The Max-Plus Algebra Approach in Modelling of Queueing Networks

Nikolai K. Krivulin
Faculty of Mathematics and Mechanics
St.Petersburg State University
Bibliotechnaya Sq.2, Petrodvorets
St.Petersburg, 198904 Russia
E-mail: krivulin@niimm.spb.su

Model design, System dynamics, Queueing networks

Abstract

A class of queueing networks which consist of single-server fork-join nodes with infinite buffers is examined to derive a representation of the network dynamics in terms of max-plus algebra. For the networks, we present a common dynamic state equation which relates the departure epochs of customers from the network nodes in an explicit vector form determined by a state transition matrix. We show how the matrix may be calculated from the service time of customers in the general case, and give examples of matrices inherent in particular networks.

1 Introduction

We consider a class of queueing networks with single-server nodes and customers of a single class. The server at each node is supplied with an infinite buffer intended for customers waiting for service. There is, in general, no restriction on the network topology; in particular, both open and closed networks may be included in the class.

In addition to the ordinary service procedure, specific fork-join operations [Baccelli and Makowski 1989, Greenberg *et al.* 1991] may be performed in each node of the network. In fact, these operations allow customers (jobs, tasks) to be split into parts, and to be merged into one, when circulating through the network.

Furthermore, we assume that there may be nodes, each distributing customers among a group of its downstream nodes by a regular round routing mechanism. Every node operating on the round routing basis passes on the consecutive customers to distinct nodes being assigned to each new customer cyclically in a fixed order.

Both the fork-join formalism and the above regular round routing scheme seem to be useful in the description of dynamical processes in a variety of actual systems, including production processes in manufacturing, transmission of messages in communication networks, and parallel data processing in multi-processor computer systems (see, e.g., [Baccelli and Makowski 1989]).

In this paper, the networks are examined so as to represent their dynamics in terms of max-plus algebra [Cuninghame-Green 1979, Olsder 1992]. The max-plus algebra approach actually offers a quite compact and unified way of describing system dynamics, which may provide a useful basis for both analytical study and simulation of queueing systems. In particular, it has been shown in [Krivulin 1994, Krivulin 1995] that the evolution of both open and closed tandem queueing systems may be described by the linear algebraic equation

$$\boldsymbol{d}(k) = T(k) \otimes \boldsymbol{d}(k-1), \tag{1}$$

where $\boldsymbol{d}(k)$ is a vector of departure epochs from the queues, $T(k)$ is a matrix calculated from service times of customers, and $\otimes$ is an operator which determines the matrix-vector multiplication in the max-plus algebra.

The purpose of this paper is to demonstrate that the network dynamics also allows of representation in the form of dynamic state equation (1). We start with preliminary max-plus algebra definitions and related results. Furthermore, we give a general description of a queueing network model, which is then refined to describe fork-join networks. It is shown how the dynamics of the networks may be represented in the form of (1).

The obtained representation is extended to describe the dynamics of tandem queueing systems and a system with regular round routing. In fact, tandem systems may be treated as trivial fork-join networks, and thus described in the same way. To represent the system with round routing, which operates differently than the other networks under examination, we first introduce an equivalent fork-join network, and then get equation (1).

2 Algebraic Definitions and Results

We start with a brief overview of basic facts about max-plus algebra, which we will exploit in the representation of queueing network models in the subsequent sections. Further details concerning the max-plus algebra theory as well as its applications can be found in [Cuninghame-Green 1979, Cohen *et al.* 1989, Olsder 1992].

Max-plus algebra is normally defined (see, for example, [Olsder 1992]) as the system $\langle \underline{\mathbb{R}}, \oplus, \otimes \rangle$, where $\underline{\mathbb{R}} = \mathbb{R} \cup \{\varepsilon\}$ with $\varepsilon = -\infty$, and for any $x, y \in \underline{\mathbb{R}}$,

$$x \oplus y = \max(x, y), \quad x \otimes y = x + y.$$

Since the new operations $\oplus$ and $\otimes$ retain most of the properties of the ordinary addition and multiplication, including associativity, commutativity, and distributivity of multiplication over addition, one can perform usual algebraic manipulations in the max-plus algebra under the standard conventions regarding brackets and precedence of $\otimes$ over $\oplus$. However, the operation $\oplus$ is idempotent; that is, for any $x \in \underline{\mathbb{R}}$, one has $x \oplus x = x$.

