THE SPANNING TREE STRUCTURE OF STATIONARY MARKOV CHAINS

D. A. R. Seeley

Department of Computer Science
The University of British Columbia
Vancouver, British Columbia, Canada

Introduction

This paper will show that the stationary probabilities of Markov Chains can be simply related by a certain graph theoretic concept. In addition to its theoretical elegance, this result will be shown to be useful for (a) hand computations on systems with relatively few numbers of states, and (b) in algorithms that develop stationary probabilities for certain queuing formulations. In addition, if certain open problems could be solved, this graph theoretic approach would open a powerful alternative method in the analysis of queues.

The associated graph of the transition probability matrix of a finite Markov Chain is a useful means of representing its structure and classifying its states as demonstrated in De GHELLINCK[1]. Call it a Markov graph. The latter reference uses the matrix-tree theorem of BOTT and MAYBERRY[2], useful in Leontief economic systems, to relate the spanning tree structure of the Markov graphs of irreducible, discrete parameter chains to their stationary probabilities. This has also been pointed out by MEDVEDEV[3].

An analogous result is presented in this paper, and proven in appendix A (see also SEELEY[4]), for the stationary probabilities of continuous parameter Markov Chains. Also call the associated graph of the transition rate matrix a Markov graph. Then these probabilities are proportional to a simple function of the directed spanning trees at each state (node). The function is the sum of the products of the transition intensities taken over each distinct spanning tree. This result can be applied to various queuing formulations by developing these trees in a recursive manner. Since this process depends only on the structure of the Markov Chain, arbitrary transition intensities may be utilized that can incorporate many different assumptions regarding the nature of queue involved. The developmental nature of the algorithm automatically provides a means of doing sensitivity analysis on certain system parameters, typically queue capacity. In order to make this result precise it will be useful to discuss some graph theoretic and related definitions.

Definitions

Let the states of a finite, irreducible, homogeneous Markov chain M, be denoted by the set $S = \{s_1, s_2, \ldots, s_n\}$. Let the differential rate matrix of such a chain in continuous time be $A = [a_{ij}]$ and the state probability vector be $p(t)$. Also, let π_i, associated with s_i, be the stationary probability of being in that state $(i=1, \ldots n)$. Define the Markov graph of M to be $G = (S,U)$ where S represents both the set of nodes of the graph and the states of M for simplicity, and U represents the set of arcs of the directed graph where $u_{ij} \in U$ iff a_{ij} is nonzero. That is, an arc leads from s_i to s_j only if this state transition can take place.

Further definitions from graph theory are necessary. For the arc u_{ij}, s_i and s_j are called the initial and terminal nodes respectively. A sequence of arcs $(u_{12}, u_{23}, \ldots, u_{mn})$ such that the terminal node of each arc is the initial node of the next is called a path. A graph is strongly connected if there exists a path between any pair of its nodes.

A subgraph $H = (T,V)$ of $G = (S,U)$ is a graph where either T and/or V are proper subsets of S and U. H will be said to span G if T is identical to S. A finite path such that the initial node of the first arc is identical to the terminal node of the last arc is called a circuit.

The spanning tree structure of Markov graphs is central to this paper. Trees containing directed arcs will be referred to either as "a tree to a node", or as "a tree rooted at a node" (the latter are sometimes known as arborescences, BERGE[5]). These two types are distinguished by the direction that the arcs take in relation to the distinguished node. In particular, we will deal with trees to any node, s_o. These may be defined by a directed graph $G = (S,U)$ wherein (a) every node, excluding s_o, is the initial node of only a single arc, (b) no arcs lead from s_o, and (c) the graph G contains no circuits. Such a graph contains n-1 arcs if $n = |S|$; an example is illustrated in figure 1. A consequence of the definition is that there is a path from every node to s_o.

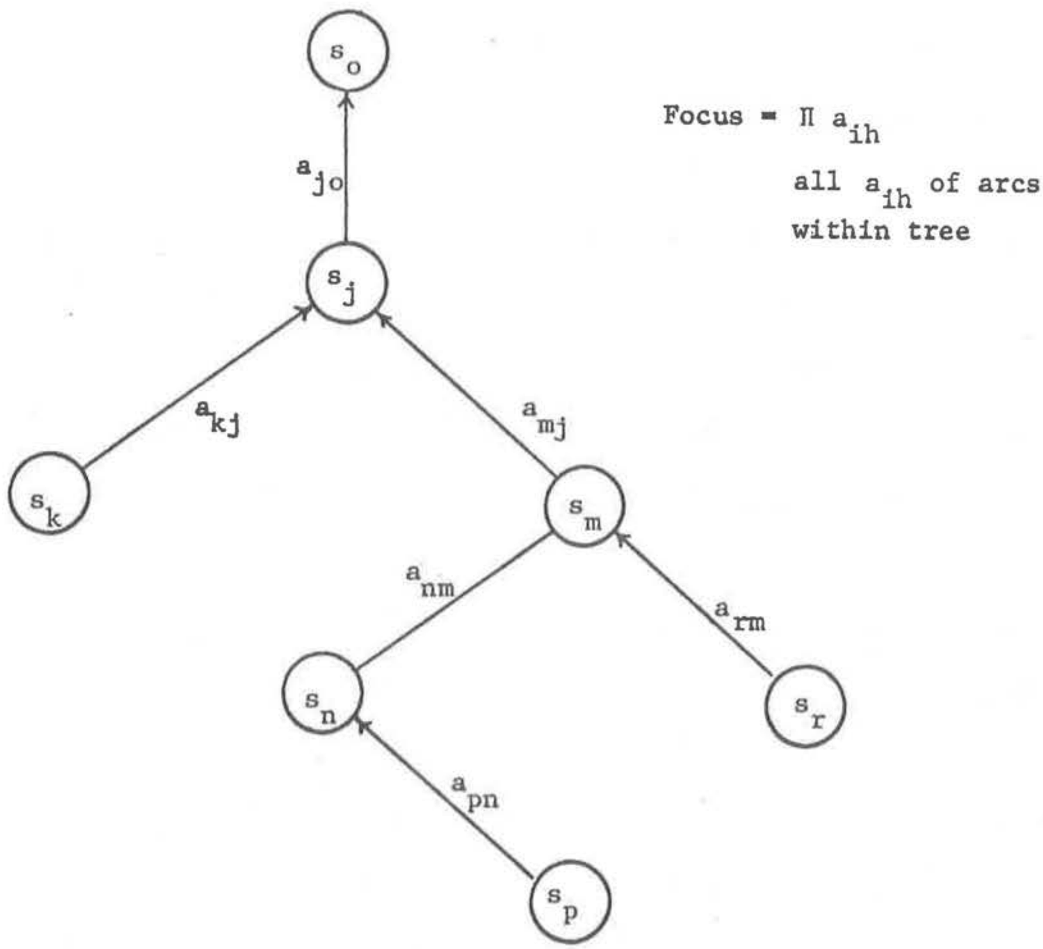


Figure 1: A Tree (S,U) at Node s_o and its Focus

The following definitions are derived from tree to a node, and are the key quantities in the stationary probability computations. In graphs wherein each arc u_{ij} has a coefficient a_{ij} associated with it, (the a_{ij} correspond both to the transition intensities of the differential rate matrix for queues and to the transition probabilities in discrete time Markov Chains) define a focus at node s_o of a particular spanning tree to that mode, to be the product of the coefficients on the arcs of the tree. This is illustrated in figure 1. Note that there may be many distinct spanning trees to a particular node, as subgraphs of any arbitrary directed graph. Define the total focus at a node to be the sum of all distinct foci at that node (i.e. each tree will have at least one arc different from each of the others).

