STABILITY AND PERFORMANCE OF STACK ALGORITHMS FOR RANDOM ACCESS COMMUNICATION MODELED AS A TREE STRUCTURED QBD MARKOV CHAIN

B. Van Houdt, C. Blondia

University of Antwerp
Department of Mathematics and Computer Science
Performance Analysis of Telecommunication Systems Research Group
Universiteitsplein, 1, B-2610 Antwerp - Belgium
{vanhoudt,blondia}@uia.ua.ac.be
http://win-www.uia.ac.be/u/pats/

ABSTRACT

In this paper, we introduce an analytical model to study the stability and the main performance measures of a binary stack algorithm for random multiple access communication. The input traffic is a discrete time Batch Markovian Arrival Process (D-BMAP). The analytical model is nearly exact (one minor approximation is required) and the analysis is based on recent results obtained from tree structured Quasi-Birth-Death (QBD) Markov chains. Apart from studying the stability of the protocol, we are also able to calculate the mean delay and other important performance measures. The method deployed in this paper can also be extended to evaluate other medium access control (MAC) protocols with an underlying stack structure.

1 INTRODUCTION

Although the techniques used in this paper are applicable to the evaluation of any medium access control (MAC) protocol with an underlying stack structure, we limit ourselves to the well known Capetanakis-Tsybakov-Mikhailov (CTM) protocol with free access. The CTM protocol has been studied for the last two decades by many researchers, starting with Capetanakis [1], Tsybakov, Massey [2, 3] and many others in the late seventies and early eighties, with Flajolet [4], Fayolle [5], Greenberg [6] and others in the late eighties and with Seri and Sidi [7] in the nineties. All these analytical models assume Poisson input traffic. Sidi, et al [7] evaluate the protocol with Markovian capture (two states: *good* and *bad*) but still adopt the Poisson assumption for the input traffic. The CTM protocol has been used in a variety of communications systems and is currently considered as the random access scheme for the IEEE 802.14 standard of the hybrid fiber coaxial (HFC) networks which are evolving from the existing residential CATV networks [8].

Some recent developments on the subject of M/G/1 type of Markov chains with a tree structure [9, 10, 11, 12, 13], allow us to extend some of these results when the Poisson input traffic is replaced by a discrete time batch Markovian arrival process (D-BMAP). To our best knowledge, it is the first time that Markov chains with a tree structure are used to evaluate a medium access control protocol. So far, Markov chains with a tree structure were mainly used to evaluate LCFS queueing systems with multiple classes of customers, each class having a different service requirement [12, 13].

The paper is organized as follows. Section 2 provides a short description of the CTM protocol with free access. In Section 3 we briefly recall the definition of a D-BMAP, whereas Section 4 reviews a Quasi-Birth-Death Markov chain with a tree structure. Next, the analytical model is presented in Section 5. The procedure required to calculate the steady state probabilities is given in Section 6. Section 7 considers the stability issues. An overview of the performance measures of interest is provided in Section 8. Finally, some numerical examples are presented in Section 9 and conclusions are drawn in Section 10.

2 THE CAPETANAKIS-TSYBAKOV-MIKHAILOV (CTM) ALGORITHM WITH FREE ACCESS

Let us briefly summarize the most salient features of the CTM protocol [1, 5, 4]. A single channel (bus, cable, broadcast medium) is shared among many users (sources, nodes, stations) that transmit packetized messages. Time is slotted and transmissions are assumed to occur at the beginning of a time slot. Each time slot has a fixed duration equal to the length of a packet. Each transmission is within the reception range of every user (in a wireless centralized LAN environment the base station could broadcast the result of each uplink transmission).

The CTM protocol is a collision resolution algorithm for which each user strives to retransmit its colliding packet till it is correctly received. The users have to resolve this contention without the benefit of any additional information on other users' activity. The CTM protocol separates users that collide recursively according to some randomization procedure into two groups. The users of the first group attempt retransmission in the next slot, while the users of the second group wait until the first group is resolved.

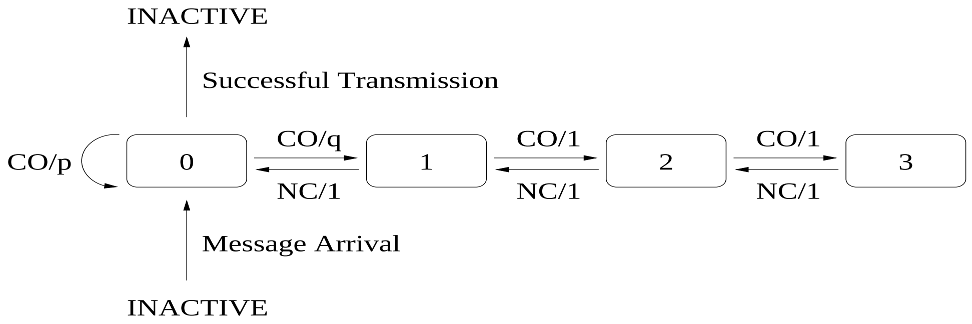

Figure 1: *State Diagram: CO = collision, NC = no collision*

The number of users, i.e., stations, is assumed to be infinite and each user holds zero or one packet. Users that hold a packet (at time t) are referred to as active users (at time t). The CTM protocol is conveniently implemented by letting each active user maintain an integer value, referred to as the current stack level. At the end of each time slot the current stack level is updated as follows (see Figure 1). A user that became active, i.e., generated a new packet, during slot $t-1$ initializes its current stack level for slot t at zero. A user is allowed to transmit in time slot t whenever its current stack level for slot t is zero. Therefore, users that became active during slot $t-1$ transmit in slot t (together with other stations that have their current stack level for slot t at zero). Suppose that slot t does not hold a collision, i.e., at most one user has its current stack level for slot t at zero. Then, users with a current stack level for slot t equal to $i, i > 0$, set their current stack level for slot $t+1$ at $i-1$ (while a successful user possibly becomes inactive). If slot t however does hold a collision, users with a current stack level for slot t equal to $i, i > 0$, set their current stack level for slot $t+1$ at $i+1$. While, users with a current stack level for slot t equal to zero split into two groups: a user joins the first group with a probability p and the second group with a probability $q = 1-p$. All the users that join the first group set their current stack level for slot $t+1$ at zero, while the users that join the second group set their current stack level for slot $t+1$ at one. An example of the transmission process is included in Figure 2. Figure 2 also includes a list of group numbers (1 or 2) for each packet to indicate which group the packet joins after each collision (in which it is involved). Thus, the list $1, 2, ...$ for packet E indicates that packet E joins the first group as a result of its first collision and the second as a result of its second collision.

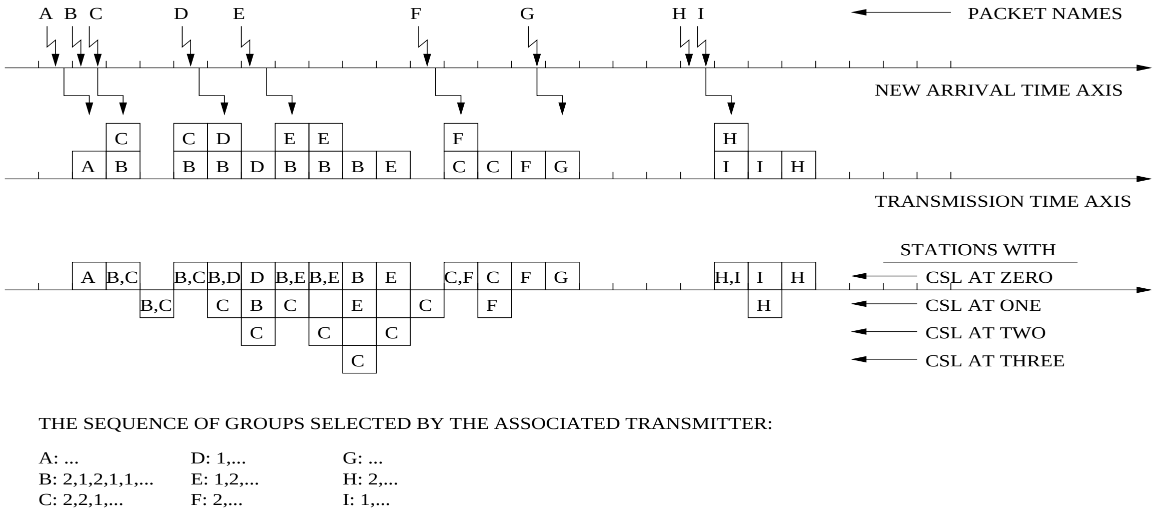

Figure 2: *Example of the Transmission Process: CSL = Current Stack Level*

Although out of the scope of this paper, the CTM protocol with free access can be improved by an adjustment that *saves doomed slots* [14, 5, 4]. Another variation of the CTM algorithm is the Q-ary CTM algorithm (after a collision stations split into Q subsets instead of two). We have successfully extended the techniques described in this paper in order to evaluate both abovementioned protocol variations.

