Applications

A Recursive Algorithm for Generating the Transition Matrices of Multistation Series Production Lines

H.T. Papadopoulos and M.E.J. O'Kelly
Department of Industrial Engineering, University College, Galway, Ireland

This paper is concerned with reliable multistation series production lines, where an operation is performed on each job by the single machine at each station and jobs for the first station arrive according to a Poisson distribution. The processing times of all the stations are exponentially distributed and no buffers are allowed between successive stations.

The structure of the transition matrices of these specific types of production lines is examined and a one-stage recursive algorithm is provided for generating them. The transition matrices are block-structured and sparse and by applying the proposed algorithm, one can create the transition matrix of a K-station line from the $(K-1)$-station transition matrix. This process avoids the necessity of writing down explicitly all the feasible states and transitions of the model, which is tedious and time-consuming especially for long production lines (i.e. for $K = 12$ stations, the number of states is 46,368).

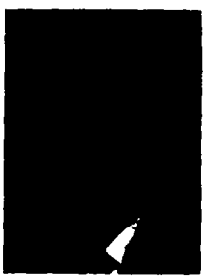
Keywords: Multistation production lines, Open queueing networks, Finite buffers, Blocking phenomenon, Quasi-Birth-Death process, Block-triagonal matrices, large sparse matrices, Matrix geometric form.

1. Introduction

Production lines of the tandem type may be modeled by applying the queueing network theory. Buzacott and Yao [2] presented an excellent review of analytical models of Flexible Manufacturing Systems (FMS).

When the capacity of the intermediate buffers is finite, the blocking phenomenon must be considered but this contributes significantly to the intractability of the exact analysis. An extensive bibliography on the analysis of queueing networks

H.T. Papadopoulos (Chris) is a Lecturer in Industrial Engineering and Operations Research at University College, Galway. He is a graduate in Mathematics and Computer Science from Aristotle University of Thessaloniki and National University of Athens. He has just completed his PhD degree in the exact analysis of production lines at Galway (National University of Ireland). His interests include the application of mathematical techniques in the analysis of Manufacturing Engineering Systems. His studies in Galway were supported by the Greek and Irish Governments. He was also the first recipient of the prestigious McLoughlin Bursary in Engineering in the College.



M.E.J. O'Kelly (Eddie) is Professor of Industrial Engineering and former Dean of the Faculty of Engineering, University College Galway. He is an electrical engineering graduate of the National University of Ireland (University, College Cork) and holds graduate degrees from California Institute of Technology and Columbia University, New York. Prior to joining University College, Galway, he worked in the electronics and engineering industry in France, Germany and Ireland and was Head of Manpower Forecasting in the Department of Labour, Dublin. He is a consultant to a number of multinational and national companies in Operations Research and Manufacturing Strategy.

Elsevier Science Publishers B.V.
Computers in Industry 12 (1989) 227-240

0166-3615/89/$3.50 © 1989 Elsevier Science Publishers B.V.

with blocking is given by Perros [10]. This interest came from their usefulness in modeling the behaviour of computer systems, telecommunication systems and more recently flexible manufacturing systems (FMS).

Exact analytical results have been reported only for a few cases involving either short lines with fairly general assumptions in regard to inter-arrival and processing times [1,6] or longer production lines with very restrictive assumptions. Hunt [5] and Hillier and Boling [4] confine their attention to exponentially or Erlang distributed processing times. The exponential assumption permits the establishment of a set of simultaneous linear equations in terms of the equilibrium probabilities of each state. These may then be solved to give the mean production rates. However, even in this case, the computation soon becomes formidable as the number of states grow rapidly with increases in the number of stages. For this reason, approximate methods of some particular configurations of open or closed queueing networks with blocking have been developed and the results were compared against simulation data (see [11] among others).

It is the purpose of this paper to provide a recursive algorithm for generating the transition matrices of long exponential production lines, thus allowing the use of existing software packages on operations on large sparse matrices, to obtain solutions of the derived large linear systems in an efficient fashion.

2. The Model

The model to be discussed is a production line consisting of K stations arranged in series, where an operation is performed on each job by the single machine in each station. A buffer of infinite capacity is assumed to exist in front of the first station, with jobs arriving according to a homogeneous Poisson distribution with rate λ. Each item enters the line at station 1, passes through all stations in order and leaves the Kth station (last) in finished form. We also allow a departing item, from the last machine, to return instantaneously to the end of the input queue (first) with a probability θ: $0 \leqslant \theta < 1$. This latter process is called feedback. With probability $\theta' = 1 - \theta$, items depart the system after being processed at station K. The processing (service) times S_i, $i = 1, 2, \ldots, K$ are independent exponentially distributed random variables with mean values $1/\mu_i$, $i = 1, 2, \ldots, K$, which are in general not identical (i.e. $\mu_i \neq \mu_j$ for $i \neq j$). We assume that the machines (stations) are reliable and all jobs in any machine are processed in a FIFO (first-in, first-out) manner. No buffers are allowed between successive stations.

Let C_i, $i = 1, 2, \ldots, K$, be the queue capacity of the ith queue (including the item in service). Then in our model C_1 is infinite, and $C_j = 1$ for $j = 2, 3, \ldots, K$. The state of the tandem configuration can be described by the vector

$$(c_1, c_2, \ldots, c_K), \tag{1}$$

where $c_1 = 0, 1, 2, \ldots$ and $c_j = 0, C_j, C_j + 1$ for $j = 2, 3, \ldots, K$, i.e. $c_j = 0, 1, 2$ for $j = 2, 3, \ldots, K$.

The random variable c_j indicates the number of items in the jth queue, which may vary from 0 to 2. The value $c_j = C_j + 1$ is used to indicate that the jth queue is full and the $(j-1)$th machine is blocked, i.e. it cannot provide service on a job, since the previous job cannot move on to the next station, because that station is busy or blocked. Of course, the last station (Kth) can never be blocked.

The state of the system under consideration is a two-dimensional stochastic process $N(t) = [N_1(t), N_2(t)]$. Both coordinate random variables are integer valued and nonnegative. $N_1(t)$ represents the number of jobs in the first queue at time t and N_1 is the number of jobs in the first queue at equilibrium (the limit of $N_1(t)$, as t tends to infinity). There is no upper limit for N_1. N_2 represents the state of the remaining subnetwork (consisting of the queues $2, 3, \ldots, K$) at equilibrium and may assume only values from 1 to m, for some finite m which is the number of states in the subnetwork. The changes in the state of the system are caused by the occurrence of various events. The occurrence times for all events have negative exponential distributions with strictly positive means. Thus the process is Markovian. Its state-space is $S = \{(i, j): i \geqslant 0, 1 \leqslant j \leqslant m\}$ with the index i specifying the total number of items (jobs) queued up or in service in the first machine (Unit-I). Such items are called "I-customers". The index j determines the state of the subnetwork (Unit-II). Upon completion of a service in Unit-I, an item becomes a "II-customer". Now, as shown by Evans [3], Wallace [13] and Latouché and Neuts [7], the transition matrix P that describes the model, has

the following block tri-diagonal form

$$P = \begin{vmatrix} A_{01} & A_0 & & & & \\ A_2 & A_1 & A_0 & & & \\ & A_2 & A_1 & A_0 & & \\ & & A_2 & A_1 & A_0 & \\ & & & \cdot & \cdot & \cdot \\ & & & & \cdot & \cdot \end{vmatrix} \quad (2)$$

and the equilibrium equations $\pi P = 0$ can be expressed in matrix-difference form as

$$\pi_k A_0 + \pi_{k+1} A_1 + \pi_{k+2} A_2 = 0, \quad (3)$$

for $k = 0, 1, 2, \ldots$ and

$$\pi_0 A_{01} + \pi_1 A_2 = 0, \quad (4)$$

for the boundary equations.

