TREE STRUCTURED QBD MARKOV CHAINS AND TREE-LIKE QBD PROCESSES

B. Van Houdt, **C. Blondia**

University of Antwerp
Department of Mathematics and Computer Science
Performance Analysis of Telecommunication Systems Research Group
Middelheimlaan, 1, B-2020 Antwerp - Belgium
{benny.vanhoudt,chris.blondia}@ua.ac.be

Abstract. In this paper we show that an arbitrary tree structured QBD Markov chain can be embedded in a tree-like QBD process with a special structure. Moreover, we present an algebraic proof that applying the natural fixed point iteration (FPI) to the nonlinear matrix equation $V = B + \sum_{s=1}^{d} U_s(I - V)^{-1}D_s$ that solves the tree-like QBD process, is equivalent to the more complicated iterative algorithm presented by Yeung and Alfa [1].

1 INTRODUCTION

Tree structured Quasi-Birth-Death (QBD) Markov chains were first introduced in 1995 by Takine et al [2] and later, in 1999, by Yeung et al [1]. More recently, Bini et al [3] have defined the class of tree-like QBD processes as a specific subclass of the tree structured QBD Markov chains. In this paper we show that an arbitrary tree structured QBD Markov chain can easily be embedded in a tree-like QBD process with a special structure. Moreover, we present an algebraic proof that applying the natural fixed point iteration (FPI) to the nonlinear matrix equation $V = B + \sum_{s=1}^{d} U_s(I - V)^{-1}D_s$ that solves the tree-like QBD process, is equivalent to the more complicated iterative algorithm presented by Yeung and Alfa [1]. Thus, Eqs. (22), (23), (28) and (29) presented in [1] are equivalent to calculating the inverse of a matrix. Apart from the FPI method, Bini, Latouche and Meini [3] have developed two more iterative algorithms for the solution of tree-like QBD processes. When combined with the construction presented in this paper, they can be used to solve any tree structured QBD Markov process.

In the past few years, tree structured QBD Markov chains have been shown to be useful in solving preemptive and non-preemptive single server queues with a LCFS service discipline that serves customers of different types, where each type has a different service requirement [2, 1, 4, 5]. Tree structured QBD Markov chains have also been used to evaluate conflict resolution algorithms of the Capetanakis-Tsybakov-Mikhailov-Vvedenskaya (CTMV) type [6, 7].

2 MARKOV CHAIN OF QUASI-BIRTH-DEATH TYPE WITH A TREE STRUCTURE - A REVIEW

Let us briefly describe a tree structured Quasi-Birth-Death (QBD) Markov chain. This type of Markov chain was first introduced in Takine et al [2] and Yeung, et al [1]. Consider a discrete time bivariate Markov chain $\{(X_t, N_t), t \geq 0\}$ in which the values of X_t are represented by nodes of a d-ary tree, and where N_t takes integer values between 1 and m. X_t is referred to as the node and N_t as the auxiliary variable of the Markov chain at time t. A description of the transitions of the Markov chain is given below.

A d-ary tree is a tree for which each node has d children. The root node is denoted as $\emptyset$. The remaining nodes are denoted as strings of integers, with each integer between 1 and d. For instance, the k-th child of the root node is represented by k, the l-th child of the node k is represented by kl, and so on. Throughout this paper we use lower case letters to represent integers and upper case letters to represent strings of integers when referring to nodes of the tree. We use '+' to denote concatenation on the right. For example, if $J = k_1\ k_2\ k_3$, then $J + k = k_1\ k_2\ k_3\ k$.

The Markov chain (X_t, N_t) is called a Markov chain of the QBD-type with a tree structure if at each step the chain can only make transitions from a node to its parent, children of its parent, or to its children. Moreover, if the chain is in state $(J + k, i)$ at time t, then the state at time $t + 1$ is determined as follows:

1. (J, j) with probability $d_k^{i,j}, k = 1, \ldots, d$,

2. $(J+s, j)$ with probability $a_{k,s}^{i,j}, k, s = 1, \ldots, d$,

3. $(J+ks, j)$ with probability $u_s^{i,j}, s = 1, \ldots, d$.

Define $m \times m$ matrices $D_k, A_{k,s}$ and U_s with respective $(i,j)^{th}$ elements given by $d_k^{i,j}, a_{k,s}^{i,j}$ and $u_s^{i,j}$. Notice that transitions from state $(J+k, i)$ do not dependent upon J, moreover, transitions to state $(J+ks, j)$ are also independent of k. When the Markov chain is in a state of the root node ($J = \emptyset$) at time t, then the state at time $t+1$ is determined as follows:

1. $(\emptyset, j)$ with probability $f^{i,j}$,

2. (k, j) with probability $u_k^{i,j}, k = 1, \ldots, d$.

Define the $m \times m$ matrix F with corresponding $(i,j)^{th}$ element given by $f^{i,j}$. A fundamental period of a tree structured QBD Markov chain that starts in the state $(J+k, i)$ is defined as the first passage time from the state $(J+k, i)$ to one of the m states (J, j) for $j = 1, \ldots, m$.

2.1 THE STATIONARY DISTRIBUTION OF THE QUEUE STRING

According to Yeung and Alfa [1] an ergodic QBD Markov chain with a tree structure has a matrix geometric stationary distribution. Define, for each string J and $1 \leq i \leq m$

$$\pi(J, i) = \lim_{t \to \infty} P[(X_t, N_t) = (J, i))]. \tag{1}$$

Denote by $\pi(J) = (\pi(J,1), \ldots, \pi(J,m))$. In order to calculate the $1 \times m$ vectors $\pi(J)$ the following three sets of $m \times m$ matrices play an important role [1].