3 DISCRETE TIME BATCH MARKOVIAN ARRIVAL PROCESSES (D-BMAP)

In this section, we briefly describe a class of tractable Markovian arrival processes commonly known as D-BMAPs, which, in general, are non-renewal and which include the discrete time variants of the Markov-modulated Poisson process, the PH-renewal process and superpositions of such processes as particular cases. Because of its versatility, it lends itself very well to modeling bursty arrival processes commonly arising in computer and communications applications.

Consider a discrete-time Markov chain on the state space $\{1, \ldots, l\}$ with transition matrix B. Suppose that at time t this chain is in some state i, $1 \leq i \leq l$. At the next time instant $t+1$, there occurs a transition to another or possibly the same state and a batch arrival may or may not occur: with probability $(b_0)_{i,j}$, $1 \leq i \leq l$, there is a transition to state j without an arrival, and with probability $(b_n)_{i,j}$, $1 \leq i \leq l, n \geq 1$, there is a transition to state j with a batch arrival of size n. A D-BMAP is fully characterized by the sequence of matrices $B_n, n \geq 0$, with elements $(b_n)_{i,j}$. Notice that $B = \sum_n B_n$. For a more detailed discussion we refer to [15].

In our model we use a single D-BMAP to determine when and how many users become active. Thus, if the D-BMAP is in state $i, 1 \leq i \leq l$, at the start of time slot t, n users become active, i.e., generate a packet, and the state at the start of time slot $t+1$ equals $j, 1 \leq j \leq l$, with a probability $(b_n)_{i,j}$. These n stations that became active are assumed to transmit their packet (for the first time) in time slot t (i.e., the arrivals are assumed to occur on the slot boundary of slot $t-1$ and slot t).

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

Let us briefly describe a tree structured Quasi-Birth-Death (QBD) Markov chain. This type of Markov chains was first introduced in Takine, et al [11] and Yeung, et al [9]. 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 Q-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 Q-ary tree is a tree for which each node has Q children. The root node is denoted as $\emptyset$. The remaining nodes are denoted as strings of integers, with each integer between 1 and Q. 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 $J = 1\ 0\ 8, k = 6$ then $J + k = 1\ 0\ 8\ 6$.

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 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 $d_k^{i,j}, k = 1, \ldots, Q$,
2. $(J+s, j)$ with probability $a_{k,s}^{i,j}, k, s = 1, \ldots, Q$,
3. $(J+ks, j)$ with probability $u_s^{i,j}, s = 1, \ldots, Q$.

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 the root state $(J = \emptyset)$ 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 $u_k^{i,j}, k = 1, \ldots, Q$.

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 $j = 1, \ldots, m$. For a more detailed description of the notations and algebra see Yeung, et al [9].

5 MARKOVIAN MODEL FOR THE CTM PROTOCOL

5.1 A First Attempt

The system at time slot t is fully specified by the state of the D-BMAP at the start of time slot $t+1$ and the current stack level for slot t of each active station. The value of all these current stack levels can be specified by a single string $s_k s_{k-1} \ldots s_1 s_0$, where s_i specifies the number of active stations with a current stack level for slot t equal to i. Therefore, the system is fully characterized by the Markov chain (V_t, W_t), where W_t denotes the state of the D-BMAP at the start of time slot $t+1$ and V_t represents the string that holds the current stack level for slot t of all active stations. It is easy to see that (V_t, W_t) is a tree structured Markov chain. Indeed, the node $s_k s_{k-1} \dots s_1 s_0$ is the parent of the nodes $s_k s_{k-1} \dots s_1 s_0 s$ for $s \geq 0$. Each node, including the root node $\emptyset$, contains l states (the l states of the D-BMAP) and has an infinite number of children. The root node, denoted as $\emptyset$, represents the case when there are no active stations.

The chain (V_t, W_t) is not of the Quasi-Birth-Death type. For instance, suppose that the chain is in the state (J, i) with $J = 2\ 5$ at time t. Therefore, 5 active stations have their current stack level for slot t at zero, i.e., transmit in slot t, and 2 active stations have their current stack level for slot t at one. Next, suppose that 3 of the five stations increase their current stack level to one as a result of the coin flip procedure. When a colliding station determines to join either the first or the second group, it is said to flip a coin (if $p = 1/2$ a fair coin). The coin flipping of all colliding stations is referred to as the coin flipping procedure. Then, at time $t+1$, the Markov chain is in the state (K, j) with $K = 2\ 3\ (2+s)$ with probability $(b_s)_{i,j}$ (i.e., s new arrivals occured on the boundary of slot t and slot $t+1$). This type of transitions (to the grandchildren of the parent node) is not allowed in a tree structured QBD Markov chain. Also, the Markov chain is no longer of the $GI/M/1$ type (see Yeung, et al [10]) and there is no simple or explicit solution for its stationary distribution.

5.2 The Actual Model

In order to solve the problem indicated in the previous section we make the number of stations with a current stack level for slot t equal to zero a part of the auxiliary variable. Active stations that have a current stack level for slot t larger than zero are referred to as backlogged stations (at time t). Consider the following Markov chain (X_t, N_t). Let X_t be the string holding the current stack level for slot t of all backlogged stations (at time t). For instance, when $X_t = s_k \dots s_2 s_1$ there are $\sum_{i=1}^{k} s_i$ backlogged stations, for $s_i \geq 0$ backlogged stations the current stack level for slot t is equal to i. In this example there are no stations with a current stack level for slot t larger than k. The sample space of the random variable X_t is $\Omega_1 = \{\emptyset\} \cup \{J : J = s_k \dots s_1, s_j \geq 0, 1 \leq j \leq k, k \geq 1\}$. Notice, the string J is allowed to have a number of leading zeros (see Note 1 for more comments on this issue). The random variable X_t has a tree structure. For instance, the children of $3\ 5\ 0\ 2$ are $3\ 5\ 0\ 2\ s, s \geq 0$. Thus, each node in the tree has an infinite number of children. N_t holds both the number of active stations with a current stack level for slot t equal to zero and the state of the D-BMAP at the start of slot $t + 1$. The sample space of the random variable N_t is $\Omega_2 = \{(n, j) \mid n \geq 0, 1 \leq j \leq l\}$.

It is easy to see that (X_t, N_t) is a Markov chain. The state space of the Markov chain is $\Omega_1 \times \Omega_2$. In order to solve this Markov chain the nodes of X_t should have a finite number of children and the auxiliary variable N_t should have a finite number of states. Therefore, the Markov chain (X_t, N_t) is approximated by another bivariate Markov chain (X_t^d, N_t^d). (X_t^d, N_t^d) is obtained by setting a maximum d on the number of stations that can have the same current stack level for slot t (including stack level zero). If a situation occurs in which $d+k, k > 0$, stations have the same current stack level for slot t, k stations are assumed to drop their packet. Thus, introducing d can cause stations to drop their packet. Packets are otherwise never dropped by a station. Nevertheless, provided that d is chosen sufficiently large there should hardly be any difference between the performance measures of (X_t^d, N_t^d) and (X_t, N_t). We state that d is chosen sufficiently large if the ratio of dropped packets due to the introduction of d is smaller than 10^{-9}, i.e., if less than one in a billion packets is dropped. The introduction of the parameter d is the only approximation required to evaluate the CTM protocol with free access. There is no obvious relationship between a sufficiently large value for d and the maximum n for which $B_n \neq 0$ (also such an n does not necessarily exist). For instance, a sufficiently large d for the bulk arrival process with $v = [4]$, as defined in Section 9, is found for $d \geq 18$ for $L = 10$, $d \geq 12$ for $L = 80$ and $d \geq 10$ for $L = 800$ (whereas $B_n = 0, n \geq 5$ for all three cases).

