Integral Expressions for the Numerical Evaluation of Product Form Expressions Over Irregular Multidimensional Integer State Spaces

D. Manjunath
Dept. of Electrical Engg.
Indian Institute of Technology
Powai, Mumbai 208 016 INDIA
dmanju@ee.iitb.ernet.in

Biplab Sikdar
Dept. of ECSE
Rensselaer Polytechnic Institute
Troy, NY 12180 USA
bsikdar@networks.ecse.rpi.edu

Keywords— **Queueing Networks, Normalising Constant.**

Abstract— **In this paper we consider stochastic systems defined over irregular, multidimensional, integer spaces that have a product form steady state distribution. Examples of such systems include closed and BCMP type of queueing networks, polymerization and genetic models where the system state is a vector of integers, $\mathbf{n} = [n_1, \cdots, n_M]$ and the steady state solution is of the form $\pi(\mathbf{n}) = \prod i = 1^M f_i(n_i)$. To obtain useful statistics from such product form solutions, $\pi(\mathbf{n})$ has to be summed over some subset of the space over which it is defined. We consider situations when these subsets are defined by a set of equalities and inequalities with integer coefficients, as is most often the case and provide integral expressions to obtain these sums. Typically, a brute force technique to obtain the sum is computationally very expensive and algorithmic solutions covering specific forms of $f_i(n_i)$ and shapes of the space over which these are known. In this paper we derive general integral expressions for arbitrary state spaces and $f_i(n_i)$. The expressions that we derive here become especially useful if the generating functions $f_i(n_i)$ can be expressed as a ratio of polynomials in which case, exact closed form expressions can be obtained for the sums. The integral expressions that we derive here have wide applications and we demonstrate them by three examples in which we model finite highway cellular systems, copy networks in multicast packet switches and BCMP queueing networks.**

I. Introduction

Consider a stochastic system whose state is represented by a vector $\mathbf{n} = [n_1, n_2, \cdots, n_M]$, n_i are non negative integers for all i. Not all of the states $\mathbf{n}$ will be defined and we will denote the set of defined states by $\mathcal{S}$. We will assume that this state space is expressed by a set of linear constraints. In the modeling of such systems very often we can express the steady state probability of the system being in state $\mathbf{n}$, $\pi(\mathbf{n})$, in the following product form

$$\pi(\mathbf{n}) = \begin{cases} G \prod_{i=1}^{M} f_i(n_i) & \mathbf{n} \in \mathcal{S} \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

where G is some normalisation constant obtained to make $\pi(\mathbf{n})$ sum to 1.0 over the state space. Obtaining performance parameters like marginal distributions of the n_is or their moments involves summing $\pi(\mathbf{n})$ over a subset of the state space which is typically computationally very expensive if a brute force enumeration technique is to be used. For some specific situations like queueing networks and stochastic knapsack models where the $f_i(n_i)$ have the form $\rho_i^{n_i}/n_i!$ or $\rho_i^{n_i}$, efficient recursive algorithms to sum over the state space are available. See for example (Buzen 1973) for queueing networks with constant population and (Ross 1995) for stochastic knapsacks. In this paper we describe a transform technique to sum functions of the form $\prod_{i=1}^{M} f_i(n_i)$ over a state space defined by a set of linear equality and/or inequality constraints with integer coeeficients. With this method, if the generating function of $f_i(n_i)$ are rationals, then the method yields a closed form expression for the sum.

Before describing the technique, we discuss examples of some models where we can use the technique that we will describe in this paper.

Example 1: In (Kaufman 1981), Kaufman discusses two types of buffer sharing policies in a communication node - complete sharing and partial sharing. Messages arrive according to a Poisson process to a communication node with a total of K buffers. There are M classes of messages that can share the buffers. A class i message requires b_i buffers. Under the partial sharing policy, K_i buffers are dedicated to class i messages cand K_0 buffers belong to the common pool. A message of class i is admitted if there are a total of b_i buffers left in the class i pool and the common pool. If a message is not admitted, it is considered lost. Let n_i be the number of messages of class i in the node. Let the vector $\mathbf{n} = [n_1, n_2, \cdots, n_M]$ be state of the system and $\mathcal{S}$ the state space for $\mathbf{n}$ under this buffer sharing policy. We see that $\mathcal{S}$ is defined by

$$\mathbf{n} \in \mathcal{S} \;\text{ if} \begin{cases} n_i b_i \leq K_i + K_0 \\ \textbf{and} \\ \sum_{i=1}^{M} n_i b_i \leq K \end{cases} \quad \text{for } \; i = 1, \cdots M$$

In the complete sharing policy, all the buffers form the common pool and the state space $\mathcal{S}$ is given by

$$\mathbf{n} \in \mathcal{S} \;\text{ if} \sum_{i=1}^{M} n_i b_i \leq K$$

The complete sharing policy is like the stochastic knapsack described in (Ross 1995). Foschini and Gopinath in (Foschini and Gopinath 1983) also use a a similar model for sharing memory in a multiprocessor system. Kaufman shows that the steady state solution for this model is of the form of Eqn (1). An example of a parameter of interest would be the probablity of the loss of messages of class i. To obtain this we will have to sum $\pi(\mathbf{n})$ over a subset of $\mathcal{S}$ that will do not have room for a new class i message.