Let $G_k, 0 \leq k \leq d$, denote the matrix whose $(i,v)^{th}$ element is the probability that the Markov chain (X_t, N_t) is in state (J, v) at the end of the fundamental period given that this period starts in state $(J+k, i)$. These matrices are stochastic for recurrent QBD Markov chains with a tree structure (Takine et al [2]). Let $R_k, 0 \leq k \leq d$, denote the matrix whose $(i,v)^{th}$ element is the expected number of visits to $(J+k, v)$ given that $(X_0, N_0) = (J, i)$ before visiting node J again. Let $V_k, 0 \leq k \leq d$, denote the matrix whose $(i,v)^{th}$ element is the taboo probability that starting from $(J+k, i)$, the chain eventually returns to a node with the same length as $J+k$ by visiting

$(J+k, v)$, under the taboo of the node J and the sibling nodes of $J+k$, i.e., the nodes $J+s, s \neq k$.

Yeung and Sengupta [5, Theorem 1] have shown that the following relationship holds for the vectors $\pi(J)$

$$\pi(J+k) = \pi(J) R_k, \tag{2}$$

where $k = 1, \ldots, d$ and the vector $\pi(\emptyset)$ is a left eigenvector of the matrix

$$F + \sum_{i=1}^{d} R_i D_i. \tag{3}$$

The vector $\pi(\emptyset)$ is normalized by

$$\pi(\emptyset) \left(I - \sum_{i=1}^{d} R_i \right) e = 1. \tag{4}$$

Thus, in order to obtain the steady-state probabilities $\pi(J)$, it suffices to determine the matrices R_i for $i = 1, \ldots, d$. Yeung and Alfa [1] have shown that the matrices R_k can be written as

$$R_k = \sum_{s=1}^{d} U_s \Lambda_{s,k}, \tag{5}$$

where the $\Lambda_{s,k}$ are $m \times m$ matrices that can be expressed in terms of V_k, with $k = 1, \ldots, d$, using Eq. (7-11) by replacing all matrices of the form $X[N]$ by X. The matrices V_k in their turn can be obtained from the following iterative formula [1, Theorem 2]:

$$V_k[N+1] = A_{k,k} + \sum_{s=1}^{d} U_s \left(\sum_{u=1}^{d} \Lambda_{s,u}[N] D_u \right), \tag{6}$$

where $V_k[0] = A_{k,k}$ and $\Lambda_{s,u}[N]$ are $m \times m$ matrices that are expressed in terms of the matrices $V_1[N], \ldots, V_d[N]$ as follows:

$$\Lambda_{k,k}[N] = T_k[N] \left[I - \sum_{u \neq k} A_{k,u} \left(\sum_{v \neq k} \Gamma_{u,v,k}^{0 \cup k}[N] \right) T_k[N] \right]^{-1}, \tag{7}$$

and

$$\Lambda_{k,s}[N] = T_k[N] \left[A_{k,s} + \sum_{u \neq k,s} A_{k,u} \left(\sum_{v \neq s} \Gamma_{u,v,s}^{0 \cup s}[N] \right) \right] \Lambda_{s,s}[N], \tag{8}$$

for $s \neq k$, where the matrices $T_k[N]$ are defined as $(I - V_k[N])^{-1}$, whereas the matrices $\Gamma^{0\cup\mathcal{K}}_{r,s,t}[N]$ with $\mathcal{K} \subseteq \{1,\ldots,d\}$, $r,s \in \{1,\ldots,d\} \setminus \mathcal{K}$ and $t \in \mathcal{K}$, are defined as

$$\Gamma^{0\cup\mathcal{K}}_{r,r,s}[N] = T_r[N] \left[I - \sum_{u \notin r\cup\mathcal{K}} A_{r,u} \left(\sum_{v \notin r\cup\mathcal{K}} \Gamma^{0\cup r\cup\mathcal{K}}_{u,v,r}[N] \right) T_r[N] \right]^{-1} A_{r,s}, \tag{9}$$

and for $t \neq r$,

$$\Gamma^{0\cup\mathcal{K}}_{r,t,s}[N] = T_r[N] \left[A_{r,t} + \sum_{u \notin r\cup t\cup\mathcal{K}} A_{r,u} \left(\sum_{v \notin t\cup\mathcal{K}} \Gamma^{0\cup t\cup\mathcal{K}}_{u,v,t}[N] \right) \right] \Gamma^{0\cup\mathcal{K}}_{t,t,s}[N]. \tag{10}$$

By repeatedly applying Eq. (9-10), all the matrices $\Gamma^{0\cup k}_{r,s,t}[N]$ occurring in Eqs. (7) and (8) can be expressed in term of the Γ-matrices $\Gamma^{0\cup\mathcal{K}_r}_{r,r,s}[N]$, where $\mathcal{K}_r = \{1,\ldots,d\} \setminus \{r\}$ and $s \in \mathcal{K}_r$, that is, $s \neq r$. From Eq. (9) it follows that

$$\Gamma^{0\cup\mathcal{K}_r}_{r,r,s}[N] = T_r[N] A_{r,s}. \tag{11}$$

As a result, the number of Γ-matrices that need to be calculated during each iterative step is $\sum_{s=1}^{d-1} \binom{d}{s} s(d-s)^2 \approx \frac{d^3 2^d}{8} = O(d^3 2^d)$, where the calculation of each of these Γ-matrices requires one or more $m \times m$ matrix products. Thus, each step requires at least $O(m^3 d^3 2^d)$ floating point multiplications (the exact number is a more complex function of d, but this lower bound suffices to given an idea of the complexity of these expressions). The complexity of this iterative scheme reduces significantly if the tree structured QBD Markov chain does not allow transitions between sibling nodes, that is, the matrices $A_{k,s} = 0$ for $k \neq s$. Yueng and Alfa [1] have shown that, in this case, the $\Lambda_{k,s}[N]$ matrices reduce to 0 if $k \neq s$ and to $T_k[N]$ for $k = s$. Hence, Eq. (6) reduces to

$$V_k[N+1] = A_{k,k} + \sum_{s=1}^{d} U_s (I - V_s[N])^{-1} D_s. \tag{12}$$

with $V_k[0] = A_{k,k}$.