There are the null and identity elements in this algebra, namely ε and 0, which satisfy the evident conditions $x \oplus \varepsilon = \varepsilon \oplus x = x$, and $x \otimes 0 = 0 \otimes x = x$, for any $x \in \underline{\mathbb{R}}$. The absorption rule involving $x \otimes \varepsilon = \varepsilon \otimes x = \varepsilon$ is also true in the algebra.

The max-plus algebra of matrices is readily introduced in the regular way. Specifically, for any $(n \times n)$-matrices $X = (x_{ij})$ and $Y = (y_{ij})$, the entries of $U = X \oplus Y$ and $V = X \otimes Y$ are calculated as

$$u_{ij} = x_{ij} \oplus y_{ij}, \quad \text{and} \quad v_{ij} = \sum_{k=1}^{n}{}_{\oplus}\, x_{ik} \otimes y_{kj},$$

where $\sum_{\oplus}$ stands for the iterated operation $\oplus$. As the null element, the matrix $\mathcal{E}$ with all entries equal to ε is taken in the algebra; the matrix $E = \mathrm{diag}(0, \ldots, 0)$ with the off-diagonal entries set to ε presents the identity.

As is customary, for any square matrix X, one can define $X^0 = E$ and $X^q = X \otimes X^{q-1} = X^{q-1} \otimes X$ for all $q = 1, 2, \ldots$. Note, however, that idempotency in this algebra leads, in particular, to the matrix identity

$$(E \oplus X)^q = E \oplus X \oplus \cdots \oplus X^q.$$

Many phenomena inherent in the max-plus algebra appear to be well explained in terms of their graph interpretations [Cuninghame-Green 1979, Cohen *et al.* 1989, Olsder 1992]. To illustrate, consider an $(n \times n)$-matrix X with entries $x_{ij} \in \underline{\mathbb{R}}$, and note that it can be treated as the adjacency matrix of an oriented graph with n nodes, provided each entry $x_{ij} \neq \varepsilon$ implies the existence of the arc (i, j) in the graph, whereas $x_{ij} = \varepsilon$ does the lack of the arc.

Let us calculate the matrix $X^2 = X \otimes X$, and denote its entries by $x_{ij}^{(2)}$. Clearly, we have $x_{ij}^{(2)} \neq \varepsilon$ if and only if there exists at least one path from node i to node j in the graph, which consists of two arcs. Moreover, for any integer $q > 0$, the matrix X^q has the entry $x_{ij}^{(q)} \neq \varepsilon$ only if there is a path with the length q from i to j.

Suppose that the graph associated with X is acyclic. It is clear that we will have $X^q = \mathcal{E}$ for all $q > p$, where p is the length of the longest path in the graph. Assume now the graph not to be acyclic, and then consider any one of its circuits. Since it is possible to construct a cyclic path of any length, which lies along the circuit, we conclude that $X^q \neq \mathcal{E}$ for all $q = 1, 2, \ldots$

Finally, we consider the implicit equation in the vector $\boldsymbol{x} = (x_1, \ldots, x_n)^T$,

$$\boldsymbol{x} = U \otimes \boldsymbol{x} \oplus \boldsymbol{v}, \tag{2}$$

where $U = (u_{ij})$ and $\boldsymbol{v} = (v_1, \ldots, v_n)^T$ are respectively given $(n \times n)$-matrix and n-vector. This equation actually plays a large role in max-plus algebra representations of dynamical systems including systems of queues [Cohen *et al.* 1989, Olsder 1992, Krivulin 1995]. The next lemma offers particular conditions for (2) to be solvable, and it shows how the solution may be calculated. One can find a detailed analysis of (2) in the general case in [Cohen *et al.* 1989].

Lemma 1 *Suppose that the entries of the matrix U and the vector $\boldsymbol{v}$ are either positive or equal to ε. Then equation* (2) *has the unique bounded solution $\boldsymbol{x}$ if and only if the graph associated with the matrix U is acyclic. Provided that the solution exists, it is given by*

$$\boldsymbol{x} = (E \oplus U)^p \otimes \boldsymbol{v},$$

where p is the length of the longest path in the graph.