Let us now consider (X_t^d, N_t^d) in more detail. X_t^d is the string that holds the current stack level for slot t of all backlogged stations. For instance, when $X_t^d = s_k \dots s_2 s_1$ then for s_i backlogged stations the current stack level for slot t equals i. The sample space of the random variable X_t^d is $\Omega_1^d = \{\emptyset\} \cup \{J : J = s_k \dots s_1, 0 \leq s_j \leq d, 1 \leq j \leq k, k \geq 1\}$. X_t^d has a tree structure, e.g., 3 10 0 8 $s, 0 \leq s \leq d$, are children of 3 10 0 8. Therefore, each node in Ω_1^d has $d+1$ children. As opposed to the general description of the tree structured QBD Markov chain we represent the children of a node by 0 to d instead of 1 to $d+1$. N_t^d represents the number of stations that transmit in slot t (i.e., the current stack level for slot t of these stations is zero) and the state of the D-BMAP at the start of slot $t+1$. The sample space of N_t^d is $\Omega_2^d = \{(n, j) \mid 0 \leq n \leq d, 1 \leq j \leq l\}$. It is easy to see that (X_t^d, N_t^d) is a Markov chain and the state space of the Markov chain (X_t^d, N_t^d) is $\Omega_1^d \times \Omega_2^d$.

We now proof that the transitions made by the Markov chain (X_t^d, N_t^d) are either transitions to a child or a parent node (except from the root node $\emptyset$). Assume that the Markov chain (X_t^d, N_t^d) is in node $J+k$ at time t, i.e., $X_t^d = J+k$. If slot t contains a collision of $c \geq 2$ stations, i.e., N_t^d is of the form (c,j) with $c \geq 2, 1 \leq j \leq l$, all backlogged stations increment their current stack level by one. Thus, the integers in the string $J+k$ shift one position to the left and $X_{t+1}^d = J+ks$ with $0 \leq s \leq c$ (s of the c colliding stations set their current stack level for slot $t+1$ at 1 as a result of the coin flip). N_{t+1}^d is determined by j, c and the probability that a station selects the first group p. Thus, a collision in slot t causes the Markov chain to make a transition to a child node (this is also the case for $X_t^d = \emptyset$). If slot t does not hold a collision, all backlogged stations decrement their current stack level by one, i.e., shift one position to the right. Hence, if slot t does not hold a collision, the chain will be in the parent node J at time slot $t+1$ (for $X_t^d = \emptyset$ the chain remains in the root node). In conclusion, the chain can only make transitions from a node to either its parent node or to one of its children.

In order for the Markov chain (X_t^d, N_t^d) to be a tree structured QBD Markov chain the following two additional conditions have to be satisfied. First, the probability of making a transition from state $(J+k,(i,j))$ to state $(J,(i',j'))$ may not dependent upon J. As noted above, such a transition takes place whenever slot t does not hold a collision. Clearly, j', the new state of the D-BMAP, is solely determined by j, the old state of the D-BMAP, and thus independent of J. While, i', the number of stations that transmit in slot $t+1$, is determined by k, the number of stations that decrease their current stack level from one to zero, and j, the old state of the D-BMAP (because this state j determines the number of new arrivals on the boundary of slot t and slot $t+1$).

Second, the probability of making a transition from state $(J+k,(i,j))$ to state $(J+ks,(i',j'))$ may not dependent upon J and k. Such a transition occurs whenever slot t does hold a collision. Again, j', the new state of the D-BMAP, is determined by j, the old state of the D-BMAP. While, s, the number of stations that increase their current stack level to one (as a result of the coin flipping), is determined by i and the probability p. Finally, i', the number of stations that transmit in slot $t+1$, is determined by i, p and j, the old state of the D-BMAP (because this state j determines the number of new arrivals).

In conclusion, the Markov chain (X_t^d, N_t^d) is a tree structured QBD Markov chain. A tree structured QBD Markov chain is fully characterized by the matrices D_k, U_s, $A_{k,s}$ and F (see Section 4). The matrices $A_{k,s}$ hold the transition probabilities that the chain (X_t^d, N_t^d)

goes from state $(J+k,(i,j))$ to the state $(J+s,(i',j'))$. These transitions are transitions between sibling nodes. Remember that the chain (X_t^d, N_t^d) can only make transitions to its parent or to its children, therefore, the entries of the matrices $A_{k,s}$ are zero. This fact reduces the memory and time requirements of the algorithm to calculate the steady state probabilities of (X_t^d, N_t^d) when it is ergodic (for details see Section 6).

The matrices D_k hold the transition probabilities that the chain (X_t^d, N_t^d) goes from state $(J+k,(i,j))$ to the state $(J,(i',j'))$. This happens when slot t does not hold a collision. Therefore, the state i, the number of stations that transmit in slot t, must be equal to 0 or 1. Moreover, the state i', the number of stations that transmit in slot $t+1$, equals k, the number of stations that decrease their current stack level from one to zero, plus some possible new arrivals. Hence,

$$D_k((i,j),(i',j')) = \begin{cases} (B_{i'-k})_{j,j'} & i \leq 1, i' \geq k, i' < d, \\ \sum_{n \geq d-k} (B_n)_{j,j'} & i \leq 1, i' \geq k, i' = d, \\ 0 & otherwise, \end{cases} \quad (1)$$

where $(B_n)_{j,j'}$ holds the probability that n new arrivals occur and that the input D-BMAP changes its state from j to j' (see Section 3).

The matrices U_s hold the transition probabilities that the chain (X_t^d, N_t^d) goes from state $(J+k,(i,j))$ to the state $(J+ks,(i',j'))$. This happens when slot t holds a collision. Therefore, the state i, the number of stations that transmit in slot t, must be larger than or equal to 2. Moreover, the state i', the number of stations that transmit in slot $t+1$, equals i, the number of stations that transmitted in slot t, minus s, the number of stations that increase their current stack level to one (as a result of the coin flipping), plus some possible new arrivals. Clearly, s can never be larger than i. Hence,

$$U_s((i,j),(i',j')) = \begin{cases} C_s^i p^{i-s} q^s (B_{i'-(i-s)})_{j,j'} & i > 1, i \geq s, i' \geq i-s, i' < d, \\ C_s^i p^{i-s} q^s \sum_{n \geq d-(i-s)} (B_n)_{j,j'} & i > 1, i \geq s, i' \geq i-s, i' = d, \\ 0 & otherwise, \end{cases} \quad (2)$$

where $(B_n)_{j,j'}$ holds the probability that n new arrivals occur and that the input D-BMAP changes its state from j to j' (see Section 3) and C_s^i denotes the number of different possible combinations of s from i different items.

Assume that the Markov chain is in node $J = \emptyset$ at time t, i.e., $X_t^d = \emptyset$. Then the transitions to the nodes s, $0 \leq s \leq d$, are governed by the matrices U_s, whereas the transitions to the root node $\emptyset$ are as follows. The matrix F holds the transition probability that chain

(X_t^d, N_t^d) goes from state $(\emptyset, (i,j))$ to the state $(\emptyset, (i',j'))$. This happens whenever slot t does not hold a collision, i.e., $i \leq 1$. The state i', the number of stations that transmit in slot $t+1$, equals the number of new arrivals (occuring on the boundary of slot t and slot $t+1$). Hence,

$$F((i,j),(i',j')) = \begin{cases} (B_{i'})_{j,j'} & i \leq 1, i' < d, \\ \sum_{n \geq d} (B_n)_{j,j'} & i \leq 1, i' = d, \\ 0 & otherwise, \end{cases} \qquad (3)$$

where $(B_n)_{j,j'}$ holds the probability that n new arrivals occur and that the input D-BMAP changes its state from j to j' (see Section 3).