A_0 is an $(m \times m)$-matrix describing the transitions in the subnetwork, which simultaneously produce inputs to the first queue;

A_1 is an $(m \times m)$-matrix describing transitions in the subnetwork which produce neither inputs to nor outputs from the first queue assuming that the queue is *not empty*;

A_2 is an $(m \times m)$-matrix describing transitions in the subnetwork which simultaneously produce outputs from the first queue; and

A_{01} is an $(m \times m)$-matrix describing transitions in the subnetwork which produce neither inputs to nor outputs from the first queue, assuming that the queue is *empty*.

A Markov chain whose equilibrium equations have the form of (3) and (4) was termed by Wallace a Quasi-Birth and Death (QBD) process. Wallace [12] has shown that, under very weak conditions, a unique suitable root exists and that Evans' algorithm does indeed converge to it. Evans [3] proposed that a "geometric" solution, i.e. solution of the form

$$\pi_k = \pi_0 R^k, \quad k \geq 0, \quad (5)$$

exists under certain conditions, and he offered an algorithm based on finding a root-matrix, R^*, which is the minimal nonnegative solution to the matrix quadratic equation

$$R^2 A_2 + R A_1 + A_0 = 0. \quad (6)$$

Papadopoulos and O'Kelly [9] applied a modified version of this algorithm and they found the mean queue-length and the marginal probabilities for the first infinite queue of the system under consideration, for $K = 2, 3, \ldots, 7$.

Latouché and Neuts [7] proposed a different algorithm for finding the matrix R.

Remarks. We define $p_2 = [p_2(j): j = 1, 2, \ldots, m]$ to be the marginal probability density function for the subnetwork (Unit-II), which has elements

$$p_2(j) = \text{Prob}[N_2 = j] = \sum_{i=0}^{\infty} \pi_{i,j},$$

$$j = 1, 2, \ldots, m, \quad (7)$$

where $\pi_{i,j}$ are the *joint probabilities*. Solving the system

$$p_2 A = 0, \qquad p_2 e = 1, \quad (8)$$

where A is the conservative stable matrix given by

$$A = A_0 + A_1 + A_2 \quad (9)$$

and e is a $(m \times 1)$ column vector, with all elements equal to 1, will give explicit results for $p_2(j)$, $j = 1, 2, \ldots, m$.

The Equilibrium Condition is given [7] by

$$p_2 A_2 e > p_2 A_0 e \quad (10)$$

From this relationship the critical input rate (λ^*) to the system can be determined. In the steady-state, this critical input rate is identical to the maximum throughput rate of the production line.

3. The Algorithm

In this section, we first discuss the problem of how to determine the number of the states in the subnetwork (Unit-II), for the model described above. We derive a recursive formula which gives the number of states of the subnetwork for any K-station production line. Next, we describe explicitly the form of the matrices A_0, A_1, A_{01} and A_2 for $K = 2$ and $K = 3$, and by comparing the relationship of their sub-blocks we provide the recursive algorithm for generating them.

3.1. The Number of States

Since the first queue has an infinite drawing population, the number of states of the whole system is also infinite. We are interested in determining the number of states m_K of the subnetwork (consisting of the queues $2, 3, \ldots, K$). The

vector that describes the subnetwork is $(c_2, c_3, \ldots, c_K)$, where every c_j, $j = 2, 3, \ldots, K$, takes $\nu = 3$ values: from 0 up to 2 $(= C_j + 1)$. The value $c_j = 2$ means that the jth machine is full and the $(j-1)$th machine is blocked. The number of permutations of ν distinct elements taken in groups of $k_1 = K - 1$, the number of stations of the subnetwork, is $\nu^{k_1} = 3^{K-1}$. So, for example, the number of possible permutations of a two-station series production line is $3^{2-1} = 3$; for $K = 3$ and $K = 4$ the number of permutations are $3^{3-1} = 9$ and $3^{4-1} = 27$, respectively. But not all these permutations are *feasible*, since for example the situation (0, 2) is inconsistent with the definition of the state 2 (the second machine (of the subnetwork) is full and the first one is blocked, but here the first machine is idle (state 0)). Thus, we have to remove this permutation. For the 2-station line there is no such a problem. The number of the (feasible) states of the subnetwork, m_2, equals the number of the possible permutations (3). Let us consider the 3-station line: the nine possible permutations are

$$S_{0,3} = \{(0, 0), (0, 1), (0, 2)\},$$

$$S_{1,3} = \{(1, 0), (1, 1), (1, 2)\},$$

$$S_{2,3} = \{(2, 0), (2, 1), (2, 2)\}.$$

This set of 9 permutations consists of 3 subsets: $S_{0,3}$, $S_{1,3}$, $S_{2,3}$, depending on the state (0, 1 or 2) of the first station of the subnetwork, which is the second for the whole system. The permutation (0, 2) is not feasible and so it is rejected. Thus, $m_3 = 9 - 1 = 8$.

For $K = 4$ the 27 possible permutations are

$$S_{0,4} = \{(0, 0, 0), (0, 0, 1), (0, 0, 2), (0, 1, 0), (0, 1, 1), (0, 1, 2), (0, 2, 0), (0, 2, 1), (0, 2, 2)\},$$

$$S_{1,4} = \{(1, 0, 0), (1, 0, 1), (1, 0, 2), (1, 1, 0), (1, 1, 1), (1, 1, 2), (1, 2, 0), (1, 2, 1), (1, 2, 2)\},$$

$$S_{2,4} = \{(2, 0, 0), (2, 0, 1), (2, 0, 2), (2, 1, 0), (2, 1, 1), (2, 1, 2), (2, 2, 0), (2, 2, 1), (2, 2, 2)\}.$$

The second $(S_{1,4})$ and third $(S_{2,4})$ subsets each consist of $8 = m_3$ states ((1, 0, 2) and (2, 0, 2) are not feasible), while the first subset $(S_{0,4})$ consists of only 5 states ((0, 0, 2), (0, 2, 0), (0, 2, 1) and (0, 2, 2) are not feasible), which correspond to the states belonging to $S_{0,3}$ and $S_{1,3}$ with the addition of 0 in front of them. Thus,

$$m_4 = 5 + 8 + 8 = (S_{0,3} + S_{1,3}) + m_3 + m_3. \quad (11)$$

However,

$$S_{0,3} + S_{1,3} = 5 = 8 - 3 = m_3 - m_2, \quad (12)$$

and from (11) and (12) we get

$$m_4 = (m_3 - m_2) + m_3 + m_3 = 3m_3 - m_2. \quad (13)$$

The number of states of a K-station production line, m_K, is given by the sum of the states of the three subsets: $S_{0,K}$, $S_{1,K}$ and $S_{2,K}$, i.e. the following equation holds

$$m_K = m_{S_{0,K}} + m_{S_{1,K}} + m_{S_{2,K}}.$$

By repeating the above process, we can easily see that the following relationships,

$$m_{S_{0,K}} = m_{S_{0,K-1}} + m_{S_{1,K-1}} = m_{K-1} - m_{K-2}, \quad (14)$$

$$m_{S_{1,K}} = m_{S_{2,K}} = m_{K-1}, \quad (15)$$

$$m_K = 3m_{K-1} - m_{K-2}, \quad (16)$$

hold for all $K \geqslant 4$ with $m_2 = 3$ and $m_3 = 8$.