The proof of the lemma may be furnished based on the above graph interpretation and idempotency of $\oplus$.

3 A General Network Description

In this section, we introduce some general notations and give related definitions, which are common for all particular networks examined below. In fact, we consider a network with n single-server nodes and customers of a single class. The network topology is described by an oriented graph $\mathcal{G} = (\mathbf{N}, \mathbf{A})$, where $\mathbf{N} = \{1, \ldots, n\}$ represents the nodes, and $\mathbf{A} = \{(i, j)\} \subset \mathbf{N} \times \mathbf{N}$ does the arcs determining the transition routes of customers.

For every node $i \in \mathbf{N}$, we introduce the set of its predecessors $\mathbf{P}(i) = \{j | (j, i) \in \mathbf{A}\}$ and the set of its successors $\mathbf{S}(i) = \{j | (i, j) \in \mathbf{A}\}$. In specific cases, there may be one of the conditions $\mathbf{P}(i) = \emptyset$ and $\mathbf{S}(i) = \emptyset$ encountered. Each node i with $\mathbf{P}(i) = \emptyset$ is assumed to represent an infinite external arrival stream of customers; provided that

$\mathbf{S}(i) = \emptyset$, it is considered as an output node intended to release customers from the network.

Each node $i \in \mathbf{N}$ includes a server and its buffer with infinite capacity, which together present a single-server queue operating under the first-come, first-served (FCFS) queueing discipline. At the initial time, the server at each node i is assumed to be free of customers, whereas in its buffer, there may be r_i, $0 \leq r_i \leq \infty$, customers waiting for service. The value $r_i = \infty$ is set for every node i with $\mathbf{P}(i) = \emptyset$, which represents an external arrival stream of customers.

For the queue at node i, we denote the kth arrival and departure epochs respectively as $a_i(k)$ and $d_i(k)$. Furthermore, the service time of the kth customer at server i is indicated by τ_{ik}. We assume that $\tau_{ik} > 0$ are given parameters for all $i = 1, \ldots, n$, and $k = 1, 2, \ldots$, while $a_i(k)$ and $d_i(k)$ are considered as unknown state variables. With the condition that the network starts operating at time zero, it is convenient to set $d_i(0) \equiv 0$, and $d_i(k) \equiv \varepsilon$ for all $k < 0$, $i = 1, \ldots, n$.

It is easy to set up an equation which relates the system state variables. In fact, the dynamics of any single-server node i with an infinite buffer, operating on the FCFS basis, is described as [Krivulin 1995]

$$d_i(k) = \tau_{ik} \otimes a_i(k) \oplus \tau_{ik} \otimes d_i(k-1). \tag{3}$$

With the system state vectors

$$\boldsymbol{a}(k) = \begin{pmatrix} a_1(k) \\ \vdots \\ a_n(k) \end{pmatrix}, \quad \boldsymbol{d}(k) = \begin{pmatrix} d_1(k) \\ \vdots \\ d_n(k) \end{pmatrix},$$

and the diagonal matrix

$$\mathcal{T}_k = \begin{pmatrix} \tau_{1k} & & \varepsilon \\ & \ddots & \\ \varepsilon & & \tau_{nk} \end{pmatrix},$$

we may rewrite equation (3) in a vector form, as

$$\boldsymbol{d}(k) = \mathcal{T}_k \otimes \boldsymbol{a}(k) \oplus \mathcal{T}_k \otimes \boldsymbol{d}(k-1). \tag{4}$$

Clearly, to represent the dynamics of a network completely, equation (4) should be supplemented with that determining the vector of arrival epochs, $\boldsymbol{a}(k)$. The latter equation may differ for distinct networks according to their operation features and topology. We will give appropriate equations for $\boldsymbol{a}(k)$ inherent in particular networks, as well as related representations of the entire network dynamics in the subsequent sections.