Example 2: Consider a cellular system with M cells. Let n_i denote the number of active calls in cell i and the vector $\mathbf{n} = [n_1, n_2, \cdots, n_M]$ denote the state of the system. The set of admissible states for $\mathbf{n}$, $\mathcal{S}$, depends on the geographical layout of the cells and the channel assignment algorithm used. It can be shown that $\mathcal{S}$ is defined by

$$\mathbf{n} \in \mathcal{S} \quad \text{if} \quad B\mathbf{n} \leq \mathbf{N}$$

where B is a $M \times M$ matrix and $\mathbf{N}$ is an $M-$dimensional vector. The matrix B and the vector $\mathbf{N}$ are chosen to describe the form of the network, the number of channels in the system, the distribution of the channels and the channel assignment algorithm (Everitt 1994). To calculate the blocking probability P_{B_i}, of a call arriving to cell i, we first define the set $\mathcal{S}_{B_i} \subseteq \mathcal{S}$ that contains all $\mathbf{n}$ satisfying the blocking conditions for cell i. Then,

$$P_{B_i} = \frac{G(\mathcal{S}_{B_i})}{G(\mathcal{S})} \quad \text{where} \quad G(\mathcal{S}_{B_i}) = \sum_{\mathbf{n} \,\in\, \mathcal{S}_{B_i}} \left\{ \prod_{i=1}^{M} f_i(n_i) \right\}$$

Example 3: Consider a discrete time copy network like the one described in (Lee 1988). There are M inputs and in every slot, input i requests a random number, c_i, of copies. Let $f_i(c_i)$ be the probability that I requests c_i copies. The copy network can only make N copies in a slot. In the simplest design, if in a slot the sum of all the copy requests exceeds N, then only the first k inputs that satisfy the conditions

$$\sum_{i=1}^{k} c_i \leq N \quad \textbf{and} \quad \sum_{i=1}^{k+1} c_i > N$$

will be "served" and the requests from the other inputs, $k+1, K+2, \cdots N$, will be discarded. It is of interest to know the probability with which a request at input i will be served. It is also of interest to know the throughput of the copy network. Lee (Lee 1988) obtains a Chernoff bound on the probability of copy requests from i being served. We provide an exact analysis using techniques developed here.

In the context of queuing networks there have been attempts at providing transform based solutions to sum the product form solution over a state space where the total population in the system is a constant. In (Harrison 1985), Harrison reported a closed form expression for a closed queueing network with M single server nodes and N jobs in the network, i.e., this result only considers $f_i(n_i) = \rho_i^{n_i}$ and a state space defined by $\sum_{i=1}^{M} n_i = N$. Gordon (Gordon 1990) derived this result differently and extended it to closed queueing networks with multiple servers, i.e., different forms of $f_i(n_i)$ were allowed. Gerasimov (Gerasimov 1992) obtains results very similar to that obtained by Gordon and extends it to BCMP queueing networks with multiple classes of jobs but he restricts himself to closed queueing networks without infinite server queues. In all these results the case where the state space is specified by more than one constraint like in the examples given above has not been discussed. In this paper we obtain analytical expressions for the the sum of $\pi(\mathbf{n})$ over an irregular state space defined by a set of linear constraints with integere coefficients. Our interest is not to prove the existence of a product form solution to these systems. Rather, we assume the existence of the product form solution and obtain analytic expressions for the normalizing constant in these networks. In Section II we discuss the types of constraints that are applicable to our method and explore the single constraint case. In Section III we extend the algorithm to the multiple constraint case and discuss the issues in numerical evaluation using our technique.

II. Single Constraint

As described earlier, we assume that the system state is a vector $\mathbf{n} = [n_1, n_2, \cdots, n_M]$, an M-dimensional vector of integers and the steady state probability of the system is given by Eqn (1). We first consider the summing of $\pi(\mathbf{n})$ over the state space defined by a single constraint. The equality constraint that we discuss first corresponds to a closed queueing network and in this we generalise the result of Gordon (Gordon 1990). The "less than or equal" constraint that we consider next corresponds to the stochastic knapsack.

A. The Equality Constraint

Let us first consider a single equality constraint on the state space given by

$$\sum_{i=1}^{M} n_i = N \tag{2}$$

where N is a positive integer. Denote by $\mathcal{S}_{eq}(N,M)$ the set of all vectors $\mathbf{n}$ that satisfy the above constraint. Define $G(\mathcal{S}_{eq}(N,M))$ as

$$G\left(\mathcal{S}_{eq}(N,M)\right) \equiv \sum_{\mathbf{n}\in\mathcal{S}_{eq}(N,M)} \prod_{i=1}^{M} f_i(n_i)$$

In the context of a closed queueing network, $G(\mathcal{S}_{eq}(N,M))$ is the normalising constant in a network of M queues with a constant population of N.

Define $\delta(k)$ as

$$\delta(k) \equiv \begin{cases} 1 & \text{for } k=0 \\ 0 & \text{for } k \neq 0 \end{cases}$$

The above function, $\delta(k)$, can be represented by a contour integral on the complex plane as follows

$$\delta(k) = \oint z^{k-1} dz$$

where the integration is on the unit circle and z is a complex variable. (This can also be thought of as the inverse z-transform of $\delta(k)$, the discrete impulse function.)

