A Markov Service Policy with Application to the Queueing Study of a DQDB Station *

Ioannis Stavrakakis[†] Sophia Tsakiridou[‡]
Department of Electrical Engineering and Computer Science
University of Vermont
Burlington, VT 05405
Tel: (802) 656-8506 FAX: (802) 656-0696

Abstract

In this paper, a queueing system is formulated, analyzed and applied for the study of the queueing behavior of a DQDB station. At first, an appropriate process is defined to capture the station's perception of the state of the distributed global queue (DQDB network) and a queueing model is formulated. In the sequel, a general queueing system is considered; the packet service time requirement is assumed to depend on the state of an underlying Markov chain at the time when the packet is forwarded to the head of the queue. Finally, the process which captures the state of the global distributed queue is approximated and the resulting queueing model for the DQDB station is related to the general queueing system and analyzed. Results for the queue occupancy distribution of the DQDB station are derived and compared with simulations.

*Research supported by the National Science Foundation under grant NCR-9011962.
[†]email : stavraka@uvm.edu
[‡]email : tsakirid@uvm.edu

I Introduction

The Distributed Queue Dual Bus (DQDB) medium access protocol has been adopted as a Metropolitan Area Network (MAN) Standard (IEEE 802.6), [1]. It is a simple protocol which can achieve full utilization of the channel capacity. The DQDB MAN is a high speed (over 100 Mbps) integrated services network designed to accommodate traffic generated by a variety of information sources, as well as provide for the interconnection of LANs. A detailed description of the Distributed Queue Dual Bus (DQDB) protocol may be found in [1]. A brief discussion on the protocol, as well as on various design and performance analysis issues, may be found in [2]-[8] and the references cited there. No bandwidth balancing mechanism or priorities are investigated in the DQDB application considered in this paper, [1, 3, 7].

In this paper a single server queueing system is developed, analyzed and applied for the evaluation of the queueing behavior of a tagged DQDB station. Although the DQDB application has motivated the study of this queueing system, its generality allows for potentially wide applicability. Due to the distributed nature of the DQDB protocol, the service provided to a tagged station depends on the station's perception of the state of the global distributed queue which models the entire network. An appropriate process, which is capable of describing the station's perception of the state of the global queue at time instants of interest is identified (balance process). The service discipline at the local queue of the tagged DQDB station is then described in terms of the balance process; the state of the process at the time at which a packet is forwarded to the head of the queue probabilistically determines the time at which the packet is serviced (transmitted).

Processes similar to the balance process, which capture the tagged station's perception of the state of the distributed queue, have been considered in DQDB station models developed in the past [2, 5, 6]. The delay analysis of the *single* buffer DQDB station model presented in [2] is facilitated by introducing the concept of the Virtual Request Counter (VRQ-CTR). The content of the VRQ-CTR at the time at which a packet arrives to the queue determines the packet waiting time in the queue. Thus, the balance process plays a role more or less similar to the VRQ-CTR process. An extension of the work in [2] is presented in [5], where fixed length message arrivals to a queue capable of holding a single message at a time are considered instead of single packet arrivals.

The general queueing system developed in this paper can be described in terms of a vacation model where a server vacation takes place whenever a packet is forwarded to the head of the queue. The duration of a vacation period is probabilistically determined by the state of an underlying Markov chain at the beginning of the vacation. Alternatively, the service policy of the system can be described in terms of the effective service time defined as the waiting time of a packet at the head of queue until its service completion (departure). The state of the underlying Markov chain at the time at which a packet is forwarded to the head of the queue determines its effective service time. Under the latter description of the service policy, the queueing system can be considered as an analog, in discrete time, of the continuous time system presented in [11], where effective service times are encountered instead of service times. Matrix analytic techniques [10] and renewal theory [19] are applied for the derivation of the queue occupancy distribution. The approximate model for the DQDB station and the balance process can be related to the general queueing system by relaxing the conditions regarding the Markovian structure of the underlying process by applying the same analysis approach.

The paper is organized as follows. In the next section a single server queueing model is formulated for the study of the DQDB station. In Section III the general queueing system is formulated and analyzed. In Section IV the queueing analysis of the DQDB station is carried out as follows. At first, the approximations necessary for the derivation of a tractable model of key processes are presented. Then, the application of the analysis of the general queueing system (Section III) is considered through the establishment of weaker sufficient conditions for the application of the developed analysis approach (Proposition 2). Numerical results for the queue occupancy distribution of the DQDB station are presented in Section V, followed by concluding remarks in Section VI.

II A queueing model for the DQDB station

In this section the distributed queueing system which models the behavior of the DQDB network is described in terms of centralized queues operating under service policies which reflect the DQDB protocol. The basic operation of the DQDB protocol is presented to the extent that is relevant to the queueing model for a DQDB station developed in this paper.

The DQDB network consists of two high speed unidirectional buses carrying information in opposite directions, Fig. 1. The network users (stations) are distributed along the buses and are capable of transmitting information to, or receiving information from, any network station. For the topology shown in Fig. 1, station i, $1 \leq i \leq M$, uses bus A to transmit information to its downstream stations $i+1, \ldots, M$ and receive information from its upstream stations $1, 2, \ldots, i-1$; it uses bus B to transmit information to its upstream stations $1, 2, \ldots, i-1$ and receive information from its downstream stations $i+1, \ldots, M$. Without loss of generality, bus A is defined as the forward bus; bus B is then defined as the reverse bus. Each bus is equipped with a terminal station, called the head of the bus, which continuously generates fixed size slots. These slots propagate downstream past each station before being discarded at the end of the bus. Since the media access protocol is identical for both network buses, packet (or segments of information) transmissions only along the forward bus (A) will be considered.