4 Fork-Join Queueing Networks

The purpose of this section is to derive an algebraic representation of the dynamics of fork-join networks which present a quite general class of queueing network models. Since we do not impose any limitation on the network topology, the models under study may be considered as an extension of acyclic fork-join queueing networks investigated in [Baccelli and Makowski 1989].

The distinctive feature of any fork-join network is that, in addition to the usual service procedure, special join and fork operations are performed in its nodes, respectively before and after service. The join operation is actually thought to cause each customer which comes into node i, not to enter the buffer at the server but to wait until at least one customer from every node $j \in \mathbf{P}(i)$ arrives. As soon as these customers arrive, they, taken one from each preceding node, are united to be treated as being one customer which then enters the buffer to become a new member of the queue.

The fork operation at node i is initiated every time the service of a customer is completed; it consists in giving rise to several new customers instead of the original one. As many new customers appear in node i as there are succeeding nodes included in the set $\mathbf{S}(i)$. These customers simultaneously depart the node, each being passed to separate node $j \in \mathbf{S}(i)$. We assume that the execution of fork-join operations when appropriate customers are available, as well as the transition of customers within and between nodes require no time.

As it immediately follows from the above description of the fork-join operations, the kth arrival epoch into the queue at node i is represented as (also, see [Baccelli and Makowski 1989, Greenberg *et al.* 1991])

$$a_i(k) = \begin{cases} \sum_{j \in \mathbf{P}(i)}^{\oplus} d_j(k - r_i), & \text{if } \mathbf{P}(i) \neq \emptyset, \\ \varepsilon, & \text{if } \mathbf{P}(i) = \emptyset. \end{cases} \tag{5}$$

In order to get this equation in a vector form, we first define $M = \max\{r_i \mid r_i < \infty, i = 1, \ldots, n\}$. Now we may rewrite (5) as

$$a_i(k) = \sum_{m=0}^{M}{}_{\oplus} \sum_{j=1}^{n}{}_{\oplus} g_{ji}^m \otimes d_j(k-m),$$

where the numbers g_{ij}^m are determined by the condition

$$g_{ij}^m = \begin{cases} 0, & \text{if } i \in \mathbf{P}(j) \text{ and } m = r_j, \\ \varepsilon, & \text{otherwise.} \end{cases} \tag{6}$$

Let us introduce the matrices $G_m = \left(g_{ij}^m\right)$ for each $m = 0, 1, \ldots, M$, and note that G_m actually presents an adjacency matrix of the partial graph $\mathcal{G}_m = (\mathbf{N}, \mathbf{A}_m)$, where $\mathbf{A}_m = \{(i,j) \mid i \in \mathbf{P}(j),\, r_j = m\}$. With these matrices, equation (5) may be written in the vector form

$$\boldsymbol{a}(k) = \sum_{m=0}^{M}{}_{\oplus} G_m^T \otimes \boldsymbol{d}(k-m), \tag{7}$$

where G_m^T denotes the transpose of the matrix G_m.

Furthermore, by combining equations (4) and (7), we arrive at the equation

$$\boldsymbol{d}(k) = \mathcal{T}_k \otimes G_0^T \otimes \boldsymbol{d}(k) \oplus \mathcal{T}_k \otimes \boldsymbol{d}(k-1) \oplus \mathcal{T}_k \otimes \sum_{m=1}^{M}{}_{\oplus}\, G_m^T \otimes \boldsymbol{d}(k-m). \qquad (8)$$

Clearly, it is actually an implicit equation in $\boldsymbol{d}(k)$, which has the form of (2), with $U = \mathcal{T}_k \otimes G_0^T$. Taking into account that the matrix $\mathcal{T}_k$ is diagonal, one can apply Lemma 1 to prove the following statement.

Theorem 1 *Suppose that in the fork-join network model, the graph $\mathcal{G}_0$ associated with the matrix G_0 is acyclic. Then equation (8) can be solved to produce the explicit dynamic state equation*

$$\boldsymbol{d}(k) = \sum_{m=1}^{M}{}_{\oplus}\, T_m(k) \otimes \boldsymbol{d}(k-m), \qquad (9)$$

with the state transition matrices

$$\begin{aligned} T_1(k) &= (E \oplus \mathcal{T}_k \otimes G_0^T)^p \otimes \mathcal{T}_k \otimes (E \oplus G_1^T), \\ T_m(k) &= (E \oplus \mathcal{T}_k \otimes G_0^T)^p \otimes \mathcal{T}_k \otimes G_m^T, \\ & \quad m = 2, \ldots, M, \end{aligned}$$

where p is the length of the longest path in $\mathcal{G}_0$.