Rewrite the constraint in Eqn (2), as $\sum_{i=1}^{M} n_i - N = 0$. $G\left(\mathcal{S}_{eq}(N,M)\right)$ can now be rewritten as

$$\begin{aligned} G\left(\mathcal{S}_{eq}(N,M)\right) &= \sum_{\mathbf{n}\in\mathcal{S}_{eq}(N,M)} \prod_{i=1}^{M} f_i(n_i) \\ &= \sum_{n_1=0}^{\infty} \cdots \sum_{n_M=0}^{\infty} \left[\prod_{i=1}^{M} f_i(n_i)\right] \delta(n_1+\cdots+n_M-N) \\ &= \sum_{n_1=0}^{\infty} \cdots \sum_{n_M=0}^{\infty} \left[\prod_{i=1}^{M} f_i(n_i)\right] \oint z^{(n_1+\cdots+n_M-N-1)}\, dz \\ &= \oint \left[\frac{1}{z^{(N+1)}}\right] \prod_{i=1}^{M} \mathcal{F}_i(z)\, dz \end{aligned}$$

Here $\mathcal{F}_i(z)$ is the z-transform of $f_i(n_i)$. Evaluation of the last integral is the sum of the residues of the integrand at the poles inside the unit circle of the integrand. This result is similar to that obtained by Gordon (Gordon 1990) for closed queueing networks except that we generalize the result to include any $f_i(n_i)$. We can further generalize Eqn (2) to include constraints of the form

$$b_1 n_1 + b_2 n_2 + \cdots + b_M n_M = N \tag{3}$$

where b_is and N are integers and $G\left(\mathbf{S}_{eq}(N,M)\right)$ is

$$G\left(\mathbf{S}_{eq}(N,M)\right) = \oint \left[\frac{1}{z^{(N+1)}}\right] \prod_{i=1}^{M} \mathcal{F}_i(z^{b_i})\, dz \tag{4}$$

Constraints of the type in Eqn (3) have been used by Kelly (Kelly 1979) to describe product form models for social interactions.

B. Inequality Constraints

We first look at the "less than or equal to ($\leq$)" type of inequality constraint like those used in defining the stochastic knapsack and then comment on the "greater than ($>$)" type of inequality constraint. Let $\mathcal{S}_{le}(N,M)$ be the set of $\mathbf{n}$ satisfying the constraint

$$b_1 n_1 + b_2 n_2 + \cdots + b_M n_M \leq N$$

where the b_is and N are integers. Denote by $\mathcal{S}_{le}(N,M)$ the set of all $\mathbf{n}$ that satisfy this constraint and let $G(\mathcal{S}_{le}(N,M))$ be defined as follows.

$$G\left(\mathcal{S}_{le}(N,M)\right) \equiv \sum_{\mathbf{n}\in\mathcal{S}_{le}(N,M)} \prod_{i=1}^{M} f_i(n_i)$$

Before we proceed we first define $\Phi_N(k)$ as follows,

$$\Phi_N(k) \equiv \begin{cases} 1 & \text{for } k \leq N \\ 0 & \text{for } k > N \end{cases}$$

$\Phi_N(k)$ is a delayed discrete step function reversed in time. The contour integral representation for this function is

$$\begin{aligned} \Phi_N(k) &= \sum_{i=0}^{N} \delta(k-i) \\ &= \oint \left[\frac{z^{(N+1)}-1}{z-1}\right] \left[\frac{z^k}{z^{(N+1)}}\right] dz \end{aligned}$$

Proceeding as before, we can obtain the following.

$$G(\mathcal{S}_{le}(N,M)) = \oint \left[\frac{z^{(N+1)}-1}{z-1}\right] \left[\frac{1}{z^{(N+1)}}\right] \prod_{i=1}^{M} \mathcal{F}_i(z^{b_i})\, dz \tag{5}$$

Note that Eqn (5) is identical to Eqn (4) except for the additional term, $(z^{N+1}-1)/(z-1)$ in the integrand.

We can obtain $G(\mathcal{S}_{le}(N,M))$ by adding another dimension to $\mathbf{n}$ and converting the inequality constraint to an equality constraint. This is similar to introducing a slack variable to convert an inequality constraint into an equality constraint in a linear programming problem.

If the state space is constrained by a "greater than ($>$)" inequality of the type

$$b_1 n_1 + b_2 n_2 + \cdots + b_M n_M > N$$

a similar technique cannot be applied because that would result in evaluating the contour integral over the unit circle of a function that has a pole at $z = 1$. In this situation we proceed as follows. Define the set of $\mathbf{n}$ satisfying the above constraint by $\mathcal{S}_{gt}(N, M)$. $G(\mathcal{S}_{gt}(N, M))$ be the sum of $\pi(\mathbf{n})$ over the set $\mathcal{S}_{gt}(N, M)$. Define $G_{tot}(N, M)$ as

$$G_{tot}(N, M) = \sum_{n_1=0}^{\infty} \cdots \sum_{n_M=0}^{\infty} \prod_{i=0}^{M} f_i(n_i) = \prod_{i=0}^{M} \mathcal{F}_i(1)$$

It can be seen that

$$G(\mathcal{S}_{gt}(N, M)) = G_{tot}(N, M) - G(\mathcal{S}_{le}(N, M))$$

where $G(\mathcal{S}_{le}(N, M))$ is the sum of $\pi(\mathbf{n})$ over the set of $\mathbf{n}$ satisfying the constraint $\sum_{i=1}^{M} b_i n_i \leq N$. The integral formula for $G(\mathcal{S}_{gt}(N, M))$ is derived as follows

$$\begin{aligned} G(\mathcal{S}_{gt}(N, M)) &= \oint \frac{\prod_{i=1}^{M} \mathcal{F}_i(1)}{z} - \\ &\quad \left[\frac{z^{(N+1)} - 1}{z^{N+1}(z-1)}\right] \prod_{i=1}^{M} \mathcal{F}_i(z^{b_i})\, dz \quad (6) \end{aligned}$$

III. Multiple Constraints

We now describe the method for summing $\pi(\mathbf{n})$ over a state space defined by multiple constraints by extending the method developed in Section II. We also discuss issues in the numerical evaluation using the techniques described here.