The Total Focus Theorem

It is well-known that the associated graph of a finite irreducible Markov Chain is strongly connected. That is, it is possible to reach any state from any other state after at most n-1 transitions. Within such a graph it will be always possible to find at least one spanning tree to each node.

The matrix-tree theorem of BOTT and MAYBERRY[2] expresses the determinant of an arbitrary matrix by using the spanning trees of a modified associated graph. In fact, the determinant is equal to the sum of all of the foci of this graph. De GHELLINCK[1] has demonstrated the utility of this result to matrices that have the special property that the diagonal elements are equal to the sum of the off-diagonal row elements. He then applied the theorem to the solution of the stationary probabilities of discrete time Markov Chains (although he used arborescences and the transpose of the transition matrix).

For simplicity, call the above matrices Kirchhoff matrices (i.e. when the diagonal elements are equal to the sum of the off-diagonal row elements) since similar results were first proven to apply to current flow in electrical circuits. These matrices reflect physical systems wherein there is a kind of flow conservation law operating. The sum of flow entering a node must equal the sum of the flows leaving that node. In such systems Bott's matrix tree theorem may be applied in order to determine the steady-state flows through the nodes; since in addition to the determinant, the principal minors are given by a function of the spanning trees to each node (i.e. their total foci).

In Markov Chains, the steady-state equations for the state probabilities yield systems of homogeneous equations whose matrix of coefficients is Kirchhoff. In discrete time, if P is the one-step transition matrix and π the stationary state probability vector, then the system is:

$$\pi \cdot (I-P) = 0 \qquad (1)$$

In continuous time, with differential rate matrix A, the corresponding system is:

$$\pi \cdot A = 0 \qquad (2)$$

It is evident that (I-P) satisfies the Kirchhoff property since the row transition probabilities of P sum to unity. The differential rate matrix A however, is derived from an approximating discrete time Markov Chain whose transition matrix T is stochastic like P. The diagonal of T however, always has a term of the form, unity minus the transition probabilities possible from that state. In formulating the system of differential equations from which equation (2) is derived, the unity on the diagonal is removed in order to form the differential, leaving:

$$\frac{d\,p(t)}{dt} = p(t) \cdot (T-I) \qquad (3)$$

The limit of the derivative of the state probability vector p(t) as t tends to infinity is taken to be the constant zero vector, leaving equation (2) in which A = (T-I). The vector π is determined from the homogeneous systems (1) and (2) by also satisfying $\sum_{i=1}^{n} \pi_i = 1$. In applying the matrix tree theorem, the coefficients of the arcs of the spanning trees associated with discrete time Markov Chains are the transition probabilities, whereas in continuous time they are the transition intensities. Hence, the following theorem may be stated for finite, irreducible, homogeneous Markov Chains.

Theorem: The stationary probability of state j of a Markov Chain is equal to the total foci of node j in the associated graph, normalized by the sum of all total foci.

This result is proven in appendix A by applying Mason's loop rule for determinants in a flowgraph version of the system of equations (3).

An Example

Consider a service system consisting of two channels in series but with no queue allowed before either channel. Assume a Poisson arrival rate of λ items to the system, some of which will be lost if the 1st channel is busy. Both service channels have exponentially distributed service times, with means $1/\mu_1$ and $1/\mu_2$ respectively. What are the steady-state probabilities?

First draw the Markov graph corresponding to this system, ignoring self-loop transitions since these cannot enter into any spanning trees. The states of the system are indicated by characters describing the status of each channel; 0 for empty, 1 for occupied, and b for channel 1 blocked after a service completion. Note that self-loops cannot enter into any trees, therefore they will be omitted in all Markov graphs.

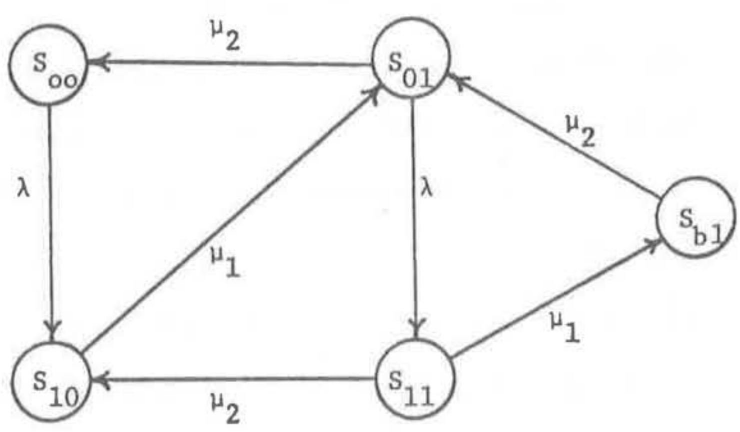
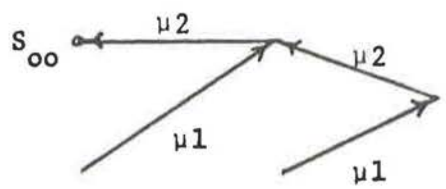
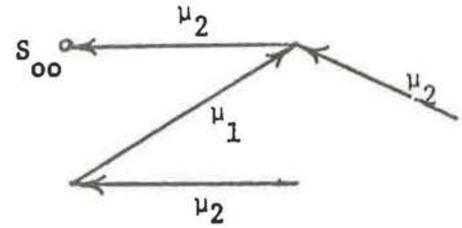


Figure 2: Markov Graph of a Tandem Service System with No Queues

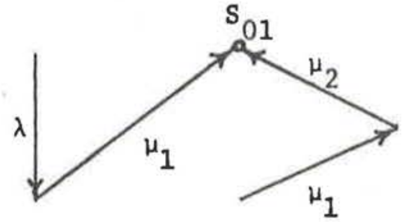
In order to solve this system using the above theorem, (1) enumerate the spanning trees at each state j and compute its total focus T(j) (2) sum all of the total foci in some convenient fashion, and normalize the T(j) yielding the stationary probabilities π_j. Hence:

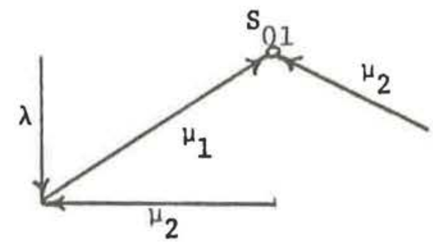
at S_{oo} there are 2 spanning trees



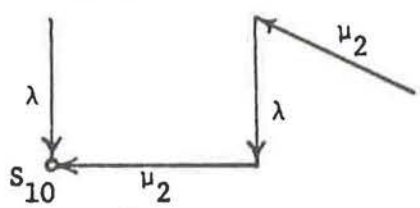


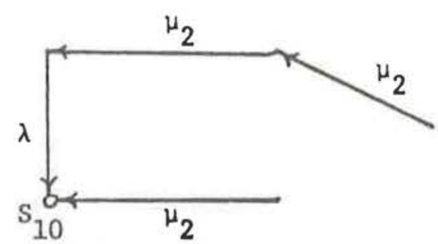
$\therefore\ T(00) = \mu_1^2 \cdot \mu_2^2 + \mu_1 \cdot \mu_2^3$

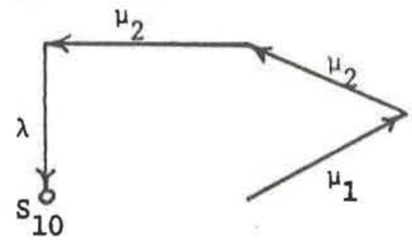




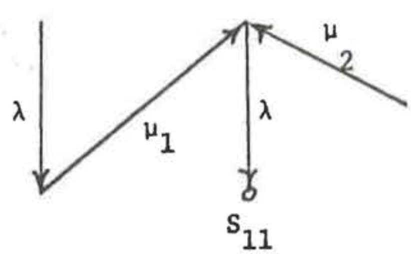
$\therefore T(01) = \lambda\mu_1^2 \cdot \mu_2 + \lambda\mu_1\, \mu_2^2$



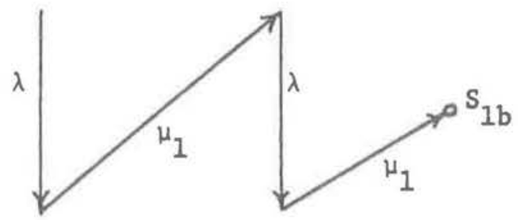




$\therefore\ T(10) = \lambda^2\mu_2^2$
$+ \lambda\mu_2^3 + \lambda\mu_1\mu_2^2$



$\therefore\ T(11) = \lambda^2\mu_1\mu_2$



$\therefore\ T(1b) = \lambda^2\mu_1^2$

Figure 3

Suppose in this instance that $\mu_1 = \mu_2 = \mu$, and that $\lambda/\mu = \rho$. If one divides each of the total foci by μ^4 this will yield the following:

$$
\begin{array}{lcl}
T'(00) & = & 2 \\
T'(01) & = & 2\rho \\
T'(10) & = & 2\rho + \rho^2 \\
T'(11) & = & \rho^2 \\
T'(b1) & = & \rho^2 \\
\hline
\text{Total} & = & 2 + 4\rho + 3\rho^2 = K
\end{array}
$$

With the constant K, many system quantities can quickly be computed: e.g.:

prob (idle system) $= \pi_{oo} = 2/K$

prob (item is lost) $= \pi_{10} + \pi_{11} + \pi_{b1} = (3\rho^2 + 2\rho)/K$.

Example 2A: Birth and Death Model

The stationary probabilities of states in birth and death continuous Markov Chains are well known [6]. However, using the Markov graph and the total focus theorem, evaluation becomes exceedingly simple to do by hand. This is due to the fact that nodes in such models have only one focus apiece, and each with a simple structure, as witness the following specific application.

Suppose a system contains a group of 4 machines each subject to an exponential breakdown rate λ, and also that 2 repairmen are available, each with exponential service rate μ. Let state of the system, j, represent j machines not working, and let $\rho = \lambda/\mu = 1/4$. Suppose that the probability that in the long run at least 2 machines are down is desired, (i.e., a machine just breaking down must wait for service), i.e. the sum of π_3 and π_4.

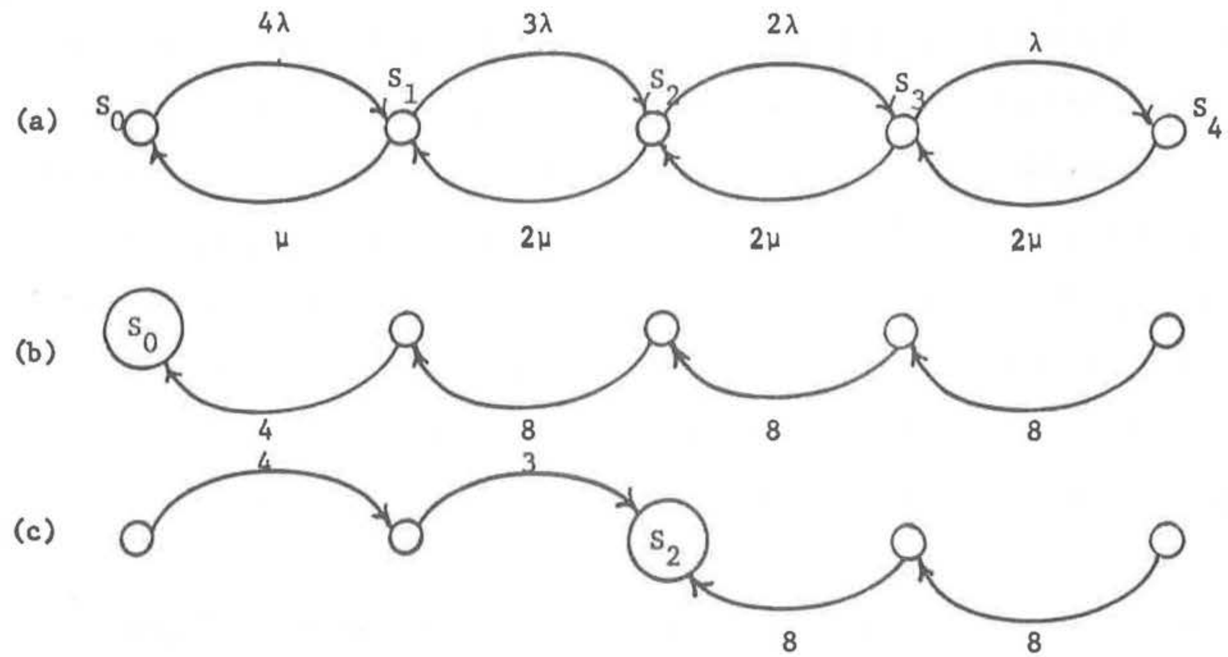


Figure 4: Markov Graph and Foci in a Machine Repair Problem

In figure 4(a) the transition intensities (ignoring self-loops) of the continuous Markov Chain corresponding to the problem, have been marked using relative magnitudes based on ρ. Figures 4(b) and 4(c) show the arcs of two of these foci, and values of T(j) proportional to the stationary probabilities are obtained as follows:

$$T(0) = 4*8*8*8 = 4*512$$
$$T(1) = 4*8*8*8 = 4*512$$
$$T(2) = 4*3*8*8 = 4*192$$
$$T(3) = 4*3*2*8 = 4*48$$
$$T(4) = 4*3*2*1 = 4*6$$
$$\text{Total} = 4*1250$$

From these figures the probability we are looking for is, $\frac{48+6}{1250} \simeq .04$

Other quantities of interest may be computed as easily.