Finally, with the extended state vector

$$\widehat{\boldsymbol{d}}(k) = \begin{pmatrix} \boldsymbol{d}(k) \\ \boldsymbol{d}(k-1) \\ \vdots \\ \boldsymbol{d}(k-M+1) \end{pmatrix},$$

one can bring equation (9) into the form of (1):

$$\widehat{\boldsymbol{d}}(k) = \widehat{T}(k) \otimes \widehat{\boldsymbol{d}}(k-1),$$

where the state transition matrix is defined as

$$\widehat{T}(k) = \begin{pmatrix} T_1(k) & T_2(k) & \cdots & \cdots & T_M(k) \\ E & \mathcal{E} & \cdots & \cdots & \mathcal{E} \\ & \ddots & \ddots & & \vdots \\ & & \ddots & \ddots & \vdots \\ \mathcal{E} & & & E & \mathcal{E} \end{pmatrix}.$$

To conclude this section, we present an example which shows how the dynamics of a particular fork-join network is described based on the above representation.

Example 1 *We consider a network with $n = 5$ nodes, depicted in Fig. 1. The initial numbers of customers in the network nodes are determined as follows: $r_1 = \infty$, $r_2 = r_4 = 0$, and $r_3 = r_5 = 1$.*

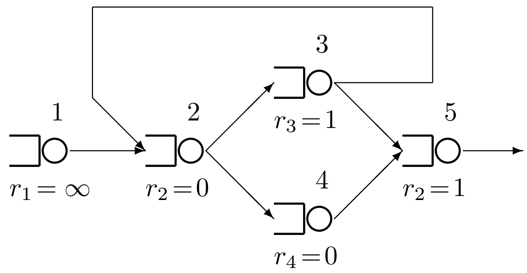

Figure 1: A fork-join queueing network.

First note that for the network, we have $M = 1$. In this case, provided that explicit representation (9) exists, it is just written in the form of (1):

$$\boldsymbol{d}(k) = T(k) \otimes \boldsymbol{d}(k-1),$$

with $T(k) = T_1(k) = (E \oplus \mathcal{T}_k \otimes G_0^T)^p \otimes \mathcal{T}_k \otimes (E \oplus G_1^T)$.

Furthermore, by applying (6), we may calculate

$$G_0 = \begin{pmatrix} \varepsilon & 0 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & 0 & \varepsilon \\ \varepsilon & 0 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{pmatrix}, \quad G_1 = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 0 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & 0 \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & 0 \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{pmatrix}.$$

Since the graph associated with the matrix G_0 is acyclic, with the length of its longest path $p = 2$, we may really describe the network dynamics through equation (1). Finally, simple algebraic manipulations give

$$T(k) = (E \oplus \mathcal{T}_k \otimes G_0^T)^2 \otimes \mathcal{T}_k \otimes (E \oplus G_1^T) = \begin{pmatrix} \tau_{1k} & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \tau_{1k}\otimes\tau_{2k} & \tau_{2k}\otimes\tau_{3k} & \tau_{2k}\otimes\tau_{3k} & \varepsilon & \varepsilon \\ \varepsilon & \tau_{3k} & \tau_{3k} & \varepsilon & \varepsilon \\ \tau_{1k}\otimes\tau_{2k}\otimes\tau_{4k} & \tau_{2k}\otimes\tau_{3k}\otimes\tau_{4k} & \tau_{2k}\otimes\tau_{3k}\otimes\tau_{4k} & \tau_{4k} & \varepsilon \\ \varepsilon & \varepsilon & \tau_{5k} & \tau_{5k} & \tau_{5k} \end{pmatrix}.$$

5 Tandem Queues

Tandem queueing systems present networks with the simplest topology determined by graphs which include only the nodes with no more than one incoming and outgoing arcs. Although no fork and join operations are actually performed in the systems, yet they may be treated as trivial fork-join networks and thus described using the representation proposed in the previous section.