Let $\mathbf{C}$ be the set of constraints and without loss of generality, let the first p of these be equality constraints and the next $(q - p)$ be inequality constraints of the *less than or equal to* type. They are defined as follows.

$$\begin{aligned} \sum_{j=1}^{M} b_{ij} n_j &= N_i \qquad \text{for } i = 1 \cdots p \\ \sum_{j=1}^{M} b_{ij} n_j &\leq N_i \qquad \text{for } i = p+1 \cdots q \qquad (7) \end{aligned}$$

Let $\mathcal{S}(\mathbf{C}, M)$ the set of all vectors $\mathbf{n}$ satisfying the set of constraints in $\mathbf{C}$. Let $G(\mathcal{S}(\mathbf{C}, M))$ be defined as

$$G(\mathcal{S}(\mathbf{C}, M)) \equiv \sum_{\mathbf{n} \in \mathcal{S}(\mathbf{C}, M)} \prod_{i=1}^{M} f_i(n_i) \qquad (8)$$

Define for $i = 1, \cdots, p$

$$\hat{\delta}_i \equiv \delta(b_{i1}n_1 + b_{i2}n_2 + \cdots + b_{iM}n_M - N_i)$$

and for $i = p+1, \cdots, q$

$$\hat{\Phi}_i \equiv \Phi_{N_i}(b_{i1}n_1 + b_{i2}n_2 + \cdots + b_{iM}n_M)$$

Note that $\hat{\delta}_i$ and $\hat{\Phi}_i$ can be written as

$$\begin{aligned} \hat{\delta}_i &= \oint z_i^{(b_{i1}n_1 + b_{i2}n_2 + \cdots + b_{iM}n_M)} \left[\frac{1}{z_i^{N_i+1}}\right] dz \\ \hat{\Phi}_i &= \oint z_i^{(b_{i1}n_1 + b_{i2}n_2 + \cdots + b_{iM}n_M)} \left[\frac{z_i^{N_i+1} - 1}{z_i^{N_i+1}(z_i - 1)}\right] dz \end{aligned}$$

$G(\mathcal{S}(\mathbf{C}, M))$ can now be derived as

$$\begin{aligned} G(\mathbf{C}, M) &= \sum_{\mathbf{n} \in \mathcal{S}(\mathbf{C}, M)} \prod_{i=1}^{M} f_i(n_i) \\ &= \sum_{n_1=0}^{\infty} \cdots \sum_{n_M=0}^{\infty} \prod_{i=1}^{M} f_i(n_i) \hat{\delta}_1 \cdots \hat{\delta}_p \hat{\Phi}_{p+1} \cdots \hat{\Phi}_q \\ &= \oint \left[\frac{1}{z_1^{N_1+1}}\right] \cdots \oint \left[\frac{1}{z_p^{N_p+1}}\right] \\ &\quad \oint \left[\frac{z_p^{N_{p+1}+1} - 1}{z_{p+1}^{N_{p+1}+1}(z_{p+1} - 1)}\right] \cdots \oint \left[\frac{z_q^{N_q+1} - 1}{z_q^{N_q+1}(z_q - 1)}\right] \\ &\quad \prod_{i=1}^{M} \mathcal{F}_i(z_1^{b_{1i}} z_2^{b_{2i}} \cdots z_q^{b_{qi}}) dz_1 \cdots dz_q \qquad (9) \end{aligned}$$

In the above expression, the first p contour integrals correspond to the first p equality constraints. The subsequent $(q-p)$ integrals correspond to the *less than or equal to* type of constraints. The $\mathcal{F}_i(z)$ are the z-transforms of $f_i(n_i)$ and the contour integrals are evaluated over the unit circle.

A. Evaluation of the Contour Integrals

We now consider the evaluation of the contour integral. All integrals are evaluated on the unit circle. From the residue theorem, for any function of the complex variable z and a closed contour C in the complex plane,

$$\oint_C \mathcal{F}(z) = \sum \text{residues of } \mathcal{F}(z) \text{ at poles inside } \mathcal{C}$$

Our contour of integration is the unit circle. Note that if the $\mathcal{F}_i(z)$ have poles inside the unit circle, they can be suitably scaled to move them out of it. In Eqn (9) the only poles of the integrand inside the unit circle corresponding to the i^{th} integration is at $z_i = 0$. The order of this pole is $(N_i + 1)$. Therefore, to evaluate the integrals in the expression for $G(\mathbf{C}, M)$ in Eqn (9) we need to evaluate the residues of the functions at $z_i = 0$ $(i = 1 \cdots q)$. Thus, the evaluation of $G(\mathbf{C}, M)$ can be accomplished by the following algorithm