Recursive Use of the Total Foci Theorem

The method of using spanning trees for computing stationary probabilities of arbitrary Markov Chains is useful only for problems small enough to be hand-solvable, such as in the previous examples. Since the number of spanning trees in strongly connected graphs of some complexity grows very rapidly as the number of arcs and nodes increases (see for example the enumerations in Harary[8]), computing total foci by enumerating spanning trees will be clearly inefficient. However, for regularly connected graphs such as those Markov graphs arising from queuing theory, the total foci may be computed recursively in a very efficient and useful manner. A general discussion and an example provide evidence for this assertion.

First, consider the new spanning trees that are formed when a single new node X is added to a strongly connected graph, as illustrated by figure 5(a). In this example, there are only 2 arcs that connect X to the original graph. It is clear that any spanning tree at X must make use of the arc leading from Y to X. Now since any old spanning trees to Y may be validly augmented this way, the total focus at X is:

$$T(X) = a \cdot T'(Y) \tag{5}$$

A similar argument for the new total focus at Y prevails, involving the inclusion of the arc from X to Z to all of the original spanning trees at Y; therefore:

$$T(Y) = b \cdot T'(Y) \tag{6}$$

An analogous argument applies partially to the new total focus at Z. However, there may now be spanning trees at Z which include both the 'a' and 'b' arcs. In order to compute this factor, consider that any arcs from Y in the construction of the spanning trees of T'(Z) must be removed. In fact, what has just been described is a graph theoretic concept known as a forest.