It should be noted that the R_k matrices can also be obtained from the iteration

$$R_k[N+1] = U_k + \sum_{s=1}^{d} R_s[N] A_{s,k} + \sum_{s=1}^{d} R_k[N] R_s[N] D_s, \tag{13}$$

where $R_k[0] = 0$. A similar equation exists for the G_k matrices [1]. One of the advantages of Eq. (6) is that once we have obtained V_k—that is, $\Lambda_{k,s}$—we can calculate R_k and G_k from $\Lambda_{k,s}$ as follows [1]

$$G_k = \sum_{s=1}^{d} \Lambda_{k,s} D_s, \qquad R_k = \sum_{s=1}^{d} U_s \Lambda_{s,k}. \tag{14}$$

The matrices G_k are not used to calculate the steady state, however, they allow us to check whether the Markov chain is positive recurrent, that is, the chain is positive recurrent if and only if the matrices G_k are stochastic [1]. For instance, in [6] we always start by checking whether the G_k matrices are stochastic and if so, we calculate—among other things—the mean delay from the steady state probabilities. Another important advantage of Eq. (6) is discussed in Section 6.

3 TREE-LIKE QUASI-BIRTH-DEATH PROCESSES - A REVIEW

Tree-Like QBD processes were first introduced by Bini, Latouche and Meini [3] and can be defined as a tree structured QBD Markov chain that meets a few additional requirements. First, tree-like QBD processes do not allow transitions between sibling nodes, that is, $A_{k,s} = 0$ for $k \neq s$. Second, the following expression holds

$$A_{1,1} = A_{2,2} = \ldots = A_{d,d} = B, \tag{15}$$

for some matrix B. For tree-like QBD processes it is clear, from the probabilistic interpretation, that the matrices V_k are all identical, hence, we can drop the subscript k. The matrix V obeys the following nonlinear matrix equation

$$V = B + \sum_{s=1}^{d} U_s (I - V)^{-1} D_s. \tag{16}$$

Bini, Latouche and Meini [3] have introduced and compared a number of algorithms to solve this matrix equation. The natural fixed point iteration (FPI), first presented in [1], works as follows. Set $V[0] = B$ and calculate $V[N+1]$ as

$$V[N+1] = B + \sum_{s=1}^{d} U_s (I - V[N])^{-1} D_s. \tag{17}$$

The fact that $V[N]$ converges (linearly) to V has been proven as a special case in [1, Theorem2]. In the next section we show that any tree structured QBD Markov chain can be reduced to a tree-like QBD process by means of a simple construction.

Before we proceed, let us define a tree-like QBD process with a generalized boundary condition. Such a stochastic process is characterized by the matrices D_k, U_k, B, F, E, E_1 and E_2 and operates on the state space $\Omega \cup \{(\Phi, i) \mid 1 \leq i \leq m_1\}$, where Ω is the state space of an ordinary tree-like QBD process. The new node Φ is a parent node of $\emptyset$ and transitions from Φ to Φ, from Φ to $\emptyset$ and from $\emptyset$ to Φ, are characterized by the $m_1 \times m_1$ matrix E, the $m_1 \times m$ matrix E_1 and the $m \times m_1$

matrix E_2, respectively. The other matrices D_k, U_k, B and F are defined as above. The main step in obtaining the stationary distribution of such a process is, obviously, still the resolution of the nonlinear equation Eq. (16).

4 EMBEDDING TREE STRUCTURED QBD MARKOV CHAINS IN TREE-LIKE QBD PROCESSES

The idea behind the construction used to reduce a tree structured QBD Markov chain to a tree-like QBD process, has been applied before by He and Alfa [4] and by Van Houdt and Blondia [6, 7] where it was used to reduce special tree structured Markov chains that were not of the GI/M/1 type, see [5] for a definition, to a tree-like QBD process.

Consider an arbitrary tree structured QBD Markov chain (X_t, N_t). Such a Markov chain is characterized by the $m \times m$ matrices F, D_k, U_k and $A_{k,s}$, with $k, s = 1, \ldots, d$. First, we construct a tree-like QBD process $(\tilde{X}_t, \tilde{N}_t)$, with a generalized boundary condition, characterized by the $md \times md$ matrices $\tilde{F}, \tilde{B}, \tilde{D}_k, \tilde{U}_k$, with $k = 1, \ldots, d$, an $m \times m$ matrix $\tilde{E}$, an $m \times md$ matrix $\tilde{E}_1$ and an $md \times m$ matrix $\tilde{E}_2$. The random variables $\tilde{X}_t$ and $\tilde{N}_t$ are defined as follows. Consider a realization $(X_t(w), N_t(w))$ of the Markov chain (X_t, N_t). Whenever $(X_t(w), N_t(w))$ is of the form $(J + k, i)$ with $k \in \{1, \ldots, d\}$ and $i \in \{1, \ldots, m\}$, we define $\tilde{X}_t(w) = J$ and $\tilde{N}_t(w) = (k, i)$. Otherwise, that is, if $(X_t(w), N_t(w)) = (\emptyset, i)$ with $i \in \{1, \ldots, m\}$, we define $\tilde{X}_t(w) = \Phi$ and $\tilde{N}_t(w) = i$. It should be clear from its construction that $(\tilde{X}_t, \tilde{N}_t)$ is a Markov chain that operates on the state space $\Omega \cup \{(\Phi, i) \mid 1 \leq i \leq m\}$, where Ω is the state space of the Markov chain (X_t, N_t). Moreover, the construction indicates that there exists a one-to-one mapping between both state spaces. Hence, the Markov chain $(\tilde{X}_t, \tilde{N}_t)$ is positive recurrent if and only if (X_t, N_t) is positive recurrent and the steady state probabilities of (X_t, N_t) are easy to obtain from those of $(\tilde{X}_t, \tilde{N}_t)$.

