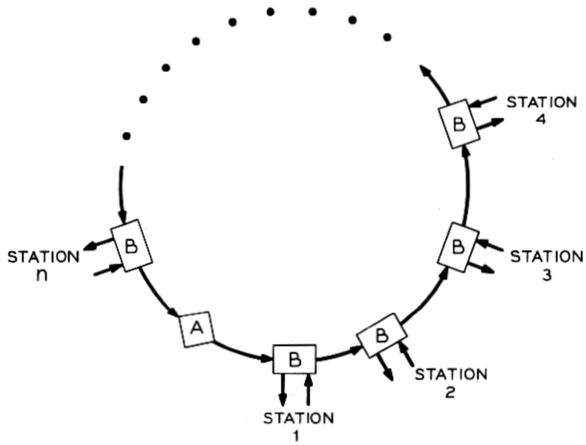

Fig. 2—A schematic description of a loop with n stations.

$k + 1$ the first bit reaches B-box 2 and is delayed there until time $2k$ when the whole address of the packet has been received. If the packet is addressed to station 2, it will be taken out and at the same time B-box 2 can start sending out its own packet. If the packet is not addressed to station 2, it will be sent from B-box 2 to B-box 3 where the same process will take place starting at time $3k$. Bits arriving at the A-box are buffered. At time L the A-box starts to send bits (at the same rate as a B-box) until a whole packet has seen sent. If the buffer is empty the A-box will wait another L time units and will start sending at time $2L$. In general, the ith B-box checks its buffer at times $mL + ik$, $m = 0, 1, 2, \cdots$, and if the buffer is empty it may start sending its own packet. If the buffer contains the address of station i, the B-box will remove the arriving packet and may, at the same time, send out its own packet. If the buffer contains an address different than station i, the B-box will pass on the arriving packet.

In a similar manner the A-box checks its buffer at times mL, $m = 1, 2, \cdots$,. If the buffer is not empty, it sends out L bits. If the buffer is empty, the A-box remains inoperative for the next L time units.

The loop time is the time it takes a bit to complete one round of the loop, and is measured in multiples of L. In the single A-box loop described here the loop time is the smallest integer greater than or equal to nk/L where n is the number of B-boxes in the loop. Clearly there is a complete analogy between the loop described here and the one presented in Fig. 1 if the number of revolving arms is taken as equal to the loop time.

It is important to note that the simulated system described here represents only one conceptual way in which the loop may be operated. It is possible, for example, that the packets will move from one B-box to another in one block rather than bit by bit. This is equivalent to placing an A-box between any two B-boxes (n rotating arms).

We have used the digital simulator to examine the queuing characteristics of several small systems. The main purpose of the simulation was to study the effects of dominating classes in nonheavy (nonsaturated) traffic situations. Numerical results are presented for an 8-station loop with two dominating classes $C_1 = (S_1, S_2, S_3, S_4)$ and $C_2 = (S_5, S_6, S_7, S_8)$. The loop time for this system was selected to equal 1 (one rotating arm) and the system was simulated for three different expected main line loads (utilization). The P matrix and average queue sizes (in packets) at the stations are shown below.

Simulated Example: 8-Station Loop (Two Dominating Classes)

$$P = \begin{array}{c} \\ S_1 \\ S_2 \\ S_3 \\ S_4 \\ S_5 \\ S_6 \\ S_7 \\ S_8 \end{array} \begin{array}{c} \begin{array}{cccccccc} S_1 & S_2 & S_3 & S_4 & S_5 & S_6 & S_7 & S_8 \end{array} \\ \left[\begin{array}{cccc:cccc} 0 & 1/3 & 1/3 & 1/3 & 0 & 0 & 0 & 0 \\ 1/3 & 0 & 1/3 & 1/3 & 0 & 0 & 0 & 0 \\ 1/3 & 1/3 & 0 & 1/3 & 0 & 0 & 0 & 0 \\ 1/3 & 1/3 & 1/3 & 0 & 0 & 0 & 0 & 0 \\ \hdashline 0 & 0 & 0 & 0 & 0 & 1/3 & 1/3 & 1/3 \\ 0 & 0 & 0 & 0 & 1/3 & 0 & 1/3 & 1/3 \\ 0 & 0 & 0 & 0 & 1/3 & 1/3 & 0 & 1/3 \\ 0 & 0 & 0 & 0 & 1/3 & 1/3 & 1/3 & 0 \end{array}\right] \end{array}$$