Let us first consider a series of n single-server queues, depicted in Fig. 2. In this open tandem system, the queue labelled with 1 is assigned to represent an infinite external arrival stream of customers, with $r_1 = \infty$. The buffers of servers 2 to n are assumed to be empty at the initial time, so we set $r_i = 0$ for all $i = 2, \ldots, n$.

As one can see, equation (5) is now reduced to

$$a_i(k) = \begin{cases} \varepsilon, & \text{if } i = 1, \\ d_{i-1}(k), & \text{if } i \neq 1, \end{cases}$$

Proc. 1996 Summer Computer Simulation Conf., Portland, OR, July 21-25, 1996, SCS, 1996, 485-490

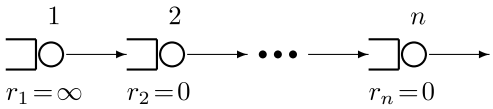

Figure 2: Open tandem queues.

Since $M = 0$, there is only one matrix

$$G_0 = \begin{pmatrix} \varepsilon & 0 & & \varepsilon \\ \vdots & \ddots & \ddots & \\ \vdots & & \ddots & 0 \\ \varepsilon & \cdots & \cdots & \varepsilon \end{pmatrix}$$

included in representation (7). It just presents an adjacency matrix of the graph describing the topology of the open tandem system. It is clear that this graph is acyclic; the length of its longest path $p = n - 1$.

Finally, with $G_1 = \mathcal{E}$, equation (9) becomes

$$\boldsymbol{d}(k) = (E \oplus \mathcal{T}_k \otimes G_0^T)^{n-1} \otimes \mathcal{T}_k \otimes \boldsymbol{d}(k-1),$$

which may be readily rewritten in the form of (1) with

$$T(k) = (E \oplus \mathcal{T}_k \otimes G_0^T)^{n-1} \otimes \mathcal{T}_k$$
$$= \begin{pmatrix} \tau_{1k} & \varepsilon & \cdots & \varepsilon \\ \tau_{1k} \otimes \tau_{2k} & \tau_{2k} & & \varepsilon \\ \vdots & \vdots & & \\ \tau_{1k} \otimes \cdots \otimes \tau_{nk} & \tau_{2k} \otimes \cdots \otimes \tau_{nk} & \cdots & \tau_{nk} \end{pmatrix}.$$

We now turn to a brief discussion of closed tandem systems. In the system shown in Fig. 3, the customers pass through the queues consecutively so as to get service at each server. After their service at queue n, the customers return to the 1st queue for a new cycle of service.

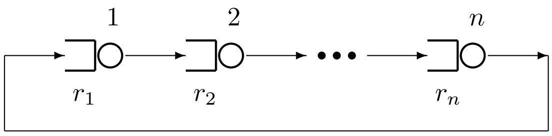

Figure 3: A closed tandem queueing system.

There is a finite number of customers in the system at the initial time; that is, we have $0 \leq r_i < \infty$ for all $i = 1, \ldots, n$. Note that the numbers r_i may all not equal 0 since at least one customer is to be present.

It is easy to understand that for the closed tandem system, equation (5) takes the form

$$a_i(k) = \begin{cases} d_n(k - r_1), & \text{if } i = 1, \\ d_{i-1}(k - r_i), & \text{if } i \neq 1, \end{cases}$$

whereas equations (7) and (8) remain unchanged. Since there is at least one customer in the system, the graph $\mathcal{G}_0$ associated with the matrix G_0 cannot coincide with the system graph $\mathcal{G}$, and so is acyclic. As a consequence, the conditions of Theorem 1 will be satisfied. We therefore conclude that the dynamics of the system is described by equation (9) and thus by (1).