Next, we consider the following three cases. First, suppose that (X_t, N_t) is in state $(\emptyset, i)$; therefore, $(\tilde{X}_t, \tilde{N}_t)$ is in state (Φ, i). In this case the chain (X_t, N_t) either makes a transition to state $(\emptyset, j)$ with a probability $F_{i,j}$ or to state (k, j) with probability $(U_k)_{i,j}$. Thus, the chain $(\tilde{X}_t, \tilde{N}_t)$ makes a transition to (Φ, j) or $(\emptyset, (k, j))$. As a result we have

$$\tilde{E} = F, \qquad \tilde{E}_1 = [U_1 \; U_2 \; U_3 \; \ldots \; U_d]. \tag{18}$$

Second, suppose that (X_t, N_t) is in state (k, i); therefore, $(\tilde{X}_t, \tilde{N}_t)$ is in state $(\emptyset, (k, i))$. Then (X_t, N_t) makes a transition to $(\emptyset, j)$ with probability $(D_k)_{i,j}$, to (s, j) with probability $(A_{k,s})_{i,j}$ or to $(k+s, j)$ with a probability $(U_s)_{i,j}$. Hence, $(\tilde{X}_t, \tilde{N}_t)$ makes a transition to (Φ, j), to $(\emptyset, (s, j))$ or to $(k, (s, j))$. Therefore, we find that

$$\tilde{E}_2 = \begin{bmatrix} D_1 \\ D_2 \\ \vdots \\ D_d \end{bmatrix}, \qquad \tilde{F} = \begin{bmatrix} A_{1,1} & A_{1,2} & \dots & A_{1,d} \\ A_{2,1} & A_{2,2} & \dots & A_{2,d} \\ \vdots & \vdots & \ddots & \vdots \\ A_{d,1} & A_{d,2} & \dots & A_{d,d} \end{bmatrix}, \tag{19}$$

whereas the matrices $\tilde{U}_k$, with $1 \leq k \leq d$, are found as

$$\tilde{U}_k = \begin{bmatrix} 0 & \dots & 0 & 0 & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & 0 & 0 & \vdots & 0 \\ U_1 & \dots & U_{k-1} & U_k & U_{k+1} & \dots & U_d \\ 0 & \dots & 0 & 0 & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \\ 0 & \dots & 0 & 0 & 0 & \dots & 0 \end{bmatrix}. \tag{20}$$

Notice that all the $m \times m$ blocks in $\tilde{U}_k$ are equal to zero except for the k-th row. Finally, suppose that (X_t, N_t) is in state $(J + r + k, i)$, with J an arbitrary string of zero or more integers; therefore, $(\tilde{X}_t, \tilde{N}_t)$ is in state $(J + r, (k, i))$. Then (X_t, N_t) makes a transition to $(J + r, j)$ with probability $(D_k)_{i,j}$, to $(J + r + s, j)$ with probability $(A_{k,s})_{i,j}$ or to $(J+r+k+s, j)$ with a probability $(U_s)_{i,j}$. Hence, $(\tilde{X}_t, \tilde{N}_t)$ makes a transition to $(J, (r, j))$, to $(J+r, (s, j))$ or to $(J+r+k, (s, j))$. As a result, we find that $\tilde{B} = \tilde{F}$ and the matrices $\tilde{D}_k$ are equal to

$$\tilde{D}_k = \begin{bmatrix} 0 & \dots & 0 & D_1 & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & D_{k-1} & 0 & \vdots & 0 \\ 0 & \dots & 0 & D_k & 0 & \dots & 0 \\ 0 & \dots & 0 & D_{k+1} & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \\ 0 & \dots & 0 & D_d & 0 & \dots & 0 \end{bmatrix}. \tag{21}$$

Notice that all the $m \times m$ blocks in $\tilde{D}_k$ are equal to zero except for the k-th column. Eqs. (18–21) fully characterize the tree-like QBD process, with a generalized boundary condition. To obtain the steady state probabilities we need to solve Eq. (16) where the matrices V, B, D_s and U_s are replaced by $\tilde{V}, \tilde{B}, \tilde{D}_s$ and $\tilde{U}_s$, that is,

$$\tilde{V} = \tilde{B} + \sum_{s=1}^{d} \tilde{U}_s (I - \tilde{V})^{-1} \tilde{D}_s. \tag{22}$$

We can resolve this equation by means of Eq. (17). From the probabilistic interpretation of $\tilde{V}$, or from Eq. (22) and the structural properties of $\tilde{D}_s$ and $\tilde{U}_s$, it should be clear that $\tilde{V}$ can be written as

$$\tilde{V} = \begin{bmatrix} \tilde{V}_1 & A_{1,2} & \dots & A_{1,d-1} & A_{1,d} \\ A_{2,1} & \tilde{V}_2 & \dots & A_{2,d-1} & A_{2,d} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ A_{d-1,1} & A_{d-1,2} & \dots & \tilde{V}_{d-1,d-1} & A_{d-1,d} \\ A_{d,1} & A_{d,2} & \dots & A_{d,d-1} & \tilde{V}_d \end{bmatrix}, \tag{23}$$

where $A_{k,s}$ are the A-matrices of (X_t, N_t). Moreover, from the probabilistic interpretation of $\tilde{V}_k$ it should be clear that $\tilde{V}_k = V_k$, where $V_k, 1 \leq k \leq d$, are the V-matrices related to the tree structured QBD Markov chain (X_t, N_t). As a result Eq. (22) reduces to

$$\tilde{V}_k[N+1] = A_{k,k} + \sum_{s=1}^{d} U_s \left(\sum_{u=1}^{d} \tilde{\Lambda}_{s,u}[N] D_u \right), \tag{24}$$

where the matrices $\tilde{\Lambda}_{s,u}[N]$ are found as follows. Define $\tilde{V}[N]$ as