$G_0 = \prod_{i=1}^{M} \mathcal{F}_i(z_1^{b_{1i}} z_2^{b_{2i}} \cdots z_s^{b_{qi}})$;

for k **from** 1 **to** p **do**

G_k =**residue**$\left(G_{k-1} * \left[\frac{1}{z_k^{N_k+1}}\right], z_k = 0\right)$;

for k **from** $p+1$ **to** q **do**

G_k = **residue**$\left(G_{k-1} * \left[\frac{1}{z_k^{N_k+1}}\right]\left[\frac{z_k^{N_k+1}-1}{z_k-1}\right], z_k = 0\right)$;

$G(\mathbf{C}, M) = G_q$;

The evaluation of these residues involves only differentiation and taking the limit of the derivative as $z_i \to 0$. *i.e.* at the k^{th} step in the above algorithm, the residue is given by (see for example (Rubenfeld 1985))

$$\text{residue}\left(G_{k-1} * \left[\frac{1}{z_k^{N_k+1}}\right], z_k = 0\right) = \lim_{z_k \to 0} \frac{1}{N_k!} \frac{d^{N_k}}{dz_k^{N_k}} [G_{k-1}] \quad \text{for } k = 1 \cdots p \quad (10)$$

$$\text{residue}\left(G_{k-1} * \left[\frac{1}{z_k^{N_k+1}}\right]\left[\frac{z_k^{N_k+1}-1}{z_k-1}\right], z_k = 0\right) = \lim_{z_k \to 0} \frac{1}{N_k!} \frac{d^{N_k}}{dz_k^{N_k}} \left[G_{k-1} * \left[\frac{z_k^{N_k+1}-1}{z_k-1}\right]\right] \quad \text{for } k = (p+1) \cdots q (11)$$

The total number of differentiations at the k^{th} step in the above algorithm is N_k and the algorithm needs $\sum_{k=1}^{q} N_k$ differentiations to evaluate $G(\mathbf{C}, M)$. Note that left hand side of Eqns (10) and (11) is the evaluation of the N_k^{th} coefficient of the Taylor series of G_{k-1} and $G_{k-1}(z_k^{N_k+1} - 1)/(z_k - 1)$ respectively.

If $\mathcal{F}_i(z)$ can be represneted as a ratio of polynomials, then we can have a partial fraction expansion of G_0 and there are well known techniques to evaluate the residues in this case (see for example (D'Azzo and Houpis 1981)). The evaluation of the residue in this situation is very efficient and can be shown to be independent of M and N_k and Eqn (9) will have a closed form expression.

IV. Application Examples

In this sections we consider three examples for the application of our technique. First, we obtain an analytical model to obtain the blocking probability in any cell of a finite highway cellular system. Next we use this technique to obtain an exact analysis of the blocking probability in the copy network of the Lee Multicast Switch. Finally, we consider application of this technique in solving a BCMP queueing network

A. Analysis of a Highway Cellular System

Consider a one dimensional highway cellular communication system with M cells and K channels. We assume that each cell, except the first and the last cell, has exactly two neighbors corresponding to the cells on the left and right. A channel that is being used in a cell is not available for use in that cell and its neighbors. If there are K channels available in the system, then an incoming call is accepted into the system if a channel can be found that is not being used in the cell or in any of its neighbors.

We consider a system that uses maximum packing strategy for channel assignment and has no handoff calls. We assume that the call arrival process to a cell is Poisson with rate λ and call duration is arbitrarily distributed with mean 1. Let the state of the system be denoted by the vector $\mathbf{n} = [n_1 \cdots n_M]$, where n_i is the total the number of active calls in cell i. The solution for the steady state probability for the state of the system has a product form and is (Everitt and MacFayden 1983)

$$\text{Prob}(\mathbf{n}) = \frac{1}{G} \prod_{i=1}^{M} \frac{\lambda_i^{n_i}}{n_i!}$$

From (Everitt and MacFayden 1983), the constraints on the state space for this system can be written as

$$n_i + n_{i+1} \leq K \quad \text{for } i = 1 \cdots M-1 \qquad (12)$$

An incoming call to an internal cell i is blocked if the total number of calls in cell i and at least in one of its neighbors, $i-1$ or $i+1$, is K. The set of states, $\mathbf{n}$, in this situation can be represented by additional boundary conditions

$$n_i + n_{i+1} = K \qquad (13)$$
$$n_i + n_{i-1} = K \qquad (14)$$

Denote by $\mathcal{S}_{B1}$ the set of all states, $\mathbf{n}$, that satisfy constraints (12) and (13), by $\mathcal{S}_{B2}$ the set of all states, $\mathbf{n}$, that satisfy constraints (12) and (14) and by $\mathcal{S}_{B12}$ the set of all states, $\mathbf{n}$, that satisfy constraints (12), (13) and (14). The set $\mathcal{S}_{B12}$ is the intersection of sets $\mathcal{S}_{B1}$ and $\mathcal{S}_{B2}$. Blocking occurs if the state is in set $\mathcal{S}_{B1}$ or set $\mathcal{S}_{B2}$ or both. Therefore, when the system is in state $\mathbf{n}$, the probability, P_B, that a call arriving in cell i is blocked is given by

$$P_B = \text{Prob}(\mathbf{n} \in \mathcal{S}_{B1}) + \text{Prob}(\mathbf{n} \in \mathcal{S}_{B2}) - \text{Prob}(\mathbf{n} \in \mathcal{S}_{B12})$$