Example 2 *Let us suppose that in the system,* $r_i = 1$ *for all* $i = 1, \ldots, n$*. Then we have* $G_0 = \mathcal{E}$ *and*

$$G_1 = \begin{pmatrix} \varepsilon & 0 & & \varepsilon \\ \vdots & \ddots & \ddots & \\ \varepsilon & \varepsilon & \ddots & 0 \\ 0 & \varepsilon & \cdots & \varepsilon \end{pmatrix}.$$

Clearly, equation (9) is now written as

$$\boldsymbol{d}(k) = \mathcal{T}_k \otimes (E \oplus G_1^T) \otimes \boldsymbol{d}(k-1);$$

that is, in the form of (1) with the matrix

$$T(k) = \mathcal{T}_k \otimes (E \oplus G_1^T)$$
$$= \begin{pmatrix} \tau_{1k} & \varepsilon & \cdots & \varepsilon & \tau_{1k} \\ \tau_{2k} & \tau_{2k} & & \varepsilon & \varepsilon \\ & \ddots & \ddots & & \\ & & \ddots & \ddots & \\ \varepsilon & \varepsilon & & \tau_{nk} & \tau_{nk} \end{pmatrix}.$$

6 A System with Round Routing

We consider an open system depicted in Fig. 4, which consists of $n = l+1$ queues labeled with $0, 1, \ldots, l$; queue

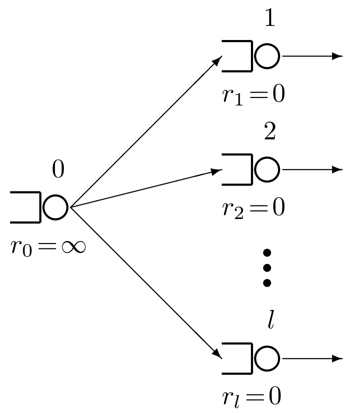

Figure 4: A queueing system with round routing.

0 is thought to represent an external arrival stream of customers. Each incoming customer has to go to one of the other queues, being chosen by a regular round routing mechanism. Upon completion of his service at the latter queue, the customer leaves the system.

The routing mechanism working in the system requires that the customer which is the first to depart queue 0, go

Proc. 1996 Summer Computer Simulation Conf., Portland, OR, July 21-25, 1996, SCS, 1996, 485-490

to the 1st queue, the second customer do to the 2nd queue, and so on. After the lth customer who is transferred to queue l, the next $(l+1)$st customer goes to the 1st queue once again, and the procedure is further repeated round and round.

To represent the dynamics of the system, let us first replace it by an equivalent fork-join network. An appropriate network, which is actually obtained by substituting new nodes $l+1$ to $2l$ for queue 0, is shown in Fig. 5.

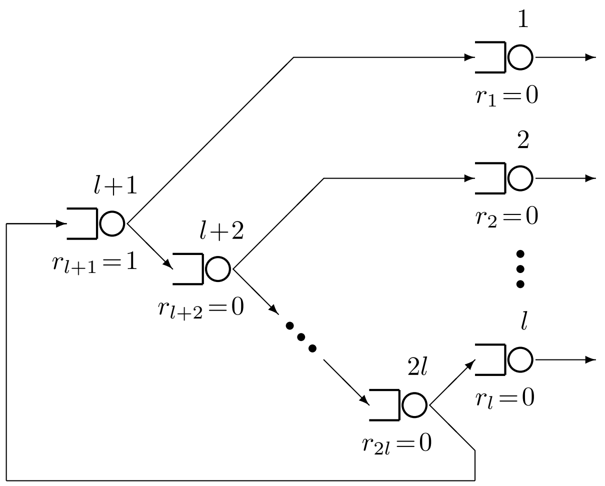

Figure 5: An equivalent fork-join network.

It is easy to see that, with the kth service time at each new node i, $i = l+1, \ldots, 2l$, defined as

$$\tau_{ik} = \tau_{0\,lk-2l+i},$$

the fork-join network behaves in much the same way as the original system with regular round routing. Specifically, the overall operation of nodes 1 to l in these queueing systems is identical, and thus both systems produce the same output stream of customers.