No.	Line and Source Utilization	Average Queue Sizes								Ave.	Max.
		S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8		
1	$\lambda = 0.239$, $\rho = 0.956$	9.7	10.1	13.8	6.7	15.6	14.6	11.2	6.8	11.1	33–58
2	$\lambda = 0.229$, $\rho = 0.916$	5.6	5.6	6.1	3.0	5.0	6.8	5.2	3.2	5.1	26–45
3	$\lambda = 0.213$, $\rho = 0.852$	2.4	2.6	1.9	1.1	2.4	2.3	2.1	1.2	2.0	16–22

The alternating priorities effect, due to domination, is demonstrated in Fig. 3. The total number of packets at the four queues of C_1 (dotted line) and C_2 (solid line) were plotted against time. For the case ρ =

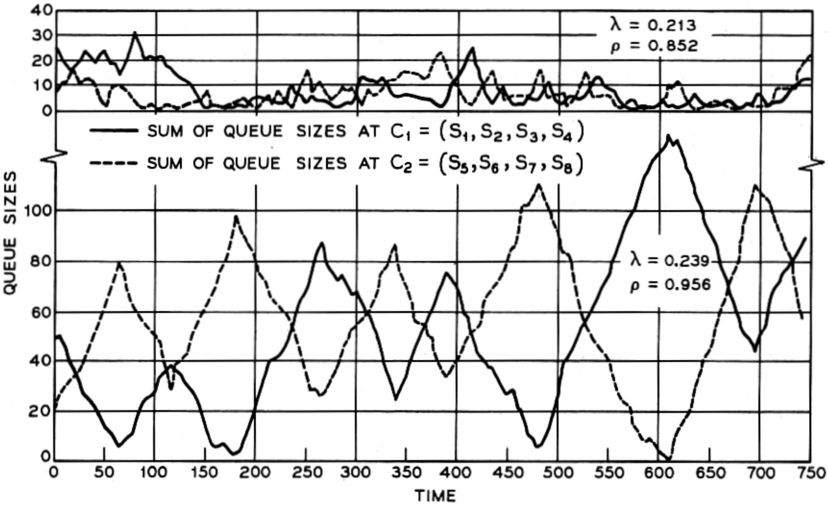

Fig. 3—Simulated queue sizes.

0.956 (high utilization) one can clearly see that only one dominating class is served at one time. When stations of C_2 take over control of the system the queues at stations of C_1 build up while at C_2 they are being depleted until the queue at S_8 reaches zero and C_1 can take over.† The average cycle time (time elapsing between two consecutive peaks of the dotted or solid lines) for this case was 110 time units. As the load on the system is decreased, the alternating priorities effect becomes less and less distinctive. For $\rho = 0.916$ (not graphed) the average cycle time reduces to 23 time units, and for $\rho = 0.852$ (see Fig. 3) alternations are very frequent and cycles are practically unnoticeable.

Notwithstanding the complete symmetry within classes the average queue sizes at S_4 and S_8 are consistently smaller than the average queue sizes at the other stations. The explanation of this phenomenon is as follows: At a moment when C_1 loses control to C_2 the queue size at S_4 is zero while at S_1, S_2, and S_3 it is greater than or equal to zero. From that moment on the queues of C_1 build up at equal average rates until the moment control returns to C_1 (peak of the dotted line in Fig. 3). At that moment the queues start being depleted at equal average

† Note that the queue size at S_8 being zero is a necessary but not sufficient condition for losing control.

rates. Since the expected queue size at S_4 is smallest, it has a higher probability of being completely depleted first. This phenomenon tends to shorten the alternations cycle.

The alternations cycle will be shorter when the number of rotating arms (loop time) is increased. Therefore large loops (local or regional loops) will be relatively more stable when high utilization occurs.

The nature of the stochastic process used for generating packets at the stations of the loop is described in Ref. 5.

An important aspect, not analyzed in this study, is the question of the order of stations in the loop. Clearly, the amount of traffic the loop can carry and the resulting congestion are strongly dependent on the specific order of the stations in the loop. In Example 1 we have assumed counterclockwise traffic direction. If we reverse the direction of traffic on the main line we shall get a different solution for the flows. The two solutions are compared in the following:

	Counterclockwise Flow Direction	Clockwise Flow Direction
	$x_1 = 0.4646$	$x_1 = 0.5000 = \lambda_1$
	$x_2 = 0.4000 = \lambda_2$	$x_2 = 0.4000 = \lambda_2$
	$x_3 = 0.5000 = \lambda_3$	$x_3 = 0.5000 = \lambda_3$
	$x_4 = 0.2333$	$x_4 = 0.0917$
	$x_5 = 0.0508$	$x_5 = 0.3667$
Total:	1.6487	1.8584

Reversing the flow direction results in an increase of 13 percent in the amount of satisfied demand.