Note 1: It is possible that a string J has a number of leading zeros. The semantics of such a string J is identical to that of the string J without the leading zeros. For instance, $J = 0\ 0\ 4\ 0\ 5$ has the same meaning as $K = 4\ 0\ 5$. Strings with leading zeros arise from the following situation. When the Markov chain (X_t^d, N_t^d) is in the root state $J = \emptyset$, i.e., $X_t^d = \emptyset$, a transition might occur to state 0. For instance, suppose that $N_t^d = (c, j)$, with $c \geq 2, 1 \leq j \leq l$, and assume that the current stack level for slot $t+1$ is set at zero for each of the c colliding stations (as a result of the coin flip procedure). Then, at time $t+1$, according to Equation (2), the Markov chain (X_t^d, N_t^d) is in the node 0. It might seem more appropriate to remain in the root node in such cases and thus to avoid strings with leading zeros. If we exclude this type of transitions and thus eliminate such strings, the node variable X_t^d would have a tree structure where every node has $d+1$ children except for the root node (who has d children). In Yeung, et al [10] this type of Markov chain is called a Markov chain with a forest structure and algorithms to calculate the steady state are provided. Both approaches lead to the same steady state probabilities (after rearranging the states appropriately). The advantage of allowing this type of transitions is that we get a slightly faster algorithm because the boundary condition is slightly less complicated.

6 THE STATIONARY DISTRIBUTION OF THE QUEUE STRING

According to Yeung and Alfa [9], a matrix geometric solution exists for an ergodic QBD Markov chain with a tree structure. The Markov chain (X_t^d, N_t^d) is aperiodic whenever the D-BMAP modeling the input traffic is aperiodic. The irreducibility is not always inherited from the input D-BMAP, e.g., D-BMAPs with $B_0 = 0$ or $B_n = 0, n \geq 2$. In Section 7 we address the problem of determining whether the Markov chain (X_t^d, N_t^d) is positive recurrent. Define, for each string $J \in \Omega_1^d$, $0 \leq i \leq d$ and $1 \leq j \leq l$

$$\pi(J,(i,j)) = \lim_{t\to\infty} P[(X_t^d, N_t^d) = (J,(i,j))]. \tag{4}$$

Denote by $\pi(J,i) = (\pi(J,(i,1)),\ldots,\pi(J,(i,l)))$ and by $\pi(J) = (\pi(J,0),\ldots,\pi(J,d))$. In order to calculate the $1 \times l(d+1)$ vectors $\pi(J)$ the following three sets of $l(d+1) \times l(d+1)$ matrices play an important role [9].

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^d, N_t^d) is in state (J,v) at the end of the fundamental period given that this period starts from state $(J+k,i)$. These matrices are stochastic for recurrent QBD Markov chains with a tree structure (Takine, et al [11]). 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 $(X_0^d, N_0^d) = (J,i)$ 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 Alfa [9] have shown that the matrices G_k and R_k can be expressed in terms of V_k. Moreover, because the QBD Markov chain (X_t^d, N_t^d) does not allow transitions between sibling nodes, they were able to shown that the following simple expressions hold

$$\begin{aligned} G_k &= (I - V_k)^{-1} D_k, & (5)\\ R_k &= U_k (I - V_k)^{-1}, & (6)\\ V_k &= A_{k,k} + \sum_{s=0}^{d} U_s G_s. & (7) \end{aligned}$$

If however transitions between sibling nodes where allowed it would still be possible to solve the chain but the equations would be more complicated and the resulting iterative scheme more time consuming [9]. Notice that the matrices $V_k, 0 \leq k \leq d$, are identical if the matrices $A_{k,k}, 0 \leq k \leq d$, are identical. For the Markov chain (X_t^d, N_t^d) the matrices $A_{k,k}, 0 \leq k \leq d$, are equal to zero, therefore the matrices $V_k, 0 \leq k \leq d$, are identical. In the remaining part of this section we drop the subscript k if we refer to V_k. Using equations

(5) and (7), we obtain

$$V = \sum_{s=0}^{d} U_s (I - V)^{-1} D_s. \tag{8}$$

As a special case of Theorem 2 in Yeung and Alfa [9], the matrix V can be obtained as $\lim_{N \to \infty} V[N]$ from the recursion

$$V[N+1] = \sum_{s=0}^{d} U_s (I - V[N])^{-1} D_s, \tag{9}$$

where $V[0] = 0$. Also, the matrices $G_s[N] = (I - V[N])^{-1} D_s$ converge to the substochastic matrices G_s. Since we do not know in advance whether the Markov chain (X_t^d, N_t^d) is stable we do not use the possible stochastic nature of the matrices G_s as a stopping criterion for the recursion in (9). We simply repeat the recursion until all matrices $G_k, 0 \leq k \leq d$, are stabilized.

Next, the matrices $R_k, 0 \leq k \leq d$, are calculated from the matrix V using equation (6). The steady state vectors $\pi(J)$ are then calculated as follows [9]

$$\pi(J + k) = \pi(J) R_k, \tag{10}$$

where $\pi(\emptyset)$ is the left invariant vector of the matrix $F + V$, i.e., $\pi(\emptyset)(F + V) = \pi(\emptyset)$, and $\pi(\emptyset)$ is normalized as $\pi(\emptyset)(I - R)^{-1}\mathbf{e} = 1$. The matrix R is defined as $\sum_{s=0}^{d} R_s$. In order to clarify the subsequent steps required to calculate the steady state probabilities we have summarized them in the following algorithm:

Algorithm:

- INPUT: a sequence of matrices $B_n, n \geq 0$, that characterize the D-BMAP input traffic.

- STEP 1: calculate the matrices $D_k, 0 \leq k \leq d$, $U_s, 0 \leq s \leq d$, and F by making use of formulas (1), (2) and (3).

- STEP 2: determine the matrix V using the iterative formula presented in (9).

- STEP 3: calculate the matrices $R_k, 0 \leq k \leq d$, by means of equation (6).

- STEP 4: determine the vector $\pi(\emptyset)$ as follows: $\pi(\emptyset) = \pi(\emptyset)(F+V)$, where $\pi(\emptyset)$ is normalized as $\pi(\emptyset)(I-R)^{-1}\mathbf{e} = 1$.
- STEP 5: calculate de steady state probabilities of interest using the equation $\pi(J+k) = \pi(J)R_k$.

REMARK: At the end of STEP 4 one can determine whether the parameter d was chosen sufficiently large (see Note 2), if not, d has to be increased and the first four steps have to be repeated, i.e., everything has to be recalculated. For most numerical examples $d = 10$ was sufficient (see Section 9). Thus, one starts with $d = 2$ and repeats the first 4 steps until d is sufficiently large. It is however possible to reduce the the computational effort by making a first estimate for the starting value of d (instead of $d = 2$). If we estimate the value of d larger than the smallest possible d for which d is sufficiently large, we are finished after one run. One must however note that the larger we choose d, the more time it requires to compute the first four steps. Therefore, one should try to limit of the margin of overestimation. During the numerical trials we noticed that there exists a strong relationship between a sufficiently large d and the burstiness, i.e., the variation on the number of arrivals in a time slot, of the input process. We used the following heuristic method to reduce the computation times: if $d = x$ was sufficiently large for a specific D-BMAP and the next D-BMAP we are about to evaluate is more, resp. less, bursty we make use of a larger, resp. smaller, first estimate for a sufficiently large d.

Note 2: We can make use of the following test to determine whether d was chosen sufficiently large. Let ρ be the load, i.e., arrival rate, of the D-BMAP modeling the aggregated input traffic. From the steady state probabilities we can calculate $\sum_{J,j} \pi(J,(1,j))$. This sum is, due to the law of total probability, equal to the probability that there is exactly one active station with a current stack level for slot t equal to zero. Therefore, this sum matches the probability of having a successful transmission. We can now compare this with the arrival rate, i.e., load, of the D-BMAP to get a value for the ratio of dropped packets. In conclusion, we state that d is chosen sufficiently large whenever $(\rho - \sum_{J,j} \pi(J,(1,j)))/\rho < 10^{-9}$.

7 STABILITY ISSUES

Flajolet and Jacquet [4] have shown that the CTM algorithm with free access is stable under a Poisson flow of arrivals if the arrival rate $\lambda < 0.360177$ (using fair coins, i.e., for $p = 1/2$). The CTM algorithm is said to be stable if the expected delay suffered by an arbitrary packet is finite. In this section we indicate how to determine whether the CTM protocol is stable under D-BMAP traffic. Define $\mathcal{S}$ as the set of all (irreducible) D-BMAPs. $\mathcal{S}$ can be splitted into two subsets $\mathcal{S}_1$ and $\mathcal{S}_2$ such that the CTM algorithm with free access is stable for $s \in \mathcal{S}_1$ and is unstable for $s \in \mathcal{S}_2$. For instance, the CTM algorithm is stable for all D-MAPs, i.e., D-BMAPs with $B_n = 0$ for $n \geq 2$.