In Table 1, the number m_K of the states of the subnetwork of reliable series production lines, without intermediate buffers, is given for $K = 2, 3, \ldots, 13$. By applying (16) one may obtain m_K for any K.

Table 1
Number of states of the subnetwork of reliable series production lines

No. of stations K	No. of states m_K
2	3
3	8
4	21
5	55
6	144
7	377
8	987
9	2,584
10	6,765
11	17,711
12	46,368
13	121,393

3.2. The Two- and Three-Station Production Lines

Consider matrices A_0, A_1, A_{01} and A_2 for $K = 2$ and $K = 3$. For $K = 2$, these are

$$A_0 = \begin{vmatrix} \lambda & 0 & 0 \\ \mu_2\theta & \lambda & 0 \\ 0 & \mu_2\theta & \lambda \end{vmatrix}, \tag{17}$$

$$A_1 = \begin{vmatrix} -\lambda-\mu_1 & 0 & 0 \\ \mu_2\theta' & -\lambda-\mu_1-\mu_2 & 0 \\ 0 & \mu_2\theta' & -\lambda-\mu_2 \end{vmatrix}, \tag{18}$$

$$A_{01} = \begin{vmatrix} -\lambda & 0 & 0 \\ \mu_2\theta' & -\lambda-\mu_2 & 0 \\ 0 & \mu_2\theta' & -\lambda-\mu_2 \end{vmatrix}, \tag{19}$$

$$A_2 = \begin{vmatrix} 0 & \mu_1 & 0 \\ 0 & 0 & \mu_1 \\ 0 & 0 & 0 \end{vmatrix}, \tag{20}$$

where $\theta' = 1 - \theta$.

The conservative matrix $A = A_0 + A_1 + A_2$ becomes

$$A = \begin{vmatrix} -\mu_1 & \mu_1 & 0 \\ \mu_2 & -\mu_1-\mu_2 & \mu_1 \\ 0 & \mu_2 & -\mu_2 \end{vmatrix}. \tag{21}$$

Matrix A does not contain any λ's and the sum of each row is zero. For $K = 3$, let us examine matrix A_1 first. This is a $(m_K \times m_K) = (8 \times 8)$ matrix and it has the following form

$$A_1 = \begin{vmatrix} A_{11} & 0 & 0 \\ A_{14} & A_{12} & 0 \\ 0 & A_{15} & A_{13} \end{vmatrix}, \tag{22}$$

where A_{1i}, $i = 1, \ldots, 5$, are sub-matrices which are described below.

A_{11} is a $(m_{S_{0,K}} \times m_{S_{0,K}}) = (2 \times 2)$, or by using (14) it is a $(m_{S_{0,K-1}} + m_{S_{1,K-1}})$ square matrix, according to the way the subset, $S_{0,K}$, of the states for $K = 3$, are created from the states for $K = 2$.

(1) Sub-matrix A_{11} is given by

$$A_{11} = \begin{vmatrix} -\lambda-\mu_1 & 0 \\ \mu_3\theta' & -\lambda-\mu_1-\mu_3 \end{vmatrix}. \tag{23}$$

At this point, we observe that matrix A_{11} has the following form

$$A_{11} = \begin{vmatrix} (A_{11})_\alpha & 0 \\ (A_{11})_\beta & (A_{11})_\gamma \end{vmatrix} \tag{24}$$

and in order to create A_{11} we need:

(i) $(A_{11})_\alpha$ which is given by

$$((A_{11})_\alpha)_{K=3} = (A_{11})_{K=2}; \tag{25}$$

(ii) $(A_{11})_\beta$ which is given by

$$((A_{11})_\beta)_{K=3} = (A_{14})_{K=2|\mu_2=\mu_3}, \tag{26}$$

(iii) $(A_{11})_\gamma$ which is given by

$$((A_{11})_\gamma)_{K=3} = (A_{12})_{K=2|\mu_2=\mu_3}, \tag{27}$$

where $(A_{ij})_{K=2|\mu_2=\mu_3}$ indicates that μ_2 *is substituted* by μ_3 in $(A_{ij})_{K=2}$.

(2) Sub-matrix A_{12} is a $(m_{S_{1,K}} \times m_{S_{1,K}}) = (3 \times 3) = (m_{K-1} \times m_{K-1})$ matrix, which is given by

$$A_{12} = \begin{vmatrix} -\lambda-\mu_1-\mu_2 & 0 & 0 \\ \mu_3\theta' & -\lambda-\mu_1-\mu_2-\mu_3 & \mu_2 \\ 0 & 0 & -\lambda-\mu_1-\mu_3 \end{vmatrix}. \tag{28}$$

Again, we observe that matrix A_{12} has the following form

$$A_{12} = \begin{vmatrix} & 0 & \\ (A_{12})_\alpha & & (A_{12})_\beta \\ 0 & & (A_{12})_\gamma \end{vmatrix}, \tag{29}$$

where,

(i) $(A_{12})_\alpha$ is a $(m_{S_{0,K}} \times m_{S_{0,K}}) = (2 \times 2)$ matrix and is equal to $(A_{11})_K$ *except that we subtract* μ_2 *from the diagonal elements*, i.e.

$$(A_{12})_\alpha = \begin{vmatrix} -\lambda-\mu_1-\mu_2 & 0 \\ \mu_3\theta' & -\lambda-\mu_1-\mu_2-\mu_3 \end{vmatrix}. \tag{30}$$

(ii) $(A_{12})_\beta$ is a $((m_{S_{1,K}} - m_{S_{0,K}}) \times (m_{S_{1,K}} - m_{S_{0,K}})) = ((3-2) \times (3-2)) = (1 \times 1) = (m_{S_{1,K-1}} \times m_{S_{1,K-1}})$ diagonal matrix with all diagonal elements equal to μ_2, and is placed beside $(A_{12})_\alpha$, with the left-bottom corner of $(A_{12})_\beta$ positioned beside the right-bottom corner of $(A_{12})_\alpha$. $(A_{12})_\beta$ describes the transitions in the subnetwork, where "machine-1 (second in the system) finishes the processing on its job and becomes blocked, since machine-2 (third in the system) is occupied". In

our case, for $K = 3$, this sub-matrix is a (1×1) matrix, i.e. $(A_{12})_\beta = \mu_2$.