In a practical situation not all orders are feasible. Still the number of feasible orders may be overwhelmingly large and an appropriate algorithm for determining best order is called for. An interesting possibility is a double loop system where each station is connected to two loops with opposite traffic directions. This may increase reliability and enable better utilization by allocating traffic in an efficient manner. One possible allocation rule is shortest distance allocation where the loop to be used is the one with the shortest travel distance for each particular message (this is an example of a possible rule and is not proposed as an optimal rule).

The bounded linear complementarity problem presented by equations (7) is of somewhat more general interest, bearing little relation to the Pierce loop. In matrix form we have

$$X + U = \lambda,$$
$$AX + Z = 1,$$
$$U^T Z = 0, \qquad U \geqq 0, \qquad Z \geqq 0, \qquad X \geqq 0.$$

Substituting $X = \lambda - U$ we obtain the same set of relations in a slightly different form.

$$AU - Z = A\lambda - 1 = q$$
$$U^T Z = 0, \qquad U \geqq 0, \qquad Z \geqq 0, \qquad U \leqq \lambda. \tag{28}$$

It is now possible to compare our problem to the Fundamental Problem† treated by Lemke,[6] and Cottle and Dantzig.[2]

$$AU - Z = q$$
$$U^T Z = 0, \qquad U \geqq 0, \qquad Z \geqq 0. \tag{29}$$

The only basic difference between the two problems is that in our problem U is bounded from above while in Lemke's problem U is unconstrained. In this respect our problem is more general. The shift-and-load procedure is, however, fundamentally based on the specific structure of A and q and may not prove useful for a wider class of parameters.

VI. ACKNOWLEDGMENT

I am indebted to J. Salz, J. F. Hayes, and D. N. Sherman for introducing me to the Pierce loop concept, and to M. Segal and S. Halfin for their helpful comments and suggestions.

A special acknowledgment is due R. R. Anderson who wrote the digital simulation and ran the numerical examples used in this study.

APPENDIX

In this appendix we outline a procedure for determining all dominating submatrices of a given stochastic $n \times n$ matrix P.

Step 1: Construct a matrix $\Pi = \{\pi_{ij}\}$ where

$$\pi_{ij} = \begin{cases} 1 & \text{if} \quad p_{ij} > 0, \\ 0 & \text{otherwise.} \end{cases}$$

† The term "Fundamental Problem" was coined by Cottle and Dantzig.[2]

Step 2: $Q = \{q_{ij}\}: = \Pi$.
Do for $j = 1, 2, \cdots, n$.
Repeat until q_{kj}, $k = 1, 2, \cdots, n$, remain unchanged.

$$q_{kj} := q_{kj} + \sum_{\{i : q_{ij}=1\}} q_{ki}, \qquad k = 1, 2, \cdots, n.$$

(Note that all additions are Boolean.)
If $S_i \rightarrow S_j$ then in the resulting matrix Q the element $q_{ij} = 1$. Otherwise $q_{ij} = 0$.

Step 3: Construct a matrix $Q^{(0)} = \{q_{ij}^{0}\}$ such that

$$q_{ij}^{(0)} = q_{ji}^{(0)} = q_{ij}q_{ji}.$$

If S_i does not belong to a communicating class then $q_{ij}^{(0)} = 0$, $j = 1, 2, \cdots, n$. Otherwise $q_{ii}^{(0)} = 1$ and $S_j \in C(i)$ if and only if $q_{ij}^{(0)} = 1$. $C(i)$ is closed (dominating) if and only if the ith row of $Q^{(0)}$ is identical to the ith row of Q.

REFERENCES

1. Pierce, J. R., "Network for Block Switching of Data," unpublished work.
2. Cottle, R. W., and Dantzig, G. B., "Complementary Pivot Theory of Mathematical Programming," *Mathematics of the Decision Sciences*, Part 1, American Math. Soc., 1968, pp. 115–136.
3. Kemeny, G. J., and Snell, J. L., *Finite Markov Chains*, New York: D. Van Nostrand Co., 1965.
4. Avi-Itzhak, B., Maxwell, W. L., and Miller, L. W., "Queuing with Alternating Priorities," *Oper. Res.*, *13*, No. 2, 1965, pp. 306–318.
5. Hayes, J. F., and Sherman, D. N., "Traffic Considerations in a Pierce Data Loop," unpublished work.
6. Lemke, C. E., "Bimatrix Equilibrium Points and Mathematical Programming," Mgt. Sci., *11*, No. 7, 1965, pp. 681–689.