A D-BMAP s belongs to $\mathcal{S}_1$ if and only if the Markov chain (X_t, N_t) is stable, i.e., positive recurrent. To test whether the Markov chain (X_t, N_t) is positive recurrent, we study the stability of the Markov chain (X_t^d, N_t^d). Clearly, the chain (X_t, N_t) is transient whenever the chain (X_t^d, N_t^d) is transient. Indeed, (X_t^d, N_t^d) behaves identical to (X_t, N_t) except that it drops a packet from time to time. Clearly, this only improves the expected delay suffered by an arbitrary packet. The stability of the chain (X_t^d, N_t^d) is however not sufficient to proof that the chain (X_t, N_t) is stable. For instance, for every $s \in \mathcal{S}$, (X_t^1, N_t^1) is stable. Even when d is chosen sufficiently large, it is still possible that the dropping of these rare packets (even when we lose less than one in a billion) causes the chain (X_t^d, N_t^d) to become stable while (X_t, N_t) is not. Hence, it is possible that we slightly overestimate the stability point of a particular arrival process. There exists only one case we can use to get an idea of the margin of overestimation: the Poisson result. Numerical results (not included in Section 10) have indicated that for $d = 10$ the overestimation is less than 0.0003% (the chain was unstable for $\lambda = 0.36018$ while the exact result by Flajolet states 0.360177). Further increasing d would result in even smaller overestimation errors.

The Markov chain (X_t^d, N_t^d) is recurrent if and only if the matrices $G_k, 0 \leq k \leq d$, are stochastic (HE [12]). Provided that the Markov chain (X_t^d, N_t^d) is recurrent, we define a heuristic measure d_s for its stability as follows. Let $\pi(i,j), 0 \leq i \leq d$ and $1 \leq j \leq l$, be the probability that the auxiliary variable N_t^d is equal to (i,j). Hence, $\pi(i,j) = \sum_J \pi(J,(i,j)) = \pi(\emptyset)(I-R)^{-1}$ (see Section 6). Let $d_s = \sum_j \pi(0,j) + \sum_j \pi(1,j) - \sum_{j,i>1} \pi(i,j)$. d_s can be seen as the difference between the drift towards the root node and the drift away from the root node. Indeed, $\sum_j \pi(0,j)$ is equal to the probability that slot

t is empty, i.e., no transmission takes place in slot t and $\sum_j \pi(1,j)$ is the probability that slot t holds a successful transmission. Therefore, $\sum_j \pi(0,j) + \sum_j \pi(1,j)$ is the probability that the Markov chain makes a transition to a parent node. While, $\sum_{j,i>1} \pi(i,j)$ represents the probability that a collision takes place in slot t, i.e., that the chain makes a transition to a child node. The difference between these two probabilities is used as a measure for the stability.

8 PERFORMANCE MEASURES

8.1 The Fundamental Period and Mean Delay

Define $\Phi_1(i,j), 0 \leq i \leq d$ and $1 \leq j \leq l$, as the expected length of a fundamental period given that this period starts from state $(J+k,(i,j))$. Notice that these expected values do not depend upon k. $\Phi_1(i,j)$ is the expected number of time slots necessary to resolve a collision of i stations provided that the D-BMAP is in state j. Let $\Phi_1(i) = (\Phi_1(i,1),\ldots,\Phi_1(i,l))$ and $\Phi_1 = (\Phi_1(0),\ldots,\Phi_1(d))$. Then, the column vector Φ_1^t (x^t denotes the transposed vector of x) obeys the following equation

$$\Phi_1^t = \mathbf{e} + \sum_{s=0}^{d} U_s[\Phi_1^t + G_s\Phi_1^t]. \tag{11}$$

This equation is obtained as follows. The expected length of the fundamental period equals one if the first slot of the period is collision free, i.e., if i equals zero or one (the first $2(d+1)$ rows of U_s are zero, i.e., $\Phi_1(i,j) = 1$ for $i = 0$ or 1). Otherwise, if the first slot holds a collision, the expected length of the fundamental period equals one (the first slot) plus the expected time required to resolve the first group plus the expected time required to resolve the second group. In order to calculate the expected time required to resolve the first group we apply the law of total probability on the state of the D-BMAP at the end of the second slot of the fundamental period (the state at the end of the first is j), on the number of colliding stations that select the second group and on the number of new arrivals occuring on the slot boundary of the first and second slot of the fundamental period. In matrix form this leads to $\sum_s U_s\Phi_1^t$. For time required to resolve the second group we also apply the law of total probability on the state of the D-BMAP at the end of the slot following the fundamental period initiated by the first group and on the number of new arrivals on the boundary of the last slot of the fundamental period initiated by the first group and the first of the period initiated by the second group. In matrix form this leads to $\sum_s U_s G_s \Phi_1^t$. Equation (11) can be solved as a set of linear equations or using an iterative method.

Define $\Upsilon(k,j), 1 \leq k \leq d$ and $1 \leq j \leq l$, as the probability that $N_t^d = (k,j)$ at an arrival instant. Details on how to calculate $\Upsilon(k,j)$ are provided in Appendix A. Thus, the probability that the transmission of a packet is successful at its first attempt is $\sum_j \Upsilon(1,j)$. Let $U(delay)$ be

$$U(delay) = \sum_{i=1}^{d} \sum_{j=1}^{l} \Upsilon(i,j) \Phi_1(i,j). \tag{12}$$

Then $U(delay)$ is an upper bound on the mean delay experienced by an arbitrary packet. It is possible to calculate the mean delay $E(delay)$ as follows.

Define $\Phi_2(i,j), 1 \leq i \leq d$ and $1 \leq j \leq l$, as the expected delay suffered by an arbitrary packet provided that the first transmission of the packet coincided with the transmission of $i-1$ other packets and provided that the D-BMAP is in state j after the first transmission. Let $\Phi_2(i) = (\Phi_2(i,1), \ldots, \Phi_2(i,l))$ and $\Phi_2 = (\Phi_2(0), \ldots, \Phi_2(d))$. The column vector Φ_2^t obeys the following equation (this equation is obtained in a similar manner as Equation (11)).

$$\Phi_2^t = \mathbf{e} + \sum_{s=0}^{d} \left(M_s U_s \Phi_2^t + N_s U_s [\Phi_1^t + G_s \Phi_2^t] \right), \tag{13}$$

where M_s and N_s are the following $(d+1)l \times (d+1)l$ diagonal matrices

$$\begin{aligned} M_s &= diag(\mathbf{0}^t, a_1(s)\mathbf{e}^t, \ldots, a_d(s)\mathbf{e}^t), & (14) \\ N_s &= diag(\mathbf{0}^t, b_1(s)\mathbf{e}^t, \ldots, b_d(s)\mathbf{e}^t), & (15) \end{aligned}$$

with $a_i(s) = 0$ for $i \leq s$, $a_i(s) = (i-s)/i$ for $i > s$, $b_i(s) = 0$ for $i < s$, $b_i(s) = s/i$ for $i \geq s$, $\mathbf{0}^t$ a $1 \times l$ vector with all elements zero and $\mathbf{e}^t$ a $1 \times l$ vector with all elements equal to one. $a_i(s)$, resp. $b_i(s)$, represents the probability that our arbitrary packet selects the first, resp. second, group after a collision knowing that s of the colliding stations select the second group. Equation (13) can be solved as a set of linear equations or using an iterative method. The expected delay experienced by a packet $E(delay)$ is found as

$$E(delay) = \sum_{i=1}^{d} \sum_{j=1}^{l} \Upsilon(i,j) \Phi_2(i,j). \tag{16}$$

8.2 Other Performance Measures

Define $\Theta(k,i,j), k \geq 0, 0 \leq i \leq d$ and $1 \leq j \leq l$, as the probability that the highest current stack level held by a station equals k and that the auxiliary variable of the Markov chain (X_t^d, N_t^d) equals (i,j). Let $\Theta(k,j) = (\Theta(k,i,1),\ldots,\Theta(k,i,l))$ and $\Theta(k) = (\Theta(k,0),\ldots,\Theta(k,d))$. Recall that it is possible that a string $J \in \Omega_1$ starts with a sequence of zeros, see Note 1. Therefore, $\Theta(k) = \sum_{J \in L(k)} \pi(J)$ with $L(k) \subset \Omega_1$, is the collection of strings J with a length $m, m \geq k$, and with exactly $m-k$ leading zeros. Define R as $\sum_{i=0}^{d} R_i$, then due to Equation (10)

$$\Theta(k) = \pi(\emptyset)(I - R_0)^{-1} \qquad k = 0, \qquad (17)$$

$$\Theta(k) = \Theta(k-1)(R - R_0) = \pi(\emptyset)(I - R_0)^{-1}(R - R_0) \qquad k = 1, \qquad (18)$$

$$\Theta(k) = \Theta(k-1)R = \pi(\emptyset)(I - R_0)^{-1}(R - R_0)R^{k-1} \qquad k > 1. \qquad (19)$$