(iii) $(A_{12})_\gamma$ is a $((m_{S_{1,K}} - m_{S_{0,K}}) \times (m_{S_{1,K}} - m_{S_{0,K}})) = ((3-2) \times (3-2)) = (1 \times 1) = (m_{S_{1,K-1}} \times m_{S_{1,K-1}})$ matrix which is equal to $(A_{12})_{K=2}$ after the substitution of μ_2 by μ_3, and which in turn, constitutes the right-bottom part of the sub-matrix $(A_{11})_{K=3} = (A_{11})_\gamma = -\lambda - \mu_1 - \mu_3$. We observe that there is no μ_2 in its diagonal elements, since these elements correspond to internal transitions (within the subnetwork), where the first station of the subnetwork (second in the line) is blocked (i.e. it has already finished the service on the existing job, which cannot move on to the next station, since that one is occupied). $(A_{12})_\gamma$ is placed directly beneath $(A_{12})_\beta$.

(3) Sub-matrix A_{13} is a $(m_{S_{2,K}} \times m_{S_{2,K}}) = (m_{S_{1,K}} \times m_{S_{1,K}}) = (3 \times 3)$ matrix, which is equal to A_{12} *except that there is no* μ_1 *in its diagonal elements*, since these correspond to those internal transitions (within the subnetwork), where the first station of the system is blocked (i.e. it has already finished processing its job, while the first station of the subnetwork (second in the line) is either busy or blocked). A_{13} is given by

$$A_{13} = \begin{vmatrix} -\lambda - \mu_2 & 0 & 0 \\ \mu_3\theta' & -\lambda - \mu_2 - \mu_3 & \mu_2 \\ 0 & 0 & -\lambda - \mu_3 \end{vmatrix}. \tag{31}$$

(4) Sub-matrix A_{14} is a $(m_{S_{1,K}} \times m_{S_{0,K}}) = (3 \times 2)$ matrix, which is placed directly beneath A_{11} and is described below:

$$A_{14} = \begin{vmatrix} 0 & \mu_2 \\ 0 & 0 \\ 0 & \mu_3\theta' \end{vmatrix}. \tag{32}$$

We observe that matrix A_{14} consists of a group of sub-matrices $(A_{14})_*$, where $*$ takes the values: α, β, γ, δ, ϵ, ζ, and η, and A_{14} can be written as follows:

$$A_{14} = \begin{vmatrix} (A_{14})_\alpha & (A_{14})_\beta & (A_{14})_\gamma \\ & (A_{14})_\delta & \\ (A_{14})_\epsilon & (A_{14})_\zeta & (A_{14})_\eta \end{vmatrix}, \tag{33}$$

where

(i) $(A_{14})_\alpha$ is a $(m_{S_{0,K-1}} \times m_{S_{0,K-1}}) = (1 \times 1)$ (for $K = 3$) matrix consisting of zero(s).

(ii) $(A_{14})_\beta$ is a $(m_{S_{0,K-1}} \times m_{S_{0,K-1}}) = (1 \times 1)$ (for $K = 3$) one-element diagonal matrix consisting of μ_2.

(iii) $(A_{14})_\gamma$ is a $(m_{S_{0,K-1}} \times (m_{S_{0,K}} - 2m_{S_{0,K-1}})) = (1 \times (2 - 2 \times 1)) = (1 \times 0)$ (for $K = 3$) matrix consisting of zero(s). For $K = 3$, $(A_{14})_\gamma$ is a degenerate matrix and this makes it difficult to understand the whole structure of the matrix A_{14}. However for $K \geq 4$, the structure becomes clear.

Sub-matrices $(A_{14})_\alpha$, $(A_{14})_\beta$ and $(A_{14})_\gamma$ constitute the first "zone" of rows of matrix A_{14}. The second "zone" of its rows contains the sub-matrix $(A_{14})_\delta$.

(iv) $(A_{14})_\delta$ is a $(m_{S_{1,K-1}} \times m_{S_{0,K}}) = (m_{S_{1,K-1}} \times (m_{S_{0,K-1}} + m_{S_{1,K-1}})) = (1 \times (1+1)) = (1 \times 2)$ matrix consisting of zeros, for $K = 3$.

(v) $(A_{14})_\epsilon$ is a $(m_{S_{2,K-1}} \times m_{S_{0,K-1}}) = (1 \times 1)$ (for $K = 3$) matrix consisting of zero(s).

(vi) $(A_{14})_\zeta$ is a $(m_{S_{2,K-1}} \times m_{S_{0,K-1}}) = (1 \times 1)$ (for $K = 3$) matrix, which is equal to the sub-matrix $(A_{11})_\beta$ of A_{11}. Finally,

(vii) $(A_{14})_\eta$ is a $(m_{S_{2,K-1}} \times (m_{S_{0,K}} - 2m_{S_{0,K-1}})) = (1 \times (2 - 2 \times 1)) = (1 \times 0)$ (for $K = 3$) matrix consisting of zeros. For $K = 3$ $(A_{14})_\eta$ is again a degenerate matrix. It is suggested to the interested reader to develop this sub-matrix for $K = 4$ so as to have a better understanding of it. In the appendix, all the sub-matrices are explicitly described, for $K = 4$.

Sub-matrices $(A_{14})_\epsilon$, $(A_{14})_\zeta$ and $(A_{14})_\eta$ constitute the third (last) "zone" of rows of (sub)matrix A_{14}. From the description of the sub-matrix A_{14} presented above, we have for $K = 3$:

$(A_{14})_\alpha = 0$,

$(A_{14})_\beta = \mu_2$,

$(A_{14})_\gamma$ does not exist (degenerate matrix),

$(A_{14})_\delta = [0,0]$,

$(A_{14})_\epsilon = 0$,

$(A_{14})_\zeta = \mu_3\theta'$, and

$(A_{14})_\eta$ does not exist (degenerate matrix).

(5) Sub-matrix A_{15} is a $(m_{S_{2,K}} \times m_{S_{1,K}}) = (3 \times 3)$ (for $K = 3$) matrix, which is placed directly beneath A_{12}. It is a square matrix, since from (15) $m_{S_{1,K}} = m_{S_{2,K}} = m_{K-1}$. It consists of two sub-matrices, having the following form:

$$A_{15} = |(A_{15})_\alpha \quad (A_{15})_\beta|, \tag{34}$$

where

(i) $(A_{15})_\alpha$ is a $(m_{S_{2,K}} \times m_{S_{0,K}}) = (3 \times 2)$ matrix, which is *exactly* the same as matrix A_{14}, generated above (this is generally true).

(ii) $(A_{15})_\beta$ is a $(m_{S_{2,K}} \times (m_{S_{1,K}} - m_{S_{0,K}})) = (3 \times (3-2)) = (3 \times 1)$ (for $K = 3$) matrix consisting of zeros and it is placed beside $(A_{15})_\alpha$.

Sub-matrix A_{15} is given below

$$A_{15} = \begin{vmatrix} 0 & \mu_2 & 0 \\ 0 & 0 & 0 \\ 0 & \mu_3\theta' & 0 \end{vmatrix}. \tag{35}$$

Generalisation. To proceed to the development of an algorithm to generate the transition matrices of a multistation series production line, it may be concluded from the discription of the various sub-matrices of A_1, given above, that in order to create matrix $(A_1)_K$, only sub-matrix $(A_{11})_K$ is required to be determined. To obtain $(A_{11})_K$ we use the sub-matrices $(A_{11})_{K-1}$, $(A_{14})_{K-1}$ and $(A_{12})_{K-1}$, where *we substitute* μ_{i-1} *by* μ_i *for* $i = K, K-1, \ldots, 3$ (i.e. backwards). The remaining sub-matrices are generated from $(A_{11})_K$ or are easily created within the algorithm (i.e. we refer to sub-matrices $(A_{14})_\beta$ and $(A_{12})_\beta$ which are diagonal matrices, with all diagonal elements equal to μ_2).