Define

$$G_{B1} \equiv \sum_{\mathbf{n} \in \mathcal{S}_{B1}} \prod_{i=1}^{M} \frac{\lambda_i^{n_i}}{n_i!}$$
$$G_{B2} \equiv \sum_{\mathbf{n} \in \mathcal{S}_{B2}} \prod_{i=1}^{M} \frac{\lambda_i^{n_i}}{n_i!}$$
$$G_{B12} \equiv \sum_{\mathbf{n} \in \mathcal{S}_{B12}} \prod_{i=1}^{M} \frac{\lambda_i^{n_i}}{n_i!}$$

The blocking probability at the internal cell i is given by

$$P_B = \frac{G_{B1} + G_{B2} - G_{B12}}{G}$$

From this, we can obtain the blocking probabilities in the any cell for various values of λ, M, and K. We can use a symbolic computation package like Maple or Mathematica and using the method described in this paper, we can obtain the symbolic expression for the residues in the $(M-1)$ steps of the algorithm of section III-A. Once the residues are evaluated and the expression for the blocking probability derived, the blocking probabilities for any call arrival rate, λ, can be easily evaluated. For comparison with a brute force enumeration approach note that we will need to first determine if a vector $\mathbf{n}$ is in the state space defined above and if it does, the product form expression corresponding to this state is computed and added to obtain Gs. This is computationally very expensive.

B. Analysis of Copy Networks of Multicast Switches

Now consider Example 3 in Section 1 of a discrete time copy network. We now apply the techniques discussed in this paper to the analysis of a copy network of the kind described in Section 1. Consider a copy network that has M inputs and can make upto N copies in each slot, an $M \times N$ copy network. In any slot, the sum of the number of copies requested by the active inputs, inputs with requests, may exceed N. In the simplest model the first i inputs that satisfy the conditions $\sum_{j=1}^{i} c_j \leq N$ and $\sum_{j=1}^{i+1} c_j > N$ are served and those requests that cannot be served are lost (c_j is the number of copies requested by input j). In the following we show how to calculate the probability that a request from input i is lost. If the requests that cannot be served are queued to be served in a subsequent slot we will also show how to do a queueing analysis using the techniques described in this paper.

Let X_i be the number of copies requested by input i and $f_i(x_i)$ its probability mass function, and $\mathcal{F}_i(z)$ the moment generating function of $f_i(x_i)$. The copy request of input i is served if $X_1 + X_2 + \cdots + X_i \leq N$. The probability of loss at port i, $P_{loss}(i)$, is then given by

$$P_{loss}(i) = 1 - \sum_{\sum_{j=1}^{i} x_j \leq N} \prod_{j=1}^{i} f_j(x_j)$$

The summation on the RHS of the above equation is carried out over all possible combinations of copy requests from ports 1 to i that sum to less than or equal to N. Therefore, following Section II-B, we can obtain the $P_{loss}(i)$s as follows

$$\begin{aligned} 1 - P_{loss}(i) &= \sum_{x_1=0}^{N} \cdots \sum_{x_i=0}^{N} \prod_{k=1}^{i} f_k(x_k) \Phi_N(x_1 + \cdots + x_i) \\ &= \oint \left[\frac{z^{(N+1)} - 1}{z - 1} \right] \left[\frac{1}{z^{(N+1)}} \right] \prod_{k=1}^{i} \mathcal{F}_i(z) dz \end{aligned}$$

Now consider the case when the requests that cannot be served are queued. Although many scheduling policies can be formulated and analysed, to illustrate the use of our technique, we will consider the simplest scheduling policy in which the requests satisfying the conditions mentioned earlier are served queued and the others are queued. Proceeding sequentially from input 1, all the input ports whose copy requests can fully served in the slot are selected for service. Note that this policy selects a packet for service only if all the copies requested by it can be generated in the given slot. Since service always starts from port 1, the service rate varies with the port number and decreases as the port address increases.

The effective service rate at an input port, the rate at which the copy requests can be actually served, depends on the arrival processes and effective service rates at the preceding ports. We model each input port as a discrete time M/M/1 queue except for the first port which always gets to be served in every slot. At port i, let the effective service rate and the probability of its input queue being empty be denoted by $\mu(i)$ and $P_{0,i}$ respectively. The number of ports that can be served in a slot depends on the copy requests of the packets at the head of the queues. Let $f_{H,i}(k)$ be the probability mass function (pmf) of the number of copies requested by the packet at the head of input queue i. From our definition above, $1 - P_{0,i}$ is the probability that the head of queue i is non empty. Hence the probability mass function of the copies requested by a packet at the head of the queue i will be,

$$f_{H,i}(x_i) \equiv \text{Prob}\{X_i = x_i\} = \begin{cases} P_{0,i} & x_i = 0 \\ (1 - P_{0,i})\, q(x_i) & x_i > 0 \end{cases}$$

with $\mathcal{F}_{H,i}(z)$ as its moment generating function. Since port 1 is always served in each slot irrespective of the copy requests of the other ports, $\mu(1) = 1.0$. Now, for ports $i = 2, \cdots M$, if port i requests k copies, its request will be served in the slot only if the sum of the copies requested by ports 1 to $i-1$ is less than or equal to $N-k$. Thus,

$$\begin{aligned} \mu(i+1) &= \text{Prob}\{\text{a pkt at port } i+1 \text{ is served }\} \\ &= \sum_{k=1}^{N} q(k) \text{ Prob} \left\{ \sum_{j=1}^{i} X_j \leq N - k \right\} \\ &= \sum_{k=1}^{N} q(k) \left[\sum_{\sum_{j=1}^{i} x_j \leq (N-k)} \prod_{j=1}^{i} f_{H,j}(x_j) \right] \end{aligned}$$

Using the results of Section II-B, it can be shown that

$\mu(i+1)$ is given by

$$\mu(i+1) = \sum_{k=1}^{N} q(k) \oint \left[\frac{z^{(N-k+1)}-1}{z^{N-k+1}(z-1)}\right] \prod_{j=1}^{i} \mathcal{F}_{H,j}(z) dz$$

Using the standard results for discrete time M/M/1 queues results, eg. (Woodward 1993) we may calculate the associated waiting times and other parameters.