At any given time a DQDB station may be either busy or idle; it is busy if a packet is queued in its buffer (to be transmitted on the forward bus), and it is idle otherwise. A busy station must notify all upstream stations of its intent to transmit by registering a request. This is implemented by setting the request bit of the first slot passing by on the reverse bus, whose request bit is reset. In turn, an upstream station that detects a request at a given time must satisfy it by allowing an empty slot to pass by on the forward bus. Stations keep track of the pending requests which need to be satisfied before they are allowed to transmit, by using a request counter (REQ-CNTR). The REQ-CNTR of an idle station is incremented by one any time a set request bit is detected. It is decremented by one each time an empty slot is detected on the forward bus, since this slot will be used to satisfy a downstream request.

The passage of a station from idle to busy state triggers the following actions : (a) The value of the REQ-CNTR is downloaded into a count-down counter (CD-CNTR). (b) The REQ-CNTR is reset to zero. (c) A request is placed on the reverse bus as explained above. During the period over which a station is busy the following occur : (a) Its CD-CNTR is decremented by one for each empty slot detected on the forward bus. (b) Its REQ-CNTR is incremented by one any time a set request bit is detected; no decrement of this counter occurs during the busy period.

When the value of the CD-CNTR reaches zero, the station is permitted to transmit into the next empty slot on the forward bus. The zero value of the CD-CNTR indicates that requests for empty slots made by the downstream stations before the tagged station became busy have been satisfied. When the packet is delivered to the forward bus, the station enters the idle state. If another packet has been waiting in the buffer, the station enters the busy state instantaneously. Notice that the above procedure prevents any station from having more than one outstanding request at any given time, which could cause monopolization of the network by a few stations.

Clearly, the implementation of the distributed queueing protocol requires that both buses be continuously monitored by all stations. The latter is necessary in order for the request process (on the reverse bus), $\left\{r_j^i\right\}_{j\geq 0}$, and the empty-slot process (on the forward bus), $\left\{e_j^i\right\}_{j\geq 0}$, to be observed. These processes are defined as follows.

$$e_j^i = \begin{cases} 1 & \text{if the } j^{th} \text{ slot on the forward bus \underline{leaving} station } i \text{ is empty} \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

$$r_j^i = \begin{cases} 1 & \text{if the request bit of the } j^{th} \text{ slot on the reverse bus \underline{arriving} at station } i \text{ is set} \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

Time instant (or slot) j denotes the end of the j^{th} slot; all events of interest are assumed to occur at the end of slots. Station i will be assumed to be the tagged station and the superscript i will be dropped for simplicity. All processes associated with a station will be assumed to be associated with the tagged station. The request, $\{r_j\}_{j\geq 0}$, and the empty-slot , $\{e_j\}_{j\geq 0}$, processes determine the value of the REQ-CNTR of the tagged station, denoted by $\left\{(REQ\text{-}CNTR)_j\right\}_{j\geq 0}$.

The network stations share the available capacity according to the DQDB protocol described at the beginning of this section. As it is the case with any distributed queueing protocol, the involved stations need to be provided with some information regarding the state of the distributed global queue. Based on this information the stations can position their service request (packet) in a conceptual centralized queue. Due to the propagation delay, the state information provided to the stations may be outdated. The "perception" of the tagged DQDB station about the state of the distributed global queue is determined by the balance process $\{b_j\}_{j\geq 0}$, defined by

$$b_0 = 0 \ , \ b_j = [b_{j-1} + r_j - e_j]^+ \quad , \ j \geq 1, \tag{3}$$

where $[x]^+ = \max(x, 0)$. Notice that b_j is equal to the number of unsatisfied requests of the downstream stations at slot j (sum of the content of REQ-CNTR and CD-CNTR). Let $S_b = \{0, 1, 2, \ldots, N_b\}$ denote the state space of the balance process. Theoretically, N_b is infinite, but for a practical, stable system the maximum number of pending requests at a station at any time instant (N_b), is bounded with probability arbitrarily close to one. For the derivation of numerical results, N_b will be truncated to some value N_{max} according to some criterion, to reduce numerical complexity (Section V).

The time sequences defined below will be involved in the study of the tagged station.

$w\langle z\rangle$: It is defined to be identical to w with the indication that w can be determined (probabilistically) from z.

$\left\{t_k^d\right\}_{k\geq 1}$: It is defined to be the sequence of slots at which a packet departure occurs.

$\left\{t_k^{adv}\right\}_{k\geq 1}$: It is defined to be the sequence of slots at which a packet advances to the Head of the Queue (HoQ) of the station.

$\left\{\hat{t}_k\right\}_{k\geq 1}$: It is defined to be the sequence of slots at which a change occurs at the HoQ. A change at the HoQ occurs if a packet advances to or leaves from the HoQ; clearly, $\left\{\hat{t}_k\right\}_{k\geq 1} = \left\{t_k^d\right\}_{k\geq 1} \cup