3.3. Steps of the Algorithm

The steps of the algorithm are described below.

To determine $(A_1)_K$:

Step 1. Write down and store into the appropriate cells, sub-matrix $(A_{11})_K$ by using the sub-matrices:
(i) $(A_{11})_\alpha = (A_{11})_{K-1}$,
(ii) $(A_{11})_\beta = (A_{14})_{K-1}$ and
(iii) $(A_{11})_\gamma = (A_{12})_{K-1}$.
Initial values:
For $(A_{11})_{K=2}$: $A_1(1, 1) = -\lambda - \mu_1$, for $(A_{14})_{K=2}$: $A_1(2, 1) = \mu_2\theta'$, for $(A_{12})_{K=2}$: $A_1(2, 2) = -\lambda - \mu_1 - \mu_2$.
Next, make the following $K-2$ substitutions:
do $i = K$, (step $= -1$), 3
  $\mu_{i-1} = \mu_i$
end do.

Step 2. (a) Create sub-matrix $(A_{12})_\alpha$ by subtracting μ_2 from the diagonal elements of A_{11} and store it into the appropriate cells, i.e.
do $i = 1$, $m_{S_{0,K}}$
do $j = 1$, $m_{S_{0,K}}$
  if (i.eq.j)
    then $(A_{12})_\alpha(i, i) = A_{11}(i, i) - \mu_2$
    else $(A_{12})_\alpha(i, j) = A_{11}(i, j)$
  end if
end do
end do.
(b) Create the diagonal sub-matrix $(A_{12})_\beta$ consisting of μ_2, i.e.
do $i = 1$, $m_{S_{1,K}} - m_{S_{0,K}}$
  $(A_{12})_\beta(i, i) = \mu_2$
end do
and store it into the appropriate cells.
(c) Create the sub-matrix $(A_{12})_\gamma$ which equals $(A_{11})_\gamma$, i.e.
do $i = 1$, $m_{S_{1,K}} - m_{S_{0,K}}$
do $j = 1$, $m_{S_{1,K}} - m_{S_{0,K}}$
  $(A_{12})_\gamma(i, j) = (A_{11})_\gamma(i, j)$
end do
end do
and store it into the appropriate cells.

Step 3. Create sub-matrix $(A_{13})_K$ by adding μ_1 to the diagonal elements of $(A_{12})_K$ and store it into the appropriate cells. i.e.
do $i = 1$, $m_{S_{2,K}}$
do $j = 1$, $m_{S_{2,K}}$
  if (i.eq.j)
    then $(A_{13})_K(i, i) = (A_{12})_K(i, i) + \mu_1$
    else $(A_{13})_K(i, j) = (A_{12})_K(i, j)$
  end if
end do
end do.

Step 4. Create the sub-matrix $(A_{14})_K$, i.e. create only sub-matrices $(A_{14})_\beta$ and $(A_{14})_\zeta$.
(a) Creation of the *diagonal* sub-matrix $(A_{14})_\beta$.
do $i = 1$, $m_{S_{0,K-1}}$
  $(A_{14})_\beta(i, i) = \mu_2$
end do
and store it into the appropriate cells.
(b) Creation of the sub-matrix $(A_{14})_\zeta$.
do $i = 1$, $m_{S_{2,K-1}}$
do $j = 1$, $m_{S_{0,K-1}}$
  $(A_{14})_\zeta(i, j) = (A_{11})_\beta(i, j)$
end do
end do
and store it into the appropriate cells.

Step 5. Create sub-matrix $(A_{15})_K$, i.e. only the sub-matrix $(A_{15})_\alpha$

do $i = 1, m_{S_{2,K}}$
do $j = 1, m_{S_{0,K}}$
$(A_{15})_\alpha(i, j) = (A_{14})_K(i, j)$
end do
end do.

To determine $(A_0)_K$

$(A_0)_K$ is a diagonal matrix with all diagonal elements equal to λ, i.e. a simple procedure (do-loop) is required:

do $i = 1, m_K$
$(A_0)_K(i, i) = \lambda$
end do.

Feedback

If we allow a feedback probability θ, for a job to return instantaneously from the last machine to the first queue, then we store the element with value $\mu_K\theta$, in those cells that contained $\mu_K\theta'$ in matrix $(A_1)_K$.

To determine $(A_{01})_K$

Matrix $(A_{01})_K$ is exactly the same as $(A_1)_K$ except that *there is no* μ_1 in the diagonal elements, since this matrix describes the transitions in the subnetwork, which simultaneously produce neither inputs to, nor outputs from the first queue, assuming that the first queue is *empty*. But μ_1 is contained in the first $m_{S_{0,K}} + m_{S_{1,K}}$ diagonal elements of $(A_1)_K$. Thus an algorithm may be developed to determine $(A_{01})_K$ by first defining a matrix $(A_{1C})_K$

do $i = 1, m_K$
do $j = 1, m_K$
$(A_{1C})_K(i, j) = 0$
end do
end do;
do $i = 1, m_{S_{0,K}} + m_{S_{1,K}}$
$(A_{1C})_K(i, i) = \mu_1$
end do.

Then, matrix $(A_{01})_K$ is obtained by adding $(A_{1C})_K$ to $(A_1)_K$, i.e.

do $i = 1, m_K$
do $j = 1, m_K$
$(A_{01})_K(i, j) = (A_1)_K(i, j) + (A_{1C})_K(i, j)$
end do
end do.

To determine $(A_2)_K$

Matrix $(A_2)_K$ is a very sparse matrix consisting of two sub-matrices $(A_2)_\alpha$ and $(A_2)_\beta$.

(i) $(A_2)_\alpha$ is a $(m_{S_{0,K}} \times m_{S_{0,A}})$ diagonal matrix with all diagonal elements equal to μ_1 and is placed beside $(A_{11})_K$. It is generated as follows

do $i = 1, m_{S_{0,A}}$
$(A_2)_\alpha(i, i) = \mu_1$
end do;

and the appropriate storage of $(A_2)_\alpha$ into matrix $(A_2)_K$ is:

do $i = 1, m_{S_{0,K}}$
$(A_2)_K(i, i + m_{S_{0,K}}) = \mu_1$
end do.

(ii) $(A_2)_\beta$ is a $(m_{S_{1,K}} \times m_{S_{2,K}})$ diagonal matrix with all diagonal elements again equal to μ_1 and it is placed beside $(A_{12})_K$. It is generated as follows

do $i = 1, m_{S_{1,A}}$
$(A_2)_\beta(i, i) = \mu_1$
end do;

and the appropriate storage of $(A_2)_\beta$ into matrix $(A_2)_K$ is obtained by the following do-loop:

do $i = 1, m_{S_{1,K}}$
$(A_2)_K(i + m_{S_{0,K}}, i + m_{S_{0,A}} + m_{S_{1,K}}) = \mu_1$
end do.