$$\tilde{V}[N] = \begin{bmatrix} \tilde{V}_1[N] & A_{1,2} & \dots & A_{1,d-1} & A_{1,d} \\ A_{2,1} & \tilde{V}_2[N] & \dots & A_{2,d-1} & A_{2,d} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ A_{d-1,1} & A_{d-1,2} & \dots & \tilde{V}_{d-1,d-1}[N] & A_{d-1,d} \\ A_{d,1} & A_{d,2} & \dots & A_{d-1,d} & \tilde{V}_d[N] \end{bmatrix}, \tag{25}$$

then $\tilde{\Lambda}_{k,s}[N]$ is the $m \times m$ matrix found on the k-th row and the s-th column of $(I - \tilde{V}[N])^{-1}$. Next, we will prove that $\tilde{\Lambda}_{k,s}[N] = \Lambda_{k,s}[N]$, where $\Lambda_{k,s}[N]$ are the Λ-matrices of (X_t, N_t) defined by Eqs. (7–11). Thus, Eqs. (7–11)—these equation were obtained by Yeung and Alfa [1] based on their probabilistic interpretation—describe how to obtain an explicit expression for the inverse of $I - \tilde{V}$ in terms of $A_{k,s}$ and $\tilde{V}_k[N]$. We start by considering two special cases.

4.1 NO TRANSITIONS BETWEEN SIBLING NODES

Suppose that (X_t, N_t) does not allow transitions between sibling nodes, that is, $A_{k,s} = 0$ for $k \neq s$. As noted in Section 2.1, Eq. (6) reduces to Eq. (12). Moreover, $\tilde{V}$, defined in Eq. (25), is a block diagonal matrix; therefore, Eq. (24), obviously reduces to Eq. (12). As a result, we find that $V_k[N] = \tilde{V}_k[N]$.

4.2 THE CASE d = 2

Suppose that each node of (X_t, N_t) has two child nodes, that is, $d = 2$. In this case one finds, by means of Eqs. (7–11), that

$$\Lambda[N] = \begin{bmatrix} \Lambda_{1,1}[N] & \Lambda_{1,2}[N] \\ \Lambda_{2,1}[N] & \Lambda_{2,2}[N] \end{bmatrix} =$$
$$\begin{bmatrix} T_1[N]\left[I - A_{1,2}T_2[N]A_{2,1}T_1[N]\right]^{-1} & T_1[N]A_{1,2}\Lambda_{2,2}[N] \\ T_2[N]A_{2,1}\Lambda_{1,1}[N] & T_2[N]\left[I - A_{2,1}T_1[N]A_{1,2}T_2[N]\right]^{-1} \end{bmatrix}.$$

It is easily checked that $(I - \tilde{V}[N])\ \Lambda[N] = I$; hence, $V_k[N] = \tilde{V}_k[N]$.

4.3 THE GENERAL CASE

The easiest way to prove the general case is to use the probabilistic interpretation of the $\Gamma^{0\cup\mathcal{K}}_{r,t,s}[N]$ and $\Lambda_{k,s}[N]$ matrices. In this section we present an algebraic proof. Because $V_k[0] = \tilde{V}_k[0]$, it is sufficient to prove that

$$\begin{bmatrix} T_1[N]^{-1} & -A_{1,2} & \dots & -A_{1,d} \\ -A_{2,1} & T_2[N]^{-1} & \dots & -A_{2,d} \\ \vdots & \vdots & \ddots & \vdots \\ -A_{d,1} & -A_{d,2} & \dots & T_d[N]^{-1} \end{bmatrix} \begin{bmatrix} \Lambda_{1,1}[N] & \Lambda_{1,2}[N] & \dots & \Lambda_{1,d}[N] \\ \Lambda_{2,1}[N] & \Lambda_{2,2}[N] & \dots & \Lambda_{2,d}[N] \\ \vdots & \vdots & \ddots & \vdots \\ \Lambda_{d,1}[N] & \Lambda_{d,2}[N] & \dots & \Lambda_{d,d}[N] \end{bmatrix} = I, \tag{26}$$

where I is the $md \times md$ unity matrix and the matrices $T_k[N]$ and $\Lambda_{k,s}[N]$ were defined in Section 2.1.

Lemma 1. *Equation (26) is satisfied if*

$$\sum_{v\neq s} \Gamma^{0\cup s}_{i,v,s}[N] = T_i[N]\left[A_{i,s} + \sum_{u\neq i,s} A_{i,u}\left(\sum_{v\neq s}\Gamma^{0\cup s}_{u,v,s}[N]\right)\right], \tag{27}$$

for $i \neq s$.

Proof. Let $k \neq s$ and premultiply (27) by $A_{k,i}$, take the sum over $i \neq k, s$, postmultiply by $\Lambda_{s,s}[N]$ and add $A_{k,s}\Lambda_{s,s}[N]$ to find that

$$\left[A_{k,s} + \sum_{i\neq k,s} A_{k,i}\left(\sum_{v\neq s}\Gamma^{0\cup s}_{i,v,s}[N]\right)\right]\Lambda_{s,s}[N] = \tag{28}$$
$$A_{k,s}\Lambda_{s,s}[N] + \sum_{i\neq k,s} A_{k,i}T_i[N]\left[A_{i,s} + \sum_{u\neq i,s} A_{i,u}\left(\sum_{v\neq s}\Gamma^{0\cup s}_{u,v,s}[N]\right)\right]\Lambda_{s,s}[N].$$

By means of Eq. (8) we can rewrite this as

$$T_k[N]^{-1}\Lambda_{k,s}[N] = A_{k,s}\Lambda_{s,s}[N] + \sum_{i\neq k,s} A_{k,i}\Lambda_{i,s}[N]. \tag{29}$$