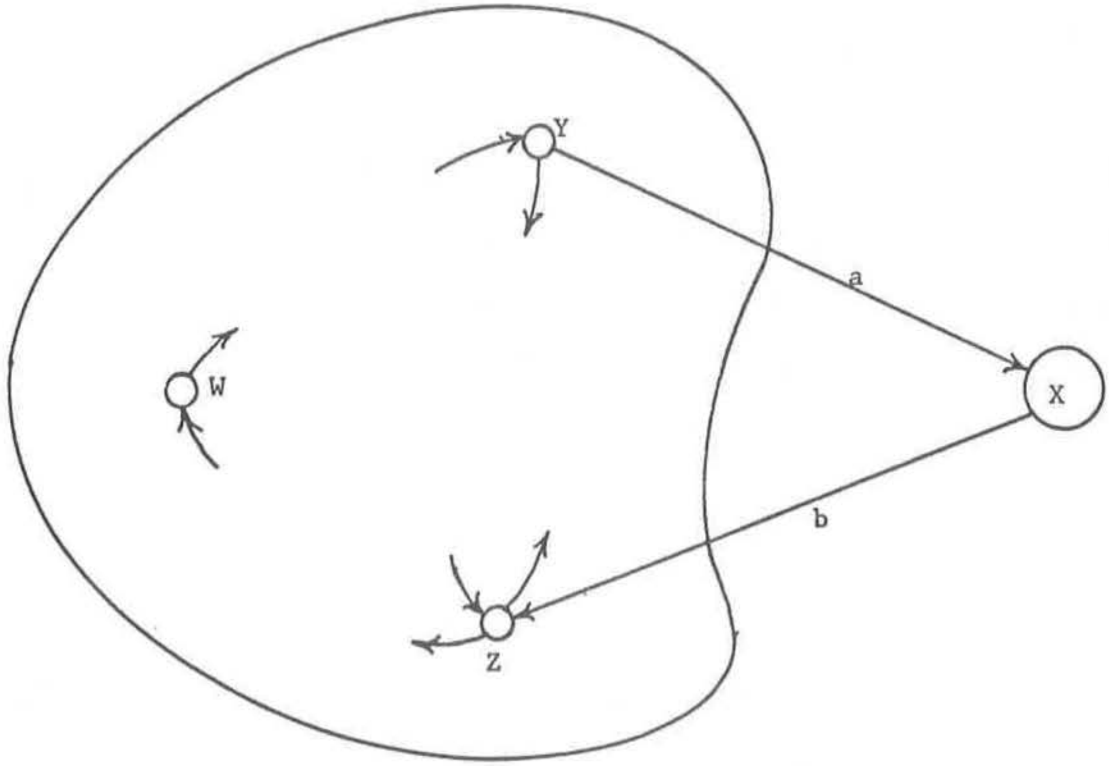


Figure 5 (a), Adding a New Node to a Strongly Connected Graph

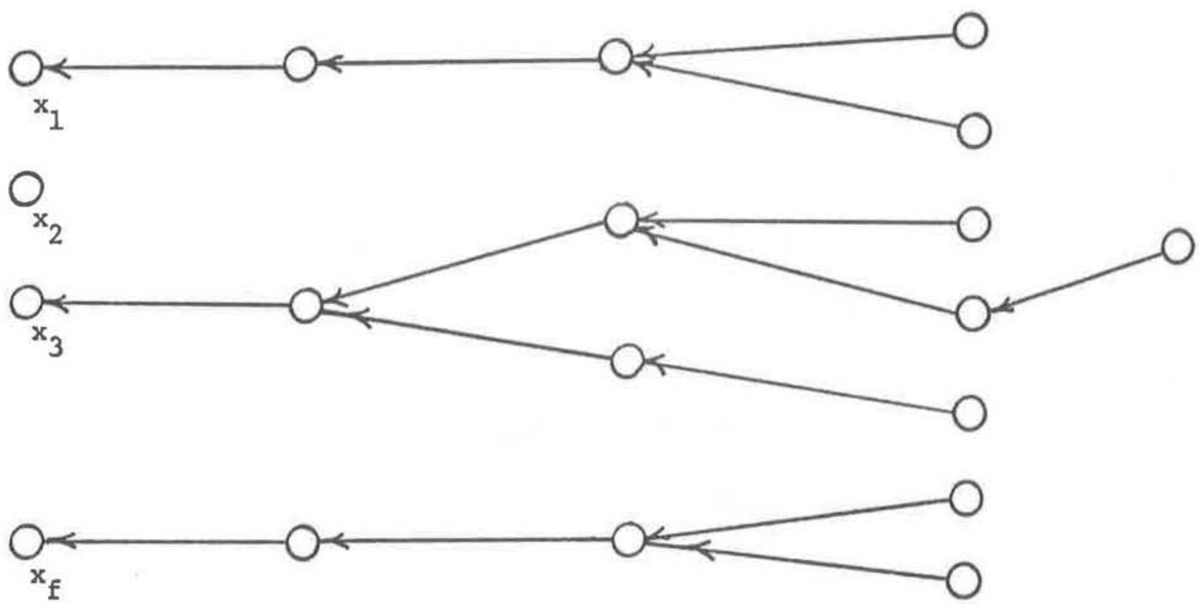


Figure 5 (b), A Forest of Foci at X_1, X_2 ..., X_f

Figure 6: Models for the Description of Growing Total Foci

Formally, a spanning forest at nodes x_1, $x_2,\ldots x_f$ of a graph G is a subgraph that contains no circuits and all nodes of G other than $x_1,x_2,\ldots x_f$ are the initial nodes of exactly one arc, while the latter are not initial nodes of any arc. A typical forest is illustrated in figure 5(b). Similarily to the notion of a total focus, call the sum of the products of arc weights for each distinct forest at a set of nodes as a total forest, denoted by $T(x_1,x_2\ldots x_f)$.

Therefore, the quantity described above is T'(Y,Z), and the new total focus at Z is:

$$T(Z) = b\cdot T'(Z) + a\cdot b\cdot T'(Y,Z) \tag{7}$$

Now consider, the new total focus at node W. Arguments analogous to those for T(Z) prevail for T(W), except that there is a further complication with the term a·b·T'(W,Y). If there were spanning forests included in T'(W,Y) that had a path from Z to Y (and necessarily not to W), then the addition of arcs 'a' and 'b' to such a forest would produce a circuit through nodes X, Y, and Z. Such terms could not be included in any new spanning tree at W.

Hence, there is need to define a new term T(W*Z,Y), which is the sum of the arc products from all distinct spanning forests at W and Y, such that W is a descendant (on a path from) Z. There may arise situations where it is easier to compute the equivalent quantity T(W,Y) - T(W,Y*Z). In any case, the new total focus at W may be exptressed as:

$$T(W) = b\cdot T'(W) + a\cdot b\cdot T'(W*Z,Y) \tag{8}$$

For any particular queuing formulation the equations for determining the stationary probabilities by means of recursive spanning trees may be developed in this manner. The general procedure is to consider four classes of equations (a) those for the new states, such as equation (5), (b) those for states neighbouring to these, such as equations (6) and (7), (c) equations for background states such as for W above, and (d) equations that compute as intermediate quantities, the various total forests.

This method becomes very difficult when considerations such as those above in defining T'(W*Z,Y) become necessary for many background states. This possibility increases when more than one state is added at each iteration as may be required for multi-parameter queuing models. This will be illustrated in a subsequent paper which solves the general k-phase service system (see for instance [4]). Now consider an example that is easily handled by this approach.

A 2-Phase Service Model

The method of phases has been extensively used to model queuing processes involving the Erlang Distribution [7]. Consider a 2-phase system allowing a maximum of n entries, in which the service facility may be occupied by only one item at a time, as described by figure 6. Notice that a_j and b_j are the exponential rates of arrival and of phase service respectively, when there a j items in the system. Hence, this model will allow for arbitrary state-dependent behaviour. The states j_1 and j_2 denote j items in the system, with the subscript indicating the current phase of service. Of course, an empty stste θ is required.

In figure 6(b) is illustrated the process whereby the stationary probabilities are computed recursively. The method proceeds by growing the Markov graph in stages, each subsequent stage yielding the total focus of each state when the queue capacity is allowed to increase by one. It starts with the initial total foci of a system with no queue. For notational convenience denote the total focus of j_1 for example, as $T_{j,1}$.

(a)

(b)