4. Conclusions and Recommendations

By applying the proposed algorithm, we have been able to create the coefficient matrices, i.e. the stable matrices A (see (9)), for K up to 11, where $m_{11} = 17{,}711$. These matrices are very sparse, i.e. the number of non-zero elements is less than 10%. There is no problem in continuing beyond $K = 11$, although the number of states increases tremendously. In the sequence, we used the NAG-Subroutine F04AXF, for solving the large sparse linear systems (8) and we obtained the throughput rate, R_K, very quickly. For the *balanced* case where the mean service times of all the stations are identical, comparing the results with the exact ones obtained from the Gauss Elimination Method and the Crout Decomposition Method (see NAG-Subroutine F04ATF), we ob-

served that for $K = 7$ with $m_7 = 377$ there was no difference in the first five decimal points, $R_7 = 0.45318$, for the balanced case (where the mean service times for all the stations are identical), while for $K = 8$ with $m_8 = 987$ there was only a negligible difference in the fifth decimal point; we obtained $R_8 = 0.44306$ by using F04AXF and $R_8 = 0.44307$ by applying the Gauss Elimination Method, which took much more time.

The algorithm described, could be easily extended to include intermediate buffers of the same size (i.e. $C_1 = \cdots = C_{K-1}$). The structure of the sub-blocks (matrices) would be similar. Asymmetry would be encountered if the buffers were of unequal capacity. These two cases are currently being investigated.

The model could also be extended to the case where service times follow a phase type distribution. The procedure would be similar. The only difference would be the dramatic increase in the number of states. In all these cases, packages for solving large sparse linear systems are used for storing only the non-zero elements $(i, j, a_{i,j})$, $a_{i,j} \neq 0$, of the transition matrices due to their excessive sparseness.

References

[1] B. Avi-Itzhak and M. Yadin, "A sequence of two servers with no intermediate queue", *Manage. Sci.*, Vol. 11, 1965, pp. 553–564.

[2] J.A. Buzacott and D.D. Yao, "Flexible Manufacturing Systems: a review of analytical models", *Manage. Sci.*, Vol. 32, No. 7, Vol. 1986, pp. 890–905.

[3] R.V. Evans, "Geometric distribution in some two dimensional queueing systems", *Oper. Res.*, Vol. 15, 1967, pp. 830–846.

[4] F.S. Hillier and R.W. Boling, "Finite queues in series with exponential or Erlang service times: a numerical approach", *Oper. Res.*, Vol. 15, 1967, pp. 286–303.

[5] G.C. Hunt, "Sequential arrays of waiting lines", *Oper. Res.*, Vol. 4, 1956, pp. 674–683.

[6] A.G. Konheim and M. Reiser, "A queueing model with finite waiting room and blocking", *J. Assoc. Comput. Mach.*, Vol. 23, 1976, pp. 328–341.

[7] G. Latouché and M.F. Neuts, "Efficient algorithmic solutions to exponential tandem queues with blocking", *SIAM J. Alg. Disc. Meth.*, Vol. 1, No. 1, 1980, pp. 93–106.

[8] M.F. Neuts, *Matrix-Geometric Solution in Stochastic Models*, The John Hopkins University Press, Baltimore, MD, 1981.

[9] H.T. Papadopoulos and M.E.J. O'Kelly, "Exact analysis of production lines", to be published.

[10] H.G. Perros, "Queueing networks with blocking: a bibliography", *Perform. Eval. Rev., ACM Sigmetrics*, Vol. 12, 1984, pp. 8–12.

[11] H.G. Perros and T. Altiok, "Approximate analysis of open networks of queues with blocking: tandem configurations", *IEEE Trans. Software Eng.*, Vol. 12, No. 3, 1986, pp. 450–461.

[12] V. Wallace, "The solution of quasi birth and death processes arising from multiple access computer systems", Ph.D. Thesis, Tech. Rep. No. 07742-6-T, Systems Engineering Lab, University of Michigan, Ann Arbor, MI, 1969.

[13] V. Wallace, "Algebraic techniques for numerical solution of queueing networks", *Proc. of a Conference at Western Michigan University*, Kalamazoo, MI, May 10–12, 1973.

Appendix—Four-Station Series Production Line

Here, the explicit derivation of the involved matrices and sub-matrices is presented for a four-station series production line, i.e. for $K = 4$, by applying the described algorithm.

A.1. Determination of $(A_1)_{K=4}$

$(A_1)_{K=4}$ is a $(m_4 \times m_4) = (21 \times 21)$ matrix and it has the following form

$$(A_1)_{K=4} = \begin{vmatrix} (A_{11})_{K=4} & 0 & 0 \\ (A_{14})_{K=4} & (A_{12})_{K=4} & 0 \\ 0 & (A_{15})_{K=4} & (A_{13})_{K=4} \end{vmatrix}, \tag{A.1}$$

where $(A_{1i})_{K=4}$, $i = 1, \ldots, 5$, are sub-matrices which are described below.

$(A_{11})_{K=4}$ is a $(m_{S_{0,4}} \times m_{S_{0,4}}) = (5 \times 5)$, or by using (14) it is a $(m_{S_{0,3}} + m_{S_{1,3}})$ square matrix, according to the way the subset $S_{0,4}$ of the states for $K = 4$ are created from the states for $K = 3$.

(1) Sub-matrix $(A_{11})_{K=4}$ is given by

$$(A_{11})_{K=4} = \begin{vmatrix} -\lambda-\mu_1 & 0 & 0 & 0 & 0 \\ \mu_4\theta' & -\lambda-\mu_1-\mu_4 & 0 & 0 & 0 \\ 0 & \mu_3 & -\lambda-\mu_1-\mu_3 & 0 & 0 \\ 0 & 0 & \mu_4\theta' & -\lambda-\mu_1-\mu_3-\mu_4 & \mu_3 \\ 0 & \mu_4\theta' & 0 & 0 & -\lambda-\mu_1-\mu_4 \end{vmatrix}. \quad (A.2)$$

Matrix $(A_{11})_{K=4}$ has the form (24), i.e.

$$(A_{11})_{K=4} = \begin{vmatrix} ((A_{11})_\alpha)_{K=4} & 0 \\ ((A_{11})_\beta)_{K=4} & ((A_{11})_\gamma)_{K=4} \end{vmatrix} \quad (A.3)$$

and in order to create $(A_{11})_{K=4}$ we need:
(i) $((A_{11})_\alpha)_{K=4}$ which is given by

$$((A_{11})_\alpha)_{K=4} = (A_{11})_{K=3|\mu_3=\mu_4}; \quad (A.4)$$

(ii) $((A_{11})_\beta)_{K=4}$ which is given by

$$((A_{11})_\beta)_{K=4} = (A_{14})_{K=3|\mu_3=\mu_4,\ \mu_2=\mu_3}; \quad (A.5)$$

and (iii) $((A_{11})_\gamma)_{K=4}$ which is given by

$$((A_{11})_\gamma)_{K=4} = (A_{12})_{K=3|\mu_3=\mu_4,\ \mu_2=\mu_3}, \quad (A.6)$$

where $(A_{ij})_{K=3|\mu_3=\mu_4,\ \mu_2=\mu_3}$ indicates that μ_3 *is substituted* by μ_4 first and then μ_2 *is substituted* by μ_3 in $(A_{ij})_{K=3}$.