We have also used the method described in this paper to analyse various other scheduling policies for the copying process in the copy network (Sikdar 1998).

C. BCMP Networks

The product form solution has been extended to a very general class of queueing networks by Baskett et al in (F. Baskett and Palacios 1975). Such queueing networks are called BCMP networks in literature. Four types of service centers and multiple classes of jobs, each with a different service requirement and routing probabilities, were permitted. We assume that there are M nodes in the queueing network and that there are R classes of jobs. We denote by n_{ir} the number of class r jobs in node i. The state of node i will be denoted by $\mathbf{y_i}$ where

$$\mathbf{y_i} = [n_{i1}, n_{i2} \cdots n_{iR}]$$

We denote the state of the queueing network by the vector $\mathbf{Y}$ which is defined to be

$$\mathbf{Y} = [\mathbf{y_1}, \mathbf{y_2} \cdots \mathbf{y_M}]$$

In this section, we will describe a technique, similar to the one used for single class networks in section III, to derive the normalizing constant. As before, we will assume that the constraints on the state space for the system are linear equalities and inequalities, except that instead of n_i, we will have n_{ir} as the variables.

$$\sum_{i=1}^{M}\sum_{r=1}^{R} b_{j,ir} n_{ir} = N_j \qquad \text{for } j = 1 \cdots p$$
$$\sum_{i=1}^{M}\sum_{r=1}^{R} b_{j,ir} n_{ir} \leq N_j \qquad \text{for } j = p+1 \cdots q$$

Let $\mathbf{C}$ be the above set of constraints and $\mathcal{S}(\mathbf{C}, M, R)$ the set of $\mathbf{Y}$ satisfying these constraints. Without loss of generality, we will assume that the nodes $1 \cdots \hat{M}$ are FCFS, PS or LCFS queues and nodes $\hat{M}+1 \cdots M$ are IS queues. From (Gelenbe and Mitrani 1980), if we define $\hat{f}_i(\mathbf{y_i})$ to be

$$\hat{f}_i(\mathbf{y_i}) \equiv \begin{cases} n_i! \prod_{r=1}^{R} f_{ir}(n_{ir}) & \text{for } i = 1 \cdots \hat{M} \\ \prod_{r=1}^{R} f_{ir}(n_{ir}) & \text{for } i = \hat{M}+1 \cdots M \end{cases}$$

the steady state probability of the system will be

$$\begin{aligned} \text{Prob}(\mathbf{Y}) &= \frac{1}{G_{bcmp}(\mathbf{C}, M, R)} \prod_{i=1}^{M} \hat{f}_i(\mathbf{y_i}) \\ &= \frac{1}{G_{bcmp}(\mathbf{C}, M, R)} \left[\prod_{i=1}^{\hat{M}} n_i! \prod_{r=1}^{R} f_{ir}(n_{ir})\right] \left[\prod_{i=\hat{M}+1}^{M} \prod_{r=1}^{R} f_{ir}(n_{ir})\right] \end{aligned}$$

where $n_i = \sum_{r=1}^{R} n_{ir}$ and $G_{bcmp}(\mathbf{C}, M, R)$ is the normalizing constant for the network obtained as follows

$$G_{bcmp}(\mathbf{C}, M, R) = \sum_{\mathbf{Y} \in \mathcal{S}(\mathbf{C}, M, R)} \left[\prod_{i=1}^{\hat{M}} n_i! \prod_{r=1}^{R} f_{ir}(n_{ir})\right] \left[\prod_{i=\hat{M}+1}^{M} \prod_{r=1}^{R} f_{ir}(n_{ir})\right] \qquad (15)$$

Notice that this is similar to the normalizing constant of single class networks except that there are MR terms in the product rather than M and there is an additional $n_i!$ term for each non-IS node. To simplify this, consider the Euler integral

$$n! = \int_0^\infty e^{-t} t^n dt$$

Substituting this for $n_i!$ in Eqn (15), we get

$$\begin{aligned} &G_{bcmp}(\mathbf{C}, M, R) = \\ &\sum_{\mathbf{Y} \in \mathcal{S}(\mathbf{C}, M, R)} \left[\prod_{i=1}^{\hat{M}} \int_0^\infty e^{-t}\, t^{\sum_{r=1}^{R} n_{ir}} dt \prod_{r=1}^{R} f_{ir}(n_{ir})\right] \left[\prod_{i=\hat{M}+1}^{M} \prod_{r=1}^{R} f_{ir}(n_{ir})\right] \\ &= \sum_{\mathbf{Y} \in \mathcal{S}(\mathbf{C}, M, R)} \left[\prod_{i=1}^{\hat{M}} \int_0^\infty e^{-t} \prod_{r=1}^{R} t^{n_{ir}} f_{ir}(n_{ir})\, dt\right] \left[\prod_{i=\hat{M}+1}^{M} \prod_{r=1}^{R} f_{ir}(n_{ir})\right] \qquad (16) \end{aligned}$$