The inverse of $(I - R_0)$, i.e., $\sum_j R_0^j$, exists because $R = \sum_i R_i$, $R_i \geq 0$ for $0 \leq i \leq d$ and the inverse of $(I - R)$, i.e., $\sum_j R^j$, exists. Define $\Gamma(k,i,j), k \geq 0, 0 \leq i \leq d$ and $1 \leq j \leq l$, as the probability that the number of backlogged stations equals k and that the auxiliary variable of the Markov chain (X_t^d, N_t^d) equals (i,j). Let $\Gamma(k,j) = (\Gamma(k,i,1),\ldots,\Gamma(k,i,l))$ and $\Gamma(k) = (\Gamma(k,0),\ldots,\Gamma(k,d))$. Then, due to Equation (10)

$$\Gamma(k) = \pi(\emptyset)(I - R_0)^{-1} \qquad k = 0, \qquad (20)$$

$$\Gamma(k) = \sum_{i=1}^{\min(k,d)} \Gamma(k-i)R_i(I - R_0)^{-1} \qquad k > 0. \qquad (21)$$

Next, define $\Lambda(k), k > 0$, as the expected number of backlogged stations with a current stack level equal to k. The probability of having $i, i > 0$, stations with a current stack level equal to $k, k > 0$, is $\sum_{J \in T(k)} \pi(J)\mathbf{e}$, where the subset $T(k) \subset \Omega_1$ is the collection of strings J for which the k-th integer from the right equals i. Hence,

$$\Lambda(k) = \sum_{i=1}^{d} i\pi(\emptyset)(I - R)^{-1}R_i R^{k-1}\mathbf{e}. \qquad (22)$$

Define $E[r]$ as the expected number of transmissions required to transmit a packet successfully. $E[r]$ is significantly smaller than $E(delay)$ because an active station only transmits whenever its current stack level is equal to zero. Let $\pi(\emptyset)(I - R)^{-1} = (\alpha(0),\ldots,\alpha(d))$, where $\alpha(i), 0 \leq i \leq d$, is a $1 \times l$ vector. Then, $E[r]$ is found as the ratio of the expected number of transmissions in slot t and the expected number of successful transmissions in slot t

$$E[r] = \frac{\sum_{k=1}^{d} k\alpha(k)\mathbf{e}}{\alpha(1)\mathbf{e}}. \tag{23}$$

Finally, let p_e, resp. p_s, resp. p_c be the probability that a time slot is empty, resp. holds a successful transmission, resp. holds a collision. Then,

$$\begin{aligned} p_e &= \alpha(0)\mathbf{e}, && (24) \\ p_s &= \alpha(1)\mathbf{e}, && (25) \\ p_c &= \sum_{i=2}^{d} \alpha(i)\mathbf{e}. && (26) \end{aligned}$$

9 NUMERICAL RESULTS

For all the numerical examples presented in this section d is chosen sufficiently large (see Note 2 for the exact definition). For most examples $d = 10$ is more than sufficient.

9.1 Stability Results

To test whether the Markov chain (X_t^d, N_t^d) is stable we calculate the matrices $G_k, 0 \leq k \leq d$, and check whether they are stochastic. If all the row sums of G_k are between $1 - 10^{-9}$ and 1, we conclude that G_k is stochastic. If there is a row in G_k for which the row sum is below $1 - 10^{-4}$ we conclude that the matrix G_k is not stochastic. If the smallest row sum of G_k is between $1 - 10^{-4}$ and $1 - 10^{-9}$ we conclude that the stochastic nature of G_k is undetermined (i.e., the recurrence of the chain (X_t^d, N_t^d) is unclear). Notice that if (X_t^d, N_t^d) is transient we can use the value of the smallest row sum d_t as a heuristic measure of instability.

As with most of the iterative formulas used in the matrix analytical approach, the number of iterations required by formula (9) increases significantly when the Markov chain (X_t^d, N_t^d) is close to instability (whereas 10 to 100 iterations suffice for many stable and unstable Markov chains, the number of iterations can become as large as a few thousands when the chain is (very) close to the instability point). This limits the precision by which instability points are determined.

Next, we determine the instability point of a number of arrival processes that belong the the class of the D-BMAP processes for $p = 1/2$. In the remainder of the paper, the instability point is also referred to as the stability point as this is the point where the CTM protocol switches between being stable and unstable.

9.1.1 The poisson process

Flajolet, et al [4] have shown that the CTM protocol (with free access) is stable for $\lambda < 0.360177147$. We start by confirming this result using our analytical model as follows. Let $B_n = e^{-\lambda}\lambda^n/n!$, for $n \geq 0$. An overview of the results is presented in Table 1. The first column represents the load λ, the second indicates whether the chain (X_t^d, N_t^d) is stable or not (S = stable, U = unstable) and the last column represents the stability measure d_s or the instability measure d_t depending on whether the Markov chain was stable or not. According to Table 1 the Markov chain (X_t^d, N_t^d) becomes unstable for λ somewhere between 0.36015 and 0.3602. No attempts were made to determine the stability point with a higher precision.

9.1.2 The erlang process

We define the Erlang process as follows. The Erlang process has independent and identically distributed interarrival times that obey an Erlang distribution with parameters k and λ. Clearly, for $k = 1$ the Erlang process is reduced to the Poisson process. The Erlang process can be modeled as a D-BMAP in the following way. Let $\beta_n = e^{-\lambda}\lambda^n/n!, n \geq 0$ and let $B_n, n \geq 0$, be $k \times k$ matrices defined as

$$(B_n)_{i,j} = \beta_{nk+j-i} \qquad nk \geq j - i, \tag{27}$$

$$(B_n)_{i,j} = 0 \qquad nk < j - i. \tag{28}$$

The stability points for the Erlang processes with $k = 2, 3$ and 4 are shown in Table 2. Clearly, increasing the parameter k results in a higher stability point. This is not surprising as the Erlang distribution becomes more deterministic when increasing k. As a function of k, the growth of the stability point decreases as k increases (this seems logical as the variance of the Erlang distribution decreases linearly in k). For instance, the stability point of the Erlang process with $k = 15$ is still well below 0.375 ($d_t = 0.3342$). Therefore, the absolute difference between the stability point for the Erlang process with $k = 1$ and

$k = 15$ is less than 1.5 percent, while the variance of the interarrival times is 15 times as large for $k = 1$ as opposed to $k = 15$.

Table 1
Stability of CTM
under Poisson Input

$\lambda = load$	S/U	d_s/d_t
0.1	S	0.9745
0.3	S	0.5207
0.35	S	0.1215
0.355	S	0.0617
0.36	S	0.0023
0.3601	S	0.0010
0.36015	S	0.0003
0.3602	U	0.9991
0.3603	U	0.9951
0.3605	U	0.9872
0.361	U	0.9678
0.3625	U	0.9120
0.37	U	0.6791
0.4	U	0.2169

Table 2
Stability of CTM
under Erlang k Input

$\lambda/k = load$	k	S/U	d_s/d_t
0.3625	2	S	0.1035
0.365	2	S	0.0199
0.3655	2	S	0.0017
0.3656	2	U	0.9965
0.3658	2	U	0.9835
0.366	3	S	0.1203
0.367	3	S	0.0468
0.3675	3	S	0.0059
0.3676	3	U	0.9973
0.368	3	U	0.9646
0.3675	4	S	0.1313
0.368	4	S	0.0574
0.369	4	U	0.9384
0.37	4	U	0.8521

9.1.3 The markov modulated poisson process

We restrict ourselves to a specific subclass of the Markov modulated Poisson processes known as the interrupted Poisson processes (IPP). These processes are 2-state MMPPs with $\lambda_1 = 0$ and $\lambda_2 = \lambda$. Transitions between state 1 and 2 can occur at the end of each time slot according to a 2×2 transition matrix T. We restrict ourselves to this subclass of the MMPPs because the IPPs are the most bursty arrival processes within the family of the MMPPs. Also, many other MMPPs have results that are very similar to the Poisson process, for instance, the MMPP with $\lambda_1 = 2\lambda_2$ and $e = f = 30$ (see below for the definition) has its stability point in the interval $[0.359, 0.36]$. Table 3 shows the stability point, for the interrupted Poisson process with the following transitions matrix T

$$T = \begin{pmatrix} 1 - 1/e & 1/e \\ 1/f & 1 - 1/f \end{pmatrix} \qquad (29)$$

where $e = f = 300$. Table 4 summarizes the results for $e = 7f = 210$. Thus, the expected sojourn time in state one, resp. state two, is e, resp. f time slots. For the second example, we let $d = 25$ for d to be sufficiently large.