(2) Sub-matrix $(A_{12})_{K=4}$ is a $(m_{S_{1,4}} \times m_{S_{1,4}}) = (8 \times 8) = (m_3 \times m_3)$ matrix, which is given by the sum of two matrices: B_1 and B_2. B_1 equals $(A_{13})_{K=4}$ as we will see below, while B_2 is a diagonal matrix with all diagonal elements equal to $-\mu_1$, i.e.

$$(A_{12})_{K=4} = B_1 + B_2, \quad (A.7)$$

where

$$B_1 = \begin{vmatrix} -\lambda-\mu_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \mu_4\theta' & -\lambda-\mu_2-\mu_4 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \mu_3 & -\lambda-\mu_2-\mu_3 & 0 & 0 & \mu_2 & 0 & 0 \\ 0 & 0 & \mu_4\theta' & -\lambda-\sum_{i=2}^{i=4}\mu_i & \mu_3 & 0 & \mu_2 & 0 \\ 0 & \mu_4\theta' & 0 & 0 & -\lambda-\mu_2-\mu_4 & 0 & 0 & \mu_2 \\ 0 & 0 & 0 & 0 & 0 & -\lambda-\mu_3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \mu_4\theta' & -\lambda-\mu_3-\mu_4 & \mu_3 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\lambda-\mu_4 \end{vmatrix}$$

and

$$B_2 = \begin{vmatrix} -\mu_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -\mu_1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\mu_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\mu_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\mu_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -\mu_1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\mu_1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\mu_1 \end{vmatrix}.$$

Again, we observe that matrix $(A_{12})_{K=4}$ has the following form (see (29))

$$(A_{12})_{K=4} = \begin{vmatrix} & 0 & \\ ((A_{12})_\alpha)_{K=4} & & ((A_{12})_\beta)_{K=4} \\ 0 & & ((A_{12})_\gamma)_{K=4} \end{vmatrix}. \tag{A.8}$$

where,

(i) $((A_{12})_\alpha)_{K=4}$ is a $(m_{S_{0,4}} \times m_{S_{0,4}}) = (5 \times 5)$ matrix and is equal to $(A_{11})_{K=4}$ *except that we subtract* μ_2 *from the diagonal elements*, i.e.

$$((A_{12})_\alpha)_{K=4}$$

$$= \begin{vmatrix} -\lambda-\mu_1-\mu_2 & 0 & 0 & 0 & 0 \\ \mu_4\theta' & -\lambda-\mu_1-\mu_2-\mu_4 & 0 & 0 & 0 \\ 0 & \mu_3 & -\lambda-\sum_{i=1}^{i=3}\mu_i & 0 & 0 \\ 0 & 0 & \mu_4\theta' & -\lambda-\sum_{i=1}^{i=4}\mu_i & \mu_3 \\ 0 & \mu_4\theta' & 0 & 0 & -\lambda-\mu_1-\mu_2-\mu_4 \end{vmatrix}. \tag{A.9}$$

(ii) $((A_{12})_\beta)_{K=4}$ is a $((m_{S_{1,4}} - m_{S_{0,4}}) \times (m_{S_{1,4}} - m_{S_{0,4}})) = ((8-5) \times (8-5)) = (3 \times 3) = (m_{S_{1,3}} \times m_{S_{1,3}})$ diagonal matrix with all diagonal elements *equal* to μ_2, and is placed beside $((A_{12})_\alpha)_{K=4}$, with the left-bottom corner of $((A_{12})_\beta)_{K=4}$ positioned beside the right-bottom corner of $((A_{12})_\alpha)_{K=4}$. $((A_{12})_\beta)_{K=4}$ describes the transitions in the subnetwork, where "machine-1 (second in the system) finishes the processing on its job and becomes blocked, since machine-2 (third in the system) is occupied". In our case, for $K = 4$, this sub-matrix is a (3×3) matrix, i.e.,

$$((A_{12})_\beta)_{K=4} = \begin{vmatrix} \mu_2 & 0 & 0 \\ 0 & \mu_2 & 0 \\ 0 & 0 & \mu_2 \end{vmatrix}.$$

(iii) $((A_{12})_\gamma)_{K=4}$ is a $((m_{S_{1,4}} - m_{S_{0,4}}) \times (m_{S_{1,4}} - m_{S_{0,4}})) = ((8-5) \times (8-5)) = (3 \times 3) = (m_{S_{1,3}} \times m_{S_{1,3}})$ matrix which is identical to the right-bottom part of the sub-matrix $(A_{11})_{K=4}$, viz.$((A_{11})_\gamma)_{K=4}$, i.e.

$$((A_{12})_\gamma)_{K=4} = \begin{vmatrix} -\lambda-\mu_1-\mu_3 & 0 & 0 \\ \mu_4\theta' & -\lambda-\mu_1-\mu_3-\mu_4 & \mu_3 \\ 0 & 0 & -\lambda-\mu_1-\mu_4 \end{vmatrix}.$$

We observe that there is no μ_2 in its diagonal elements, since these elements correspond to internal transitions (within the subnetwork), where the first station of the subnetwork (second in the line) is blocked (i.e. it has already finished the service on the existing job, which cannot move on to the next station, since that one is occupied). $((A_{12})_\gamma)_{K=4}$ is placed directly beneath $((A_{12})_\beta)_{K=4}$.

(3) Sub-matrix $(A_{13})_{K=4}$ is a $(m_{S_{2,4}} \times m_{S_{2,4}}) = (m_{S_{1,4}} \times m_{S_{1,4}}) = (8 \times 8)$ matrix, which is equal to B_1 defined above, i.e. it is equal to $(A_{12})_{K=4}$ *except that there is no* μ_1 *in its diagonal elements*, since these correspond to those internal transitions (within the subnetwork), where the first station of the system is blocked (i.e. it has already finished processing its job, while the first station of the subnetwork (second in the line) is either busy or blocked), i.e. it holds

$$(A_{13})_{K=4} = B_1. \tag{A.10}$$

(4) Sub-matrix $(A_{14})_{K-4}$ is a $(m_{S_{1,4}} \times m_{S_{0,4}}) = (8 \times 5)$ matrix, which is placed directly beneath $(A_{11})_{K-4}$ and is described below:

$$(A_{14})_{K-4} = \begin{vmatrix} 0 & 0 & \mu_2 & 0 & 0 \\ 0 & 0 & 0 & \mu_2 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mu_3 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mu_4\theta' & 0 \end{vmatrix}. \tag{A.11}$$

We observe that matrix $(A_{14})_{K-4}$ consists of a group of sub-matrices $((A_{14})_*)_{K-4}$, where $*$ takes the values: α, β, γ, δ, ϵ, ζ, and η, and $(A_{14})_{K-4}$ can be written as in formula (33): where, here,

(i) $((A_{14})_\alpha)_{K-4}$ is a $(m_{S_{0,3}} \times m_{S_{0,3}}) = (2 \times 2)$ matrix consisting of zeros,i.e.