The dynamics of the fork-join network is described through equations (4) and (7) with $n = 2l$ and $M = 1$. Taking into account that equation (5) now becomes

$$a_i(k) = \begin{cases} d_{l+i}(k), & \text{if } i = 1, \ldots, l, \\ d_{2l}(k-1), & \text{if } i = l+1, \\ d_{i-1}(k), & \text{if } i = l+2, \ldots, n, \end{cases}$$

we may represent the matrices G_0 and G_1 as

$$G_0 = \begin{pmatrix} \mathcal{E}_{(l\times l)} & \mathcal{E}_{(l\times l)} \\ E_{(l\times l)} & F_{(l\times l)} \end{pmatrix}, \quad G_1 = \begin{pmatrix} \mathcal{E}_{(l\times l)} & \mathcal{E}_{(l\times l)} \\ \mathcal{E}_{(l\times l)} & H_{(l\times l)} \end{pmatrix},$$

where $\mathcal{E}_{(l\times l)}$ and $E_{(l\times l)}$ respectively denote the null and unit $(l \times l)$-matrices,

$$F_{(l\times l)} = \begin{pmatrix} \varepsilon & 0 & & \varepsilon \\ \vdots & \ddots & \ddots & \\ \vdots & & \ddots & 0 \\ \varepsilon & \cdots & \cdots & \varepsilon \end{pmatrix}, \quad H_{(l\times l)} = \begin{pmatrix} \varepsilon & \cdots & \cdots & \varepsilon \\ \vdots & \ddots & & \vdots \\ \varepsilon & & \ddots & \vdots \\ 0 & \varepsilon & \cdots & \varepsilon \end{pmatrix}.$$

As it is easy to verify, the graph associated with the matrix G_0 is acyclic, with the length of its longest path $p = l$. In this case, one can apply Theorem 1 so as to obtain equation (1) with the state transition matrix

$$T(k) = (E \oplus \mathcal{T}_k \otimes G_0^T)^l \otimes \mathcal{T}_k \otimes (E \oplus G_1^T).$$

Example 3 *Let us calculate the state transition matrix $T(k)$ for the system with $l = 3$. The above representation and idempotency in the max-plus algebra lead us to the matrix*

$$T(k) = \begin{pmatrix} \tau_{1k} & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \tau_{1k}\otimes\tau_{4k} \\ \varepsilon & \tau_{2k} & \varepsilon & \varepsilon & \varepsilon & \tau_{2k}\otimes\tau_{4k}\otimes\tau_{5k} \\ \varepsilon & \varepsilon & \tau_{3k} & \varepsilon & \varepsilon & \tau_{3k}\otimes\tau_{4k}\otimes\tau_{5k}\otimes\tau_{6k} \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \tau_{4k} \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \tau_{4k}\otimes\tau_{5k} \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \tau_{4k}\otimes\tau_{5k}\otimes\tau_{6k} \end{pmatrix}.$$

References

[Baccelli and Makowski 1989] Baccelli, F. and A.M. Makowski. 1989. "Queueing Models for Systems with Synchronization Constraints." *Proceedings of the IEEE* 77, no.1 (Jan.): 138-160.

[Cohen *et al.* 1989] Cohen, G; P. Moller; J.-P. Quadrat; and M. Viot. 1989. "Algebraic Tools for the Performance Evaluation of Discrete Event Systems." *Proceedings of the IEEE* 77, no.1 (Jan): 39-58.

[Cuninghame-Green 1979] Cuninghame-Green, R.A. 1979. *Minimax Algebra.* Lecture Notes in Economics and Mathematical Systems, 166. Springer-Verlag, Berlin.

[Greenberg *et al.* 1991] Greenberg, A.G.; B.D. Lubachevsky; and I. Mitrani. 1991. "Algorithms for Unboundedly Parallel Simulations." *ACM Transactions on Computer Systems* 9, no.3 (Aug.): 201-221.

[Krivulin 1994] Krivulin, N.K. 1994. "Using Max-Algebra Linear Models in the Representation of Queueing Systems." In *Proceedings of the 5th SIAM Conference on Applied Linear Algebra* (Snowbird, UT, June 15-18). SIAM, Philadelphia, PA, 155-160.

[Krivulin 1995] Krivulin, N.K. 1995. "A Max-Algebra Approach to Modeling and Simulation of Tandem Queueing Systems." *Mathematical and Computer Modelling* 22, no.3: 25-37.

[Olsder 1992] Olsder, G.J. 1992. "About Difference Equations, Algebras and Discrete Events." *CWI Syllabus* 31, 180-209.