Tables 3 and 4 show that the interval [0.3466, 0.348] includes the stability point of both IPPs. Thus, although the second IPP is by far the most bursty (i.e., the distribution of the number of arrivals in a time slot has a higher variation) of the two, their stability point differs less than 0.14 percent. As for the influence of correlation, we found that the interval [0.348, 0.349] contains the stability point of the IPP with $e = f = 30$. Comparing this with the results in Table 3, we see that correlation slightly decreases the stability of the CTM protocol (in our example less than 0.24 percent).

Table 3
Stability of CTM
under Int. Poisson Input
for $e = f = 300$

$\lambda/2 = load$	S/U	d_s/d_t
0.325	S	0.0673
0.34	S	0.0222
0.345	S	0.0072
0.3466	S	0.0025
0.348	U	0.9965
0.35	U	0.9843
0.36	U	0.9279

Table 4
Stability of CTM
under Int. Poisson Input
for $e = 7f = 210$

$\lambda/8 = load$	S/U	d_s/d_t
0.34	S	0.0202
0.345	S	0.0056
0.346	S	0.0027
0.3466	S	0.0009
0.348	U	0.9952
0.35	U	0.9856
0.36	U	0.9449

9.1.4 The bulk arrival process

The bulk arrival process is defined as a discrete time arrival process characterized by an $1 \times n$ vector v and a length L. The arrival pattern of this process consists of a repetition of identical cycles. The first part of each cycle consists of a set of batches, characterized by v. For instance $v = [2, 3, 2]$ means that we first have a batch of size 2, in the next time slot we have a batch of size 3, followed by a batch of size 2. The second part of the cycle is a silent period with a geometrically distributed length with average L. The Bulk arrival process can be described by the following D-BMAP. Let $v = [v_1, \ldots, v_m]$ and let $B_n, n \geq 0$, be a set of $m + 1 \times m + 1$ matrices with

$$(B_{v_j})_{j,j+1} = 1, \tag{30}$$

$$(B_0)_{m+1,1} = 1/L, \tag{31}$$

$$(B_0)_{m+1,m+1} = 1 - 1/L, \tag{32}$$

with $1 \leq j \leq m$. The other components of the matrices B_n are equal to zero. The load of a Bulk arrival process equals $\sum_j v_j/(L + m)$.

Table 5
Stability of CTM
under Bulk Arrival Input

$v_1/(L+1)$	v_1	S/U	d_s/d_t
0.3448	2	S	0.0161
0.347826	2	S	0.0026
0.348432	2	U	0.9987
0.3509	2	U	0.9008
0.3428	3	S	0.0259
0.349040	3	S	0.0024
0.349854	3	U	0.9926
0.3529	3	U	0.8446
0.3478	4	S	0.0033
0.348432	4	S	0.0012
0.349040	4	U	0.9916
0.3509	4	U	0.9287

Table 6
Stability of CTM
under Bulk Arrival Input

$\sum v_i/(L+2)$	$\sum v_i$	S/U	d_s/d_t
0.3488	2+1	S	0.0046
0.349854	2+1	S	0.0005
0.350050	2+1	U	0.9969
0.3504	2+1	U	0.9803
0.3484	3+1	S	0.0017
0.348735	3+1	S	0.0006
0.349040	3+1	U	0.9953
0.3509	3+1	U	0.9330
0.3448	2+2	S	0.0090
0.346620	2+2	S	0.0026
0.347826	2+2	U	0.9838
0.3484	2+2	U	0.9631

Table 5 presents the results for $m = 1$ with $v = [2], [3]$ and $[4]$. Table 6 holds the results for $m = 2$ with $v = [2, 1], [3, 1]$ and $[2, 2]$. For each of these processes we gradually decrease L, i.e., increase the load, until the CTM protocol becomes unstable. Perhaps somewhat surprisingly, the $v = [2, 2]$ process is the first of the six processes to become unstable (load $\rho \in [0.346620,\ 0.347826]$), then the $v = [2]$ process, followed by either the $v = [4]$ or the $v = [3, 1]$ process (we did not attempt to distinguish these two processes), next the $v = [3]$ process and finally the $v = [2, 1]$ process. From these results it follows that it is not always the most bursty process that results in the lowest stability point.

9.1.5 Summary

The stability point of the CTM protocol under D-BMAP input depends upon the exact definition of the input process. For instance, the Poisson process, the Erlang processes, the Markov modulated Poisson processes and the bulk arrival processes all result in a different stability point. Moreover, it is often difficult to state *a priori* from the characteristics which of two input processes results in a higher stability point.

On the other hand, the stability point of a D-BMAP process is never far below the stability point of the Poisson process (in our examples: at most 1.4 percent). Thus, the CTM protocol seems to maintain its good stability characteristics under D-BMAP input traffic. Clearly, we can always define a D-BMAP with a load $0 \leq \rho \leq 1$ for which the CTM protocol is stable, for example a D-MAP. Also, although correlation in the input traffic reduces the stability point somewhat, it does not devastate the stability.

An interesting open problem related to this is whether there exists a load ρ_{min} such that the CTM protocol with free access (with $p = 1/2$) is stable under all D-BMAPs with a load $\rho < \rho_{min}$. During the numerical trials, we did not find a D-BMAP with a load smaller than $\rho = 0.34657 = \log(2)/2$ for which the CTM protocol became unstable. For instance, the $v = [2,2,2,2]$, $v = [2,2,2,2,2]$, $v = [5]$, $v = [10]$ Bulk arrival process and the IPP with $e = f = 3000$ turned out to be stable for a load of $\log(2)/2$. The value $\log(2)/2$ is no stranger to the CTM protocol because Flajolet and Jacquet [4] have shown that the expected length of a busy period initiated by a collision of n stations increases asymptotically as $2n/log(2)$ provided that no new arrivals occur. This result also indicates that the bulk arrival process $v = [n]$ with a load smaller than $\log(2)/2$ is unlikely to cause instability even for large values of n and L. The question raises whether it is at all possible to find a D-BMAP with a load smaller that $\log(2)/2$ than makes the CTM protocol with free access unstable.

9.2 Performance Results

In the previous subsection, we demonstrated that correlation and variation generally slightly decrease the stability point of the CTM protocol. In this subsection, the influence of correlation and variation on the mean delay, mean number of transmissions, etc. is shown to be much more profound. Consider the class of D-BMAPs that models the superposition of M discrete time independent and identically distributed (i.i.d.) interrupted Bernoulli processes (IBPs). A superposition of M i.i.d. IBPs is characterized by a set of 4 parameters (M, d, α, β), where $1/d$ is the probability that an IBP generates a packet when ON, $1/\alpha$ is the mean sojourn time in the ON state and $1/\beta$ is the mean sojourn time in the OFF state. Details on how to model this input process as an $(M+1)$-state D-BMAP can be found in [16], where these input processes are used to model the superposition of variable bit rate (VBR) sources.

The first four columns of Table 7 represent the parameters that characterize the D-BMAP, $E(delay)$ represents the mean delay, $E[r]$ the expected number of transmissions required to successfully transmit a packet, $E[B]$ the expected number of backlogged stations, $T[B]$ the 10^{-9}-quantile of the number of backlogged stations, $T[M]$ the 10^{-9}-quantile of the maximum current stack level held by a station and p_c is the probability that a time slot contains a collision. The load ρ in each of the scenarios in Table 7 is 0.25.