Eqn (16) is similar to Eqn (8) and the techniques developed in section III can be used. The only difference is that there is an additional integration for every node that is not a IS queue. Using the same technique that was used in the derivation of Eqn (9) we obtain

$$G_{bcmp}(\mathbf{C}, M, R) =$$

$$\oint \left[\frac{1}{z_1^{N_1+1}}\right] \cdots \oint \left[\frac{1}{z_p^{N_p+1}}\right] \oint \left[\frac{z_p^{N_{p+1}+1}-1}{z_{p+1}^{N_{p+1}+1}(z_{p+1}-1)}\right]$$
$$\cdots \oint \left[\frac{1}{z_q^{N_q+1}}\right] \left[\frac{z_q^{N_q+1}-1}{z_q-1}\right]$$
$$\left[\prod_{i=1}^{\hat{M}} \int_0^\infty e^{-t} \prod_{r=1}^{R} \mathcal{F}_{ir}(t z_1^{b_{1,ir}} \cdots z_q^{b_{q,ir}})\, dt\right]$$
$$\left[\prod_{i=\hat{M}+1}^{M} \prod_{r=1}^{R} \mathcal{F}_{ir}(z_1^{b_{1,ir}} \cdots z_q^{b_{q,ir}})\right] dz_1 \cdots dz_q \qquad (17)$$

where $\mathcal{F}_{ir}(z)$ is the z-transform of $f_{ir}(n_{ir})$. Here, we have used the property that the z-transform of $a^n f(n)$ is $\mathcal{F}(az)$.

V. Discussion and Conclusion

The important contribution of this paper is the development of a method to derive analytical expression using $z-$transforms and contour integrals to obtain the sum of a product form expression over an irregular $M-$dimensional integer space defined by linear constraints with integer coefficients. This method can be used in many situations like calculating the normalising constant in separable queueing networks or in models that deal with vectors of independent integer random variables like the one on copy networks described here.

As is evident, the problem that we address is similar to the evaluation of normalising constant in product form queueing networks over irregular state spaces. Many techniques for this that have been reported in literature are usually algorithmic. They assume specific forms for $f_i(n_i)$ and also have only one constraint on the state space. There have also been a few attempts at developing analytical expressions but these too consider very special cases. We have obtained a generalised expression that can be in almost all cases of practical interest. Specifically, note that the results from (Harrison 1985; Gordon 1990) can be considered special cases of Eqn (4) and the result from (Gerasimov 1992) can be considered to be a special case of Eqn (17) with one equality constraint and no infinite server nodes.

Finally, we wish to mention that in (Nelson 1993), Nelson discusses many other models, such as genetic and polymerization models, which have a solution which also need a summing of a product form expression over an irregular multidimensional integer space. The method reported in this paper can also be applied to such systems.

References

Buzen, J. P. (1973). Computational algorithms for closed queueing networks with exponential servers. *Communications of the ACM 16*, 527–531.

D'Azzo, J. J. and C. H. Houpis (1981). *Linear Control System: Analysis and Design*. McGraw-Hill Book Company.

Everitt, D. E. (1994). Traffic engineering of the radio interface for cellular mobile networks. In *Proceedings of the IEEE*, Volume 82, pp. 1372–1382. IEEE.

Everitt, D. E. and N. W. MacFayden (1983). Analysis of multicellular mobile radio telephone systems with loss. *British Telecom Journal 1*, 37–45.

F. Baskett, K. M. Chandy, R. R. M. and F. G. Palacios (1975). Open, closed and mixed networks of queues with different classes of customers. *Journal of the ACM 22*, 248–260.

Foschini, G. J. and B. Gopinath (1983). Sharing memory optimally. *IEEE Transactions on Communications COM-31*, 352–360.

Gelenbe, E. and I. Mitrani (1980). *Analysis and Synthesis of Computer Systems*. Academic Press.

Gerasimov, A. I. (1992). Integral method for calculating closed queueing networks. *Problems of Information Transmission 28*, 184–194.

Gordon, J. J. (1990). The evaluation of normalizing constants in closed queueing networks. *Operations Research 38*, 863–869.

Harrison, P. G. (1985). On normalizing constants in queueing networks. *Operations Research 33*, 464–468.

Kaufman, J. (1981). Blocking in a shared resource environment. *IEEE Transactions on Communications COM-29*, 1474–1481.

Kelly, F. P. (1979). *Reversibility and Stochastic Networks*. Chichester: Wiley.

Lee, T. T. (1988). Nonblocking copy networks for multicast packet switching. *IEEE Journal on Selected Areas Communications 6*, 1455–1467.

Nelson, R. (1993). The mathematics of product form queuing networks. *ACM Computing Surveys 23*, 339–369.

Ross, K. W. (1995). *ultiservice Loss Models for Broadband Telecommunication Networks*. Berlin: Springer.

Rubenfeld, L. A. (1985). *A First Course in Applied Complex Variables*. Wiley.

Sikdar, B. (1998). Queueing analysis of scheduling policies in copy networks of space based multicast atm switches. Master's thesis, Indian Institute of Technology, Kanpur, Dept. of Electrical Engineering.

Woodward, M. E. (1993). *Communication and Computer Networks : Modelling with Discrete-Time Queues*. London: Pentech Press.