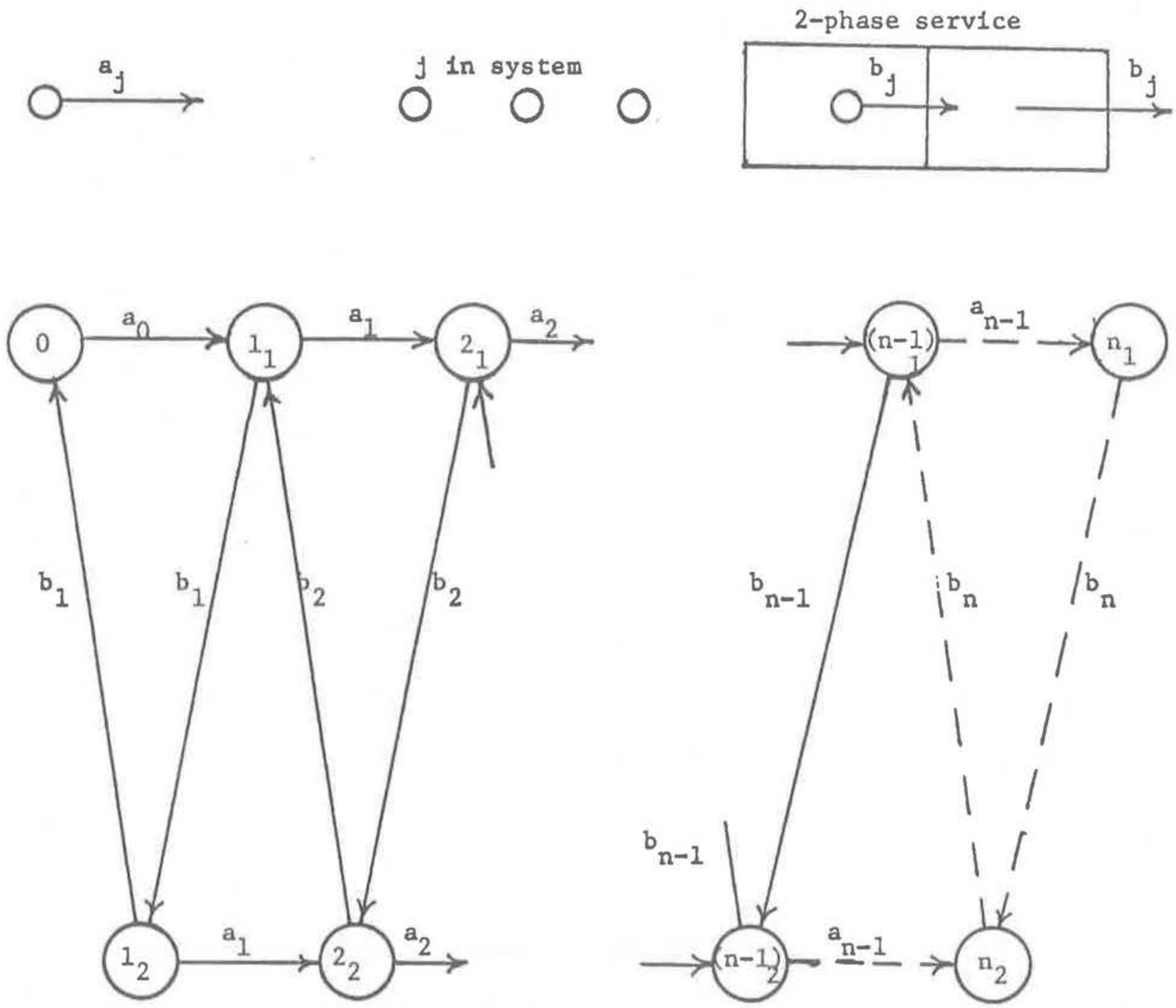


Figure 6: Markov Graph for 2-Phase Service System by Recursion

The initial total foci for the above system (no queue allowed) is simply:

$$T_\theta = b_1 \cdot b_1$$

$$T_{11} = T_{12} = a_0 \cdot b_1 \qquad (9)$$

since there is only one spanning tree at each state.

Next consider the inductive process of increasing the system capacity to n from n-1, as shown by the dotted arcs in figure 5(b). If one assumes that total the previous foci of states θ and j_1, j_2 for $j = 1,2,\ldots n-1$ are known and represented by T'_θ, $T'_{j,1}$ and $T'_{j,2}$, then the total foci of states n_1 and n_2 can be easily computed.

Spanning trees can be formed at n_1 by only 2 ways: (1) using $T'_{n-1,1}$

and the arcs leading from n_2 to $(n-1)_1$ and from $(n-1)_1$ to n_1, and (2) using the previous total forest at nodes $(n-1)_1$, $(n-1)_2$, and the arcs along the path passing through the nodes $(n-1)_2$, n_2, $(n-1)_1$, and n_1. Where the sum of the arc products of the previous total forest is denoted by Forest'. Hence:

$$T_{n,1} = a_{n-1} \cdot b_n \cdot T'_{n-1,1} + a^2_{n-1} \cdot b_n \cdot \text{Forest}' \tag{10}$$

At n_2 however, there are 3 ways of forming spanning trees, each utilizing the previous total foci at either of $(n-1)_1$ or $(n-1)_2$, or the aforementioned total forest. Hence:

$$T_{n,2} = a_{n-1} \cdot b_n \cdot (T'_{n-1,1} + T'_{n-1,2}) + a^2_{n-1} \cdot b_n \quad \text{Forest}' \tag{11}$$

Now, examine the remaining neighbourhood and background states. Adding 2 nodes to the graph necessitates that they each contribute arcs to any new spanning trees formed at these states. In addition to the two b arcs there are two new a_{n-1} arcs available. Recall however, that circuits cannot appear in a tree. This would necessarily happen, if for instance the upper a_{n-1} arc were used. The lower a_{n-1} arc can only be used in forming a spanning tree at node $(n-1)_1$. This is done by removing from consideration the b_{n-1} arc between the nodes $(n-1)_1$ and $(n-1)_2$, leaving the total forest referred to above. Hence:

$$T_{n-1,1} = b^2_n \cdot (T'_{n-1,1} + a_{n-1} \cdot \text{Forest}') \tag{12}$$

Neither of the a_{n-1} arcs may enter into a spanning tree at $(n-1)_2$, and all remaining states must be reached by means of a path from states $(n-1)_1$ to $(n-1)_2$ to $(n-2)_1$. Therefore, if "e" represents any state not formulated above, the remaining total foci to be recomputed can be found from:

$$T_e = b^2_n \cdot T'_e \tag{13}$$

All that remains to be determined is the recurrence relationship for the new total forest at nodes n_1 and n_2. New spanning forests may be formed by (a) the old forest and the two a_{n-1} arcs, (b) using the upper a_{n-1} arc and the previous total focus at $(n-1)_1$ which isolates the node n_2, and (c) in an analogous fashion, isolating node n_1. Hence:

$$\text{Forest} = a^2_{n-1} \cdot \text{Forest}' + a_{n-1} \cdot (T'_{n-1,1} + T'_{n-1,2}) \qquad (14)$$

Using the above equations therefore, allows one to compute the stationary probabilities of the 2-phase service system for increasing system capacities and using transition intensities that may depend upon system state (or phase if desired). This just requires normalizing the total foci at each stage. To show that this is an efficient method the following experiment was carried out. Programs were written in Algol W to solve this queuing system by both (a) the equations (9) - (14), and (b) Gaussian elimination for the corresponding system of linear equations. (This is also a useful method of verification of the formulae).

The following is the execution time in milliseconds for each method. The recursive spanning tree method has times that are necessarily cumulative including normalizations at intermediate stages. The number of states reflect the maximum capacities of $n = 2,4,8,16...$

Number of States	Recursive Spanning Trees	Gaussian Elimination
5	21 msec.	35 msec.
9	83 "	159 "
17	284 "	1,133 "
33	1021 "	11,727 "

Quite clearly, the recursive method is much faster, apparently of a time complexity of somewhat less than n^2. Also, an advantage of the recursive method over any closed analytic solution is its ability to use arbitrary values for the transition intensities. The author has found the approach utilized in the previous example applicable in several other cases, including bulk queues, and k-phase service and arrival queues.

Several questions raise themselves at this point. How complex can the Markov graph be, before this approach becomes computationally inferior to solving linear equations? Recall that one of the virtues of spanning trees is the arbitrary manner by which transition intensities may be defined. Does there exist an algorithm for recursively computing the spanning trees for the complete graphs (or will orthodox determinant computations always be faster)? Does there exist a notation that can describe the structure of Markov graphs (of queuing models) in such a way that it may be combined with the total focus theorem in order to automatically yield the required recursive equations?

Conclusions

Described in this paper is a theorem that relates the stationary probabilities of Markov Chains to the spanning tree structure of their associated graphs in a simple fashion. So simple in fact, that an intuitive explanation of the role that spanning trees play in equilibrium flow is desirable. In addition to its elegance and its utility for hand calculations of small systems, it has been shown that the theorem may be usefully applied to the highly regular graphs of some queuing formulations in a recursive manner. This fact and some of the unsolved questions its use raises, make the recursive spanning tree method of solving queuing formulations an attractive alternative to traditional approaches.

References

1. Guy de Ghellinck, "Application de la Theorie des Graphes Matrices de Markov et Diagrammes Dynamiques", Cahiers du Centre d'etudes de Rechereche Operationelle, Brussels. Vol. 3 #1, 1961.