Table 7
Performance Measures of CTM under
a superposition of IBP traffic

M	d	$1/\alpha$	$1/\beta$	$E(delay)$	$E[r]$	$E[B]$	$T[B]$	$T[M]$	p_c
20	40	20	20	5.28	2.24	1.76	59	78	0.1349
20	40	100	100	6.49	2.32	2.04	87	109	0.1431
20	40	500	500	8.58	2.39	2.55	189	223	0.1496
10	20	20	20	5.78	2.27	1.88	66	85	0.1375
10	20	100	100	8.62	2.43	2.55	118	143	0.1532
10	20	500	500	16.04	2.56	4.37	320	372	0.1661
10	10	20/3	20	10.97	2.62	3.09	124	139	0.1705
10	10	100/3	100	32.46	2.96	8.37	401	412	0.2041
10	10	500/3	500	128.65	3.17	32.37	1638	1636	0.2239
5	10	20	20	6.75	2.31	2.11	78	96	0.1413
5	10	100	100	13.90	2.61	3.82	177	206	0.1702
5	10	500	500	40.30	2.84	10.37	579	658	0.1925
Poisson	-	-	-	4.79	2.20	1.65	51	71	0.1318

Table 7 shows that the influence of the correlation between the number of arrivals in two consecutive slots is less significant when the arrivals are less bursty (i.e., $M = 20, d = 40$) and the results are to some extend comparable to the Poisson results. On the other hand, when the input process is more bursty (i.e., $M = 10, d = 10$ and $M = 5, d = 10$) the impact of the correlation on these performance measures is much more profound and the performance results are no longer comparable to the Poisson conditions.

Note 3: We can also use the analytical model to study the impact of specific arrival patterns on the performance measures. For instance, what is the worst possible pattern when N packets are generated ? To study specific arrival patterns we can make use of the Bulk arrival process where v contains the arrival pattern and L is extremely large.

10 CONCLUSIONS

In this paper, a nearly exact analytical model to evaluate the stability and the performance of the Capetanakis-Tsybakov-Mikhailov (CTM) protocol with free access under D-BMAP input traffic was introduced. The model involves the construction of a discrete-time Quasi-Birth-Death Markov chain with a tree structure. To our best knowledge, it is the first time that such Markov chains are used to evaluate medium access control (MAC) protocols. The main idea behind this model can also be extended in a natural way to evaluate other MAC protocols with an underlying stack structure. For instance, we successfully extended the techniques used in this paper in order to evaluate other important protocol variations of the CTM protocol, i.e., the Q-ary CTM protocol and the CTM protocol that *saves doomed slots* [14, 4]. Details on these extensions and their results can be found in [17].

Using numerical examples, we demonstrated that the stability point of the CTM protocol depends upon the exact definition of the D-BMAP. Also, it is often difficult to state *a priori* which of two input processes results in a higher stability point. Still, the CTM protocol seems to remain its good stability characteristics when the Poisson input traffic is replaced by a D-BMAP. The highest degeneration of the stability point observed was 1.4 percent. We did not succeed in finding a D-BMAP with a load below log(2)/2 for which the CTM protocol was unstable. This raises the question whether it is at all possible to find such a D-BMAP.

On the other hand, we demonstrated that introducing correlation and variation highly affects the other performance measures (delay, number of backlogged stations, and so on) of the CTM protocol and the results no longer correspond with the ones obtained under the Poisson assumption.

References

[1] Capetanakis, J.I. Tree algorithms for packet broadcast channels. IEEE Trans. Inform. Theory, **1979**, *25* (5), 319–329.

[2] Massey, J.L. Multi-users communication networks. CISM Courses and Lectures No. 256, Ed. G. Longo, Springer Verlag Wien-New York, **1981**.

[3] Massey, J.L. Some new approaches to random-access communications. Performance, P.J. Courtois and G. Latouche (editors), **1987**.

[4] Flajolet, P.; Jacques, P. Analytic models for tree communication protocols. INRIA, Technical Report No 648, **1987**.

[5] Fayolle, G.; Flajolet, P.; Hofri, M.; Jacquet, P. Analysis of a stack algorithm for random multiple-access communication. IEEE Transactions on Information Theory, **1985**, *IT-21* (2), 244–254.

[6] Greenberg, A.G.; Flajolet, P.; Ladner, R. E. Estimating the multiplicity of conficts to speed their resolution in mulitple access channels. Journal of the Association for Computing Machinery, **1987**, *34* (2), 289–325.

[7] Seri, M.; Sidi, M. Splitting algorithms in channels with markovian capture. European Transactions on Telecommunications and related Technologies, **1994**, *5* (1), 19–26.

[8] Lin, Y-D. On IEEE 802.14 medium access control protocol. IEEE Communication Surveys, **1998**.

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

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

[11] 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.

[12] HE, Q. Classification of markov processes of matrix M/G/1 type with a tree structure and its applications to the MMAP[K]/G[K]/1 queue. Stochastic Models, **2000**, *16* (5).

[13] HE, Q. Classification of markov processes of M/G/1 type with a tree structure and its applications to queueing models. O.R Letters, **1999**, *26*, 67–80.

[14] Bertsekas, D.; Gallager, R. Data networks. Prentice-Hall, Inc., **1992**.

[15] Blondia, C. A discrete-time batch markovian arrival process as B-ISDN traffic model. Belgian Journal of Operations Research, Statistics and Computer Science, **1993**, *32* (3,4).

[16] Blondia, C.; Casals, O. Statistical multiplexing of VBR sources: A matrix-analytical approach. Performance Evaluation, **1992**, *16*, 5–20.

[17] Van Houdt, B.; Blondia, C. Throughput of Q-ary splitting algorithms for contention resolution in communication networks. Submitted to Adv. in Perf. Eval., **2000**.

A Calculating $\Upsilon(k, j)$

Basically, $\Upsilon(k, j), 1 \leq k \leq d$ and $1 \leq j \leq l$, equals the probability that the first transmission of a packet coincides with the transmission of $k-1$ other packets and that the state of the D-BMAP modeling the input traffic is j after this first transmission. Let $\delta_s = \pi(\emptyset)(I-R)^{-1}R_s, 0 \leq s \leq d$, and $\delta = \pi(\emptyset)(I-R)^{-1}$. Clearly, δ_s and δ are $1 \times l(d+1)$ vectors. Thus δ_s can be written as $\delta_s = (\delta_s(0), \ldots, \delta_s(d))$, where $\delta_s(i), 0 \leq i \leq d$, are $1 \times l$ vectors. Similarly, $\delta = (\delta(0), \ldots, \delta(d))$ and $\pi(\emptyset) = (\pi_0(\emptyset), \ldots, \pi_d(\emptyset))$.

Both equations presented below are a natural extension of the common method used in an M/G/1 type of Markov chain to calculate the steady state probabilities of the Markov chain at an arrival instant given the steady state probabilities at an arbitrary time instant [15] and by observing that the packets that are dropped (due to d) are dropped before their first transmission attempt (see Equations (1), (2) and (3)). Let $\Upsilon(k) = (\Upsilon(k, 1), \ldots, \Upsilon(k, l))^t$. For $k < d$, we get

$$\begin{aligned} \Upsilon(k) &= \frac{1}{p_s}\left[\sum_{i=0}^{1}\left(\pi_i(\emptyset)kB_k + \sum_{s=0}^{k}\delta_s(i)(k-s)B_{k-s}\right) + \right. \\ &\left. \sum_{i=2}^{d}\delta(i)\sum_{s=0}^{\min(i,k)} C_s^i p^s(1-p)^{i-s}(k-s)B_{k-s}\right], \end{aligned} \tag{33}$$

where $\beta B = \beta$ and $\beta \mathbf{e} = 1$. p_s was defined in Section 8. For $k = d$, we have

$$\begin{aligned} \Upsilon(d) &= \frac{1}{p_s}\left[\sum_{i=0}^{1}\left(\pi_i(\emptyset)d\sum_{j\geq d} B_j + \sum_{s=0}^{d}\delta_s(i)(d-s)\sum_{j\geq d} B_{j-s}\right) + \right. \\ &\left. \sum_{i=2}^{d}\delta(i)\sum_{s=0}^{\min(i,k)} C_s^i p^s(1-p)^{i-s}(d-s)\sum_{j\geq d} B_{j-s}\right]. \end{aligned} \tag{34}$$