This proves that the k-th row multiplied by the s-th column of Eq. (26) is zero for $k \neq s$. For $k = s$, we premultiply (27) by $A_{s,i}$, sum over $i \neq s$, postmultiply by $-T_s[N]$ and add I, the $m \times m$ unity matrix, to find that

$$\left[I - \sum_{i\neq s} A_{s,i}\left(\sum_{v\neq s}\Gamma^{0\cup s}_{i,v,s}[N]\right)T_s[N]\right] = \tag{30}$$
$$I - \sum_{i\neq s} A_{s,i}T_i[N]\left[A_{i,s} + \sum_{u\neq i,s} A_{i,u}\left(\sum_{v\neq s}\Gamma^{0\cup s}_{u,v,s}[N]\right)\right]T_s[N].$$

Using Eqs. (7) and (8), Eq. (30) can be written as

$$I = T_s[N]^{-1}\Lambda_{s,s}[N] - \sum_{i\neq s} A_{s,i}\Lambda_{i,s}[N]. \tag{31}$$

This completes the proof of this lemma. □

Let $\mathcal{K} \subseteq \{1,\ldots,d\}$ and denote $|\mathcal{K}|$ as the number of elements in the set $\mathcal{K}$, then we have the following lemma:

Lemma 2. *Let $\mathcal{K} \subseteq \{1,\ldots,d\}\setminus\{i\}$, with $1 \leq |\mathcal{K}| \leq d-1$, then*

$$\sum_{v\notin\mathcal{K}}\Gamma^{0\cup\mathcal{K}}_{i,v,s}[N] = T_i[N]\left[A_{i,s} + \sum_{u\notin\mathcal{K}\cup i} A_{i,u}\left(\sum_{v\notin\mathcal{K}}\Gamma^{0\cup\mathcal{K}}_{u,v,s}[N]\right)\right], \tag{32}$$

where $s \in \mathcal{K}$.

Proof. We use backward induction on the size $|\mathcal{K}|$ of the set $\mathcal{K}$. Let $|\mathcal{K}| = d-1$; hence, $\mathcal{K} = \{1,\ldots,d\}\setminus\{i\}$. For such a set $\mathcal{K}$, in view of Eq. (11), Eq. (32) reduces to

$$\Gamma^{0\cup\mathcal{K}}_{i,i,s}[N] = T_i[N]A_{i,s}. \tag{33}$$

Suppose that Eq. (32) is valid for any set $\mathcal{K}$ with $|\mathcal{K}| > n \geq 1$. We now show that Eq. (32) is valid for any set $\mathcal{K}$ with $|\mathcal{K}| = n$. By means of Eq. (10) we have

$$\sum_{v\notin\mathcal{K}\cup i}\Gamma^{0\cup\mathcal{K}}_{i,v,s}[N] = \sum_{v\notin\mathcal{K}\cup i} T_i[N]\left[A_{i,v} + \sum_{u\notin\mathcal{K}\cup i\cup v} A_{i,u}\left(\sum_{t\notin\mathcal{K}\cup v}\Gamma^{0\cup\mathcal{K}\cup v}_{u,t,v}[N]\right)\right]\Gamma^{0\cup\mathcal{K}}_{v,v,s}[N].$$

$$(34)$$

The size of $\mathcal{K}_1$, defined as $\mathcal{K} \cup v$, is $n+1$; therefore, we find, for $u \notin \mathcal{K}_1$, by induction that

$$\begin{aligned}
&\left(\sum_{t \notin \mathcal{K} \cup v} \Gamma^{0 \cup \mathcal{K} \cup v}_{u,t,v}[N] \right) \Gamma^{0 \cup \mathcal{K}}_{v,v,s}[N] \\
&\quad = T_u[N] \left[A_{u,v} + \sum_{w \notin \mathcal{K} \cup u \cup v} A_{u,w} \left(\sum_{x \notin \mathcal{K} \cup v} \Gamma^{0 \cup \mathcal{K} \cup v}_{w,x,v}[N] \right) \right] \Gamma^{0 \cup \mathcal{K}}_{v,v,s}[N] \\
&\quad = \Gamma^{0 \cup \mathcal{K}}_{u,v,s}[N],
\end{aligned} \tag{35}$$

where the second equality follows from Eq. (10). If we combine Eq. (34) with Eq. (35), we obtain

$$\begin{aligned}
\sum_{v \notin \mathcal{K} \cup i} \Gamma^{0 \cup \mathcal{K}}_{i,v,s}[N] &= \sum_{v \notin \mathcal{K} \cup i} T_i[N] \left[A_{i,v} \Gamma^{0 \cup \mathcal{K}}_{v,v,s}[N] + \sum_{u \notin \mathcal{K} \cup i \cup v} A_{i,u} \Gamma^{0 \cup \mathcal{K}}_{u,v,s}[N] \right] \\
&= T_i[N] \sum_{v \notin \mathcal{K} \cup i} \sum_{u \notin \mathcal{K} \cup i} A_{i,u} \Gamma^{0 \cup \mathcal{K}}_{u,v,s}[N] \\
&= T_i[N] \sum_{u \notin \mathcal{K} \cup i} A_{i,u} \sum_{v \notin \mathcal{K} \cup i} \Gamma^{0 \cup \mathcal{K}}_{u,v,s}[N] \\
&= T_i[N] \sum_{u \notin \mathcal{K} \cup i} A_{i,u} \sum_{v \notin \mathcal{K}} \Gamma^{0 \cup \mathcal{K}}_{u,v,s}[N] - T_i[N] \sum_{u \notin \mathcal{K} \cup i} A_{i,u} \Gamma^{0 \cup \mathcal{K}}_{u,i,s}[N].
\end{aligned}$$

Therefore, Eq. (32) is fulfilled if

$$\Gamma^{0 \cup \mathcal{K}}_{i,i,s}[N] = T_i[N] A_{i,s} + T_i[N] \sum_{u \notin \mathcal{K} \cup i} A_{i,u} \Gamma^{0 \cup \mathcal{K}}_{u,i,s}[N]. \tag{36}$$