$$((A_{14})_\alpha)_{K-4} = \begin{vmatrix} 0 & 0 \\ 0 & 0 \end{vmatrix}.$$

(ii) $((A_{14})_\beta)_{K-4}$ is a $(m_{S_{0,3}} \times m_{S_{0,3}}) = (2 \times 2)$ diagonal matrix with all diagonal elements equal to μ_2, i.e.

$$((A_{14})_\beta)_{K-4} = \begin{vmatrix} \mu_2 & 0 \\ 0 & \mu_2 \end{vmatrix}.$$

(iii) $((A_{14})_\gamma)_{K-4}$ is a $(m_{S_{0,3}} \times (m_{S_{0,4}} - 2m_{S_{0,3}})) = (2 \times (5 - 2 \times 2)) = (2 \times 1)$ matrix consisting of zeros, i.e.

$$((A_{14})_\gamma)_{K-4} = \begin{vmatrix} 0 \\ 0 \end{vmatrix}.$$

(iv) $((A_{14})_\delta)_{K-4}$ is a $(m_{S_{1,3}} \times m_{S_{0,4}}) = (m_{S_{1,3}} \times (m_{S_{0,3}} + m_{S_{1,3}})) = (3 \times (2 + 3)) = (3 \times 5)$ matrix consisting of zeros, i.e.

$$((A_{14})_\delta)_{K-4} = \begin{vmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{vmatrix}.$$

(v) $((A_{14})_\epsilon)_{K-4}$ is a $(m_{S_{2,3}} \times m_{S_{0,3}}) = (3 \times 2)$ matrix consisting of zeros, i.e.

$$((A_{14})_\epsilon)_{K-4} = \begin{vmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{vmatrix}.$$

(vi) $((A_{14})_\zeta)_{K-4}$ is a $(m_{S_{2,3}} \times m_{S_{0,3}}) = (3 \times 2)$ matrix, which is equal to the sub-matrix $((A_{11})_\beta)_{K-4}$ of $(A_{11})_{K-4}$, i.e.

$$((A_{14})_\zeta)_{K-4} = \begin{vmatrix} 0 & \mu_3 \\ 0 & 0 \\ 0 & \mu_4\theta' \end{vmatrix}.$$

Finally,

(vii) $((A_{14})_\eta)_{K-4}$ is a $(m_{S_{2,3}} \times (m_{S_{0,4}} - 2m_{S_{0,3}})) = (3 \times (5 - 2 \times 2)) = (3 \times 1)$ matrix consisting of zeros, i.e.

$$((A_{14})_\eta)_{K-4} = \begin{vmatrix} 0 \\ 0 \\ 0 \end{vmatrix}.$$

(5) Sub-matrix $(A_{15})_{K=4}$ is a $(m_{S_{2,4}} \times m_{S_{1,4}}) = (8 \times 8)$ matrix, which is placed directly beneath $(A_{12})_{K=4}$ and it is a square matrix, since from formula (15) $m_{S_{1,4}} = m_{S_{2,4}} = m_{K=3}$. It consists of two sub-matrices, having the form (34), where

(i) $((A_{15})_\alpha)_{K=4}$ is a $(m_{S_{2,4}} \times m_{S_{0,4}}) = (8 \times 5)$ matrix, which is exactly the same as matrix $(A_{14})_{K=4}$, generated above.

(ii) $((A_{15})_\beta)_{K=4}$ is a $(m_{S_{2,4}} \times (m_{S_{1,4}} - m_{S_{0,4}})) = (8 \times (8-5)) = (8 \times 3)$ matrix consisting of zeros and it is placed beside $((A_{15})_\alpha)_{K=4}$.

Sub-matrix $(A_{15})_{K=4}$ is given below

$$(A_{15})_{K=4} = \begin{vmatrix} 0 & 0 & \mu_2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mu_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mu_3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mu_4\theta' & 0 & 0 & 0 & 0 \end{vmatrix}. \tag{A.12}$$

A.2. Determination of $(A_0)_{K=4}$

$(A_0)_{K=4}$ is a diagonal matrix with all diagonal elements equal to λ, i.e. a simple do-loop is required:

do $i = 1, 21$
  $(A_0)_{K=4}(i, i) = \lambda$
end do;

Feedback. If we allow a feedback probability θ, for a job to return instantaneously, from the last machine (4th) to the first queue, then we store the element with value $\mu_4\theta$, in those cells that contained $\mu_4\theta'$ in matrix $(A_1)_{K=4}$.

A.3. Determination of $(A_{01})_{K=4}$

Matrix $(A_{01})_{K=4}$ is exactly the same as $(A_1)_{K=4}$ except that *there is no* μ_1 in the diagonal elements, since this matrix describes the transitions in the subnetwork, which simultaneously produce neither inputs to, nor outputs from the first queue, assuming that the first queue is *empty*. But μ_1 is contained in the first $m_{S_{0,4}} + m_{S_{1,4}} = 5 + 8 = 13$ diagonal elements of $(A_1)_{K=4}$. Thus an algorithm may be developed to determine $(A_{01})_{K=4}$ by first defining a matrix $(A_{1C})_{K=4}$:

do $i = 1, 21$
do $j = 1, 21$
  $(A_{1C})_{K=4}(i, j) = 0$
end do
end do;
do i = 1, 13
  $(A_{1C})_{K=4}(i, i) = \mu_1$
end do;

then, matrix $(A_{01})_{K=4}$ is obtained by adding $(A_{1C})_{K=4}$ to $(A_1)_{K=4}$, i.e.

do $i = 1, 21$
do $j = 1, 21$
$(A_{01})_{K-4}(i, j) = (A_1)_{K-4}(i, j) + (A_{1C})_{K-4}(i, j)$
end do
end do.

A.4. Determination of $(A_2)_{K-4}$

Matrix $(A_2)_{K-4}$ is a very sparse matrix consisting of two sub-matrices $((A_2)_\alpha)_{K-4}$ and $((A_2)_\beta)_{K-4}$.

(i) $((A_2)_\alpha)_{K-4}$ is a $(m_{S_{0,4}} \times m_{S_{0,4}}) = (5 \times 5)$ diagonal matrix with all diagonal elements equal to μ_1 and is placed beside $(A_{11})_{K-4}$. It is generated as follows

do $i = 1, 5$
$((A_2)_\alpha)_{K-4}(i, i) = \mu_1$
end do;

and the appropriate storage of $((A_2)_\alpha)_{K-4}$ into matrix $(A_2)_{K-4}$ is:

do $i = 1, 5$
$(A_2)_{K-4}(i, i + 5) = \mu_1$
end do.

(ii) $((A_2)_\beta)_{K-4}$ is a $(m_{S_{1,4}} \times m_{S_{2,4}}) = (8 \times 8)$ diagonal matrix with all diagonal elements equal to μ_1 and is placed beside $(A_{12})_{K-4}$. It is generated as follows

do $i = 1, 8$
$((A_2)_\beta)_{K-4}(i, i) = \mu_1$
end do;

and the appropriate storage of $((A_2)_\beta)_{K-4}$ is obtained by the following do-loop:

do $i = 1, 8$
$(A_2)_{K-4}(i + 5, i + 13) = \mu_1$
end do.