2. R. Bott and J.P. Mayberry, "Matrices and Trees", *Economic Activity Analysis*, edited by O. Morganstern. J. Wiley and Sons.

3. G.A. Medvedev, "Analysis of Discrete Markov Systems by Means of Stochastic Graphs", translated from Avtomatika i Telemekhanika, Vol. 26, No. 3.

4. D.A.R. Seeley, "The Structure of Homogeneous Markov Chains", M.A.Sc Thesis, Department of Industrial Engineering, University of Toronto, 1966.

5. C. Berge, *The Theory of Graphs*, Methuen. 1962

6. E. Parzen, *Stochastic Processes*, Holden-Day. 1960

7. D.R. Cox and W.L. Smith, "*Queues*" Methuen. 1961

8. F. Harary, "*Graph Theory*", Addison Wesley. 1969

Appendix A: Proof of the Total Focus Theorem

Preamble

Consider the differential rate matrix A of a continuous parameter Markov chain. The derivative of the state probability vector is:

$$\frac{dp(t)}{dt} = p(t)\cdot A \qquad (15)$$

Taking the Laplace transform of this equation and denoting $L\{p(t)\}$ as $P(s)$, yields:

$$P(s) = \frac{1}{s} \cdot (P(s)\cdot A + \text{initial conditions}) \qquad (16)$$

In the theorem, this set of equations (16) will be represented by a flowgraph. A flowgraph represents linear equations relating variables in a directed graph form, wherein the nodes represent variables and the coefficients of the arcs represent the transmittance between variables. The dependency of a variable X_j upon another X_i in such a graph is given by Mason's loop rule.

$$D_{ij} = \frac{\sum_k L_{ij(k)}\cdot d_{ij(k)}}{D} \qquad (17)$$

where: D is the determinant of the flowgraph,

$L_{ij(k)}$ is the transmittance of the k-th path of all paths from X_i to X_j

$d_{ij(k)}$ is the k-th path factor, found by eliminating the loops touching the nodes along the k-th path, from the expression for the determinant.

The determinant is given by:

$$\begin{aligned} D = 1 &- (\text{sum of all loop transmittances}) \\ &+ (\text{sum of the products of all disjoint pairs of loops}) \\ &- (\ldots \text{disjoint triple products}\ldots) \\ &(-1)^n (\text{sum of all disjoint n-products of loops}) \end{aligned} \qquad (18)$$

where disjoint implies that loops have no nodes in common. Note that loop corresponds to a circuit in graph theory. Circuits of only one arc will be referred to as self-loops.

In order to get the stationary solution, the final value theorem will be applied to the flowgraph dependencies of (16). It is at this stage that the form of the solution switches to a graph theoretic representation. Note that the differential rate matrix A has the special Kirchhoff property that the diagonal elements are equal to the negative of the sum of the remaining terms in each respective row. Also (transition) intensity will be used synonomously with flowgraph transmittance in this context. Consider figure 7 as illustrative of (16), the flowgraph of which shall be referred to as a Markov flowgraph.

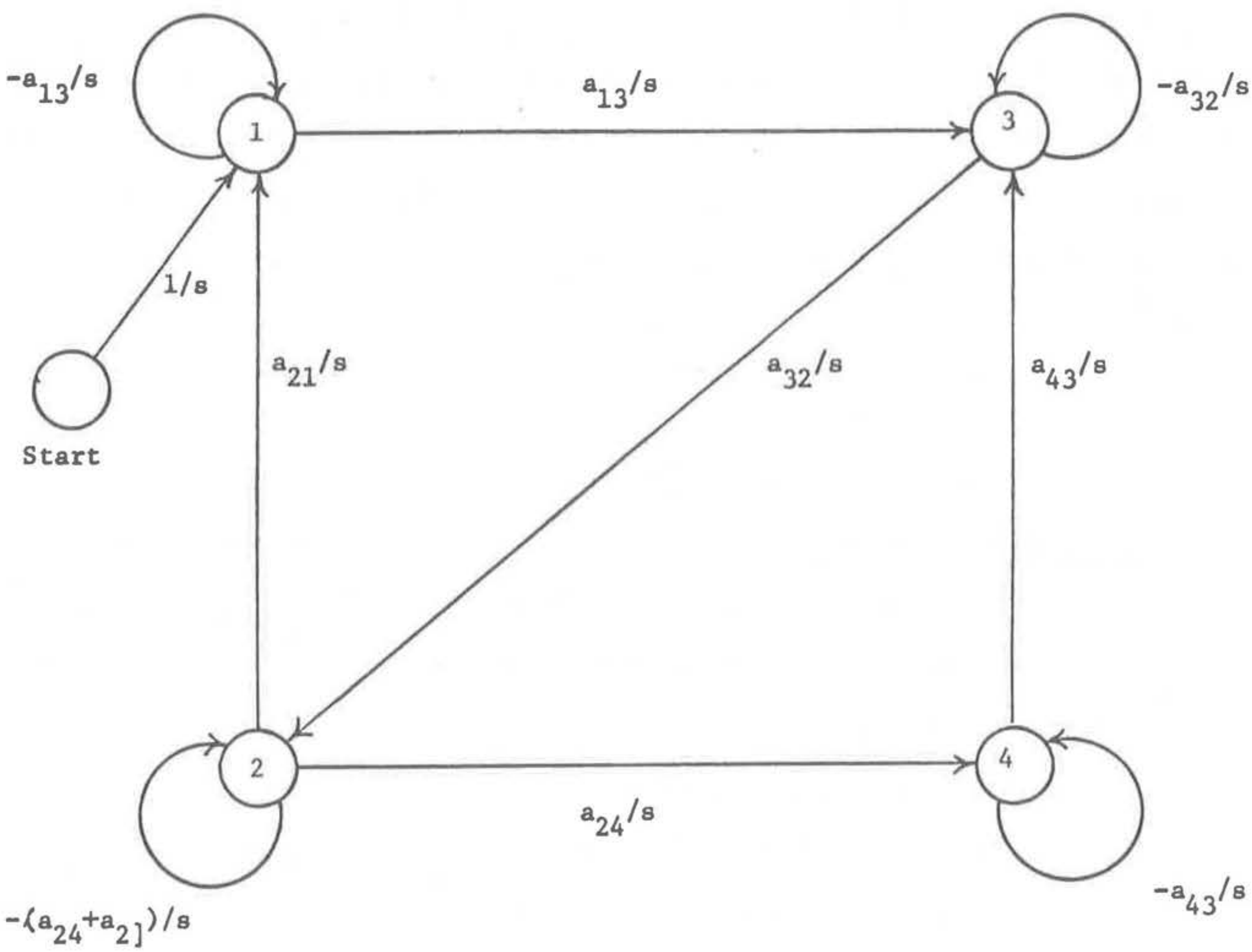


Figure 7: Illustrative Markov Flowgraph

Note the following characteristics of the graph: (i) The initial condition is at state 1 for convenience and without loss of generality; (ii) The self-loops at each node, are the negative of the sum of the intensities leaving that node. In the proof, the different coefficients in the self-loop term will be considered separately; (iii) The dependency of the state probabilities upon the initial condition as time advances to infinity will be formulated.