Using Eq. (35), this is equivalent with

$$\Gamma^{0 \cup \mathcal{K}}_{i,i,s}[N] = T_i[N] A_{i,s} + T_i[N] \sum_{u \notin \mathcal{K} \cup i} A_{i,u} \sum_{t \notin \mathcal{K} \cup i} \Gamma^{0 \cup \mathcal{K} \cup i}_{u,t,i}[N] \Gamma^{0 \cup \mathcal{K}}_{i,i,s}[N]. \tag{37}$$

Next, we define $\Upsilon^{0 \cup \mathcal{K}}_i[N] = \sum_{u \notin \mathcal{K} \cup i} A_{i,u} \sum_{t \notin \mathcal{K} \cup i} \Gamma^{0 \cup \mathcal{K} \cup i}_{u,t,i}[N]$ and rewrite Eq. (37) as

$$T_i[N]^{-1} \Gamma^{0 \cup \mathcal{K}}_{i,i,s}[N] = A_{i,s} + \Upsilon^{0 \cup \mathcal{K}}_i[N] \Gamma^{0 \cup \mathcal{K}}_{i,i,s}[N]. \tag{38}$$

By means of Eq. (9) we find

$$\left[I - \Upsilon^{0 \cup \mathcal{K}}_i[N] T_i[N] \right]^{-1} A_{i,s} = A_{i,s} + \Upsilon^{0 \cup \mathcal{K}}_i[N] T_i[N] \left[I - \Upsilon^{0 \cup \mathcal{K}}_i[N] T_i[N] \right]^{-1} A_{i,s}, \tag{39}$$

and this equation is equivalent to

$$I = \left[I - \Upsilon_i^{0\cup\mathcal{K}}[N]T_i[N]\right] + \Upsilon_i^{0\cup\mathcal{K}}[N]T_i[N]. \tag{40}$$

This completes the proof of this lemma. □

If we combine this lemma for $\mathcal{K} = \{s\}$ with Lemma 1 we obtain the following theorem.

Theorem 1. *The matrices $V_k[N]$ obtained through Eqs. (6–11) are equal to the matrices $\tilde{V}_k[N]$ obtained through Eqs. (24–25), that is, $\Lambda_{k,s}[N] = \tilde{\Lambda}_{k,s}[N]$ for all $N \geq 0$.*

If we use Eq. (24–25) to obtain the matrices $V_k[N+1]$ from $V_k[N]$, we need approximately $O(m^3d^3)$ floating point multiplications, a significant improvement, for d large, over Eqs. (6–11) that require at least $O(m^3d^32^d)$ multiplications. Recall that $O(m^3d^32^d)$ was a rough underestimation of the actual number of multiplications required to perform (6-11).

We end this section by considering the following special case. Suppose that the tree structure QBD Markov chain (X_t, N_t) allows transitions between sibling nodes $J+k$ and $J+s$ if $k \geq s$, that is, $A_{k,s} = 0$ if $k < s$. Such a Markov chain was presented in [7] in order to analyze the modified binary CTM algorithm (in this paper we reduced the chain to a tree-like QBD process based on the specific characteristics of the application). Obviously, the matrices $\tilde{V}$ and $\tilde{V}[N]$ are lower triangular block matrices, therefore, their inverse can be calculated by $O(m^3d^2)$ multiplications.

5 STRUCTURAL PROPERTIES OF THE $\tilde{G}_k$ AND $\tilde{R}_k$ MATRICES

In this section we study the structure of the $\tilde{G}_k$ and $\tilde{R}_k$ matrices related to the Markov chain $(\tilde{X}_t, \tilde{N}_t)$. Yeung and Alfa [1, Section 6.1] have shown that $\tilde{G}_k = (I - \tilde{V}_k)\tilde{D}_k$. As a result of Eq. (21), we find that the matrix $\tilde{G}_k$ is zero except for the k-th block column. Moreover, because of Theorem 1 and Eq. (14) we find

$$\tilde{G}_k = \begin{bmatrix} 0 & \dots & 0 & G_1 & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & G_k & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & G_d & 0 & \dots & 0 \end{bmatrix}, \tag{41}$$

where G_i are the G-matrices related to the Markov chain (X_t, N_t). Similarly we find

$$\tilde{R}_k = \begin{bmatrix} 0 & \dots & 0 & 0 & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ R_1 & \dots & R_{k-1} & R_k & R_{k+1} & \dots & R_d \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & 0 & 0 & \dots & 0 \end{bmatrix}, \tag{42}$$

where R_i are the R-matrices related to the Markov chain (X_t, N_t).

6 NUMERICAL EXAMPLES

In this section we provide some numerical examples to indicate that calculating the R_k matrices from the V_k matrices, that is, using Eq. (14), might be much more efficient compared to applying Eq. (13) where the matrices R_k are found without first retrieving the V_k matrices.

We consider the same application as Yeung and Alfa in [1]. Consider a preemptive resume Last Come First Served MAP/PH/1 queue in continuous time. The arrival process is described by two sub-generator matrices F_0 and F_1 with dimensions m. The service times follow a d dimensional phase type distribution represented by (β, S), with $S_0 = -Se$. A customer whose service is preempted at phase j will resume his service at phase j when it reenters the service system. This queue can be modeled by a tree structured QBD Markov chain characterized by the matrices F, D_k, U_k, $A_{k,s}$ as indicated in [1]. Next, we consider 12 numerical examples, six with $d = 3$ and six with $d = 6$. For each of these examples we calculate the R_k matrices following two different approaches (a comparison with similar results can be conducted for the G_k matrices). First, we compute R_k, for $k = 1, \dots, k$, by means of Eq. (13). Second, we reduce the tree structured QBD Markov chain to a tree-like QBD Markov process and apply Eq. (24) —this equation was proven to be equivalent to Eq. (6). When we compare both methods we will refer to the number of iterations required. It should be noted however that each iteration of Eq. (24) requires d times as many floating point multiplications as Eq. (13).

For $d = 3$, we define $\beta = [0.1\ 0.3\ 0.6]$, $m = 2$ and the matrices S, F_0 and F_1 as follows:

$$S = \begin{bmatrix} -5 & 0.5 & 1 \\ 0.2 & -4/q & 0.1 \\ 0.1 & 0 & -4/q \end{bmatrix}, \quad F_0 = \begin{bmatrix} -5 & 5 - 3/q \\ 4 - 3/q & -4 \end{bmatrix} \quad F_1 = \begin{bmatrix} 1.5/q & 1.5/q \\ 2/q & 1/q \end{bmatrix}.$$

	$d=3$		$d=6$	
q	*Eq. (13)*	*Eq. (24)*	*Eq. (13)*	*Eq. (24)*
1	232	84	192	71
2	424	75	313	56
4	902	77	686	56
6	1539	85	1263	65
8	2439	98	2268	83
10	3823	120	4508	125

Table 1: *Number of Iterations required*

The six examples are obtained by setting $q = 1, 2, 4, 6, 8$ and 10. A higher value of q results in a longer service time and in a smaller arrival rate. For each of the six examples we have a stable queue. For $d = 6$, we use the same arrival process as with $d = 3$. We define $\beta = [0.1\ 0.1\ 0.1\ 0.15\ 0.25\ 0.3]$ and the matrix S as

$$S = \begin{bmatrix} -5 & 0.3 & 0.2 & 0 & 0.5 & 0.5 \\ 0.05 & -4/q & 0.1 & 0.05 & 0 & 0.1 \\ 0.05 & 0 & -4/q & 0.03 & 0.02 & 0 \\ 0.02 & 0.05 & 0.05 & -5 & 0.1 & 0.5 \\ 0.01 & 0 & 0.05 & 0.04 & -5/q & 0.01 \\ 0 & 0.05 & 0.05 & 0.1 & 0.1 & -4/q \end{bmatrix}.$$

Again, six examples are obtained by setting $q = 1, 2, 4, 6, 8$ and 10. The influence of q on the queue is the same as before. The results are presented in Table 1, using the following stopping criterion:

$$f\left(\sum_{k=1}^{d} V_k[N+1] - \sum_{k=1}^{d} V_k[N]\right) < 10^{-10}, \tag{43}$$

where $f(X)$ denotes the sum of all the entries of a matrix X.

The results in Table 1 are not difficult to explain. It should be clear from Eq. (13), that the (i, v)-th element of the matrix $R_k[N]$, found by Eq. (13), holds the expected number of visits to state $(J + k, v)$ given that $(X_0, N_0) = (J, i)$ before visiting the node J again and this by means of a path of length N or less. Whereas, the (i, v)-th element of $R_k[N]$, found by Eq. (24), also holds the expected number of visits to the state $(J + k, v)$ given that $(X_0, N_0) = (J, i)$ before visiting the node J again, but this time by means of a path of arbitrary length that only uses the levels $|J| + 1$ to $|J| + N + 1$ (see [1, 3]). Thus, the number of iterations required by Eq. (24) is always smaller. Moreover, as the transitions between sibling nodes become more frequent, the difference between the number of iterations required by both algorithms increases. In our numerical example, we noticed that increasing q results in fewer arrivals, but longer service times. Thus, the number of transitions between sibling nodes increases significantly when q increases. This explains the results in Table 1. In conclusion, whenever an application has a high number of transitions between sibling nodes, compared to the number of transitions to parent or child nodes, it is more efficient to calculate the R_k matrices by means of the V_k matrices.

7 CONCLUSIONS

In this paper we have shown that an arbitrary tree structured QBD Markov chain can easily be embedded in a tree-like QBD process with a special structure. Moreover, we presented an algebraic proof that applying the natural fixed point iteration (FPI) to the nonlinear matrix equation $V = B + \sum_{s=1}^{d} U_s(I - V)^{-1}D_s$ that solves the tree-like QBD process, is equivalent to the more complicated iterative algorithm presented by Yeung and Alfa [1]. Thus, Eqs. (22), (23), (28) and (29) presented in [1] are equivalent to calculating the inverse of a matrix. Apart from the FPI method, Bini, Latouche and Meini [3] have developed two more algorithms for the solution of tree-like QBD processes. When combined with the construction presented in this paper they can be used to solve any tree structured QBD Markov process. Finally, we have also shown that for applications that have a high number of transitions between sibling nodes, compared to the number of transitions to parent or child nodes, it is more efficient to calculate the R_k matrices by means of the V_k matrices.

ACKNOWLEDGEMENTS

The first author is a postdoctoral Fellow of the FWO-Flanders. We would like to thank the reviewers for their valuable comments.

References

[1] Yeung, R.W.; Alfa, A.S. The quasi-birth-death type markov chain with a tree structure. Stochastic Models, **1999**, *15* (4), 639–659.

[2] Takine, T.; Sengupta, B.; Yeung, R.W. A generalization of the matrix M/G/1 paradigm for markov chains with a tree structure. Stochastic Models, **1995**, *11* (3), 411–421.

[3] Bini, D. A.; Latouche, G.; Meini, B. Solving nonlinear matrix equations arising in tree-like stochastic processes. Linear Algebra Appl., **2003**, *366*, 39–64.

[4] He, Q.; Alfa, A.S. The discrete time MMAP[K]/PH[K]/1/LCFS-GPR queue and its variants. In *Proc. of the 3rd Int. Conf. on Matrix Analytic Methods*, pages 167–190, Leuven (Belgium), 2000.

[5] Yeung, R.W.; Sengupta, B. Matrix product-form solutions for markov chains with a tree structure. Adv. Appl. Prob., **1994**, *26*, 965–987.

[6] Van Houdt, B.; Blondia, C. Throughput of Q-ary splitting algorithms for contention resolution in communication networks. To appear in Adv. in Perf. Eval., **2003**.

[7] Van Houdt, B.; Blondia, C. Stability and performance of stack algorithms for random access communication modeled as a tree structured QBD markov chain. Stochastic Models, **2001**, *17* (3), 247–270.