Denote as system focus the sum of the total foci at each node in a Markov graph of an irreducible system. The theorem may now be stated formally.

The Total Focus Theorem

Theorem: The stationary probability of any state j, in a finite, homogeneous, continuous time, irreducible Markov Chain is the total focus of the Markov graph at node j divided by the system focus.

Proof: The following lemmas which describe the characteristic terms in a Markov flowgraph determinant will be utilized.

Lemma 1: In the expansion of the determinant, there do not exist terms that include intensities whose corresponding arcs form a circuit of length greater than one arc (self-loop).

Proof: First note that any number of other intensities may be present in such terms, and that for every intensity there exists a component of a self-loop which is opposite in sign. Also, note that in the expansion of the determinant, the sign of a product which consists of k disjoint loops, of which p are self-loops, is $(-1)^k \cdot (-1)^p$.

Most important is the fact that there exists a 1-1 correspondence between each term that involves a circuit, and the term involving components of self-loops of the nodes of that circuit, those components which have the opposite sign of the arcs of the circuit. For example in figure 7, consider the circuit (loop) (a_{24}, a_{43}, a_{32}). For every term involving its intensities a_{24}/s, a_{43}/s, a_{32}/s, there is a corresponding term using $-a_{24}/s$, $-a_{43}/s$, $-a_{32}/s$ from the product of the three self-loops $-(a_{24}+a_{21})/s$, $-a_{43}/s$, and $-a_{32}/s$. Each term utilizes the nodes 2, 3 and 4. Therefore, any set of loops that is disjoint to one of these terms is necessarily disjoint to the other.

Now the contribution to the sign of any term from a circuit (not self-loop) is always -1, since the intensities of its arcs are positive and k=1. However, the contribution to the sign of the corresponding term using the self-loop components are always negative. Therefore, for every term developed from a circuit (any number of circuits, for that matter) there must exist a term of the same magnitude, but opposite in sign. Therefore all such terms are cancelled out in the determinant expansion. Q.E.D.

Corollary: The only terms that will remain in the expansion will be positive.

This follows from the fact that only terms composed of self-loops and the 1 are left in the loop rule expansion.

Lemma 2: No terms composed of n or greater intensities can appear, where n is the number of states in the chain.

Proof: A circuit would be formed by n or more intensities (or arcs). Since only self-loops can be involved, only one component corresponding to an arc from each self-loop can contribute to a product that would be disjoint. Now examine the graph formed by the corresponding arcs to these components. Using a theorem of BERGE, if n edges (ignoring the direction of arcs) were used then a cycle (non-directed) would necessarily form. In our case, that this cycle would correspond to a circuit follows from the fact that each node can be the initial node of one arc only. A cycle that has arcs going in different directions necessitates at least one node that is initial for at least 2 arcs. Therefore the only terms including n intensities correspond to circuits and must be cancelled by lemma 1.

Consequently, the only terms, in addition to 1, that will appear in the determinant are products of all possible disjoint, self-loop components, whose intensities do not embed any corresponding circuits, i.e.:

$$\det = 1 + V_1/s + V_2/s^2 + \dots V_{n-1}/s^{n-1} \tag{19}$$

where the V_k are the sums of all non-circuital k-products of intensities that do not have the same initial node (state). Equation (4) may be rewritten as:

$$\det = \frac{1}{s^{n-1}} \left(s^{n-1} + s^{n-2}V_1 + s^{n-3}V_2 + \dots + V_{n-1}\right) \tag{20}$$

Now consider the i-th path from the starting node to the state of interest j (there must be at least one since the graph is strongly connected). Assume it is of length $\ell(i)$, $1 \le \ell(i) \le n$, i.e., it passes through $\ell(i)+1$ nodes including the start and j. According to the rule by which the path factor is formed, the contribution of this path to the dependence of s_j upon the start will be:

$$\frac{d_i}{s^{\ell(i)}} \cdot \frac{1}{s^{n-1}} \left(s^{n-1} + s^{n-2}V_1^* + \dots s^{\ell(i)-1}V_{n-\ell(i)}^*\right) \tag{21}$$

where $d_i/s^{\ell(i)}$ is the path transmittance (a product of intensities), and V_k^* denotes the removal of any terms that possess an intensity of a node along the path. No products containing more than $n-\ell(i)$ components will appear since that would necessarily require an intensity from a path node, recalling that only one intensity from each node may appear in such products.

In accordance with the loop rule, (21) is now summed over all possible paths in order to find the overall dependence of $p_j(s)$ on the initial state. If the final value theorem is applied to determine the long run state probability, this yields

$$\pi_j = \lim_{s \to o} \frac{p_j(s)}{s^{n-1}} \cdot \frac{\sum_i d_i \cdot (s^{n-\ell(i)} + s^{n-\ell(i)-1} \cdot V_1^* + \dots V_{n-\ell(i)}^*)}{(s^{n-1} + s^{n-2}V_1 + \dots V_{n-1})/s^{n-1}}$$

$$= \lim_{s\to o} \sum_i d_i \cdot \frac{(s^{n-\ell(i)}+s^{n-\ell(i)-1}V_1^* +\ldots+V^*_{n-\ell(i)})}{(s^{n-1}+s^{n-2}V_1+\ldots+V_{n-1})}$$

$$= \sum_i \frac{d_i \cdot V^*_{n-\ell(i)}}{V_{n-1}} \quad (j-1,\ldots n) \tag{22}$$

V_{n-1} consists of all possible products of intensities such that there is only one intensity associated with any node and no associated circuits are formed. Therefore, V_{n-1} consists of all foci in the graph (excluding the initial arc) and is identical to the definition of a system focus. Now consider a term $d_i \cdot V^*_{n-\ell(i)}$. Since the path associated with d_i contributes $\ell(i)$-1 intensities a (a+1 only, being associated with the initial arc), there are again n-1 intensities from every node excluding s_j in this term. It follows that it is a focus at node s_j. It also follows that the summation over i represents the total focus at s_j because in every focus at s_j there is a path from any other node to s_j, so that all foci at s_j are represented in the summation. This would be the case no matter where the initial state occurred or indeed if all states had an initial probability. Therefore, it follows that

$$\pi_j = \frac{\text{total focus at state } j}{\text{system focus}} \tag{23}$$

Q.E.D.

Acknowledgments: The author appreciates the useful discussions with Jacob Levi regarding the presentation of the ideas in the paper. Also, the financial support of the National Research Council of Canada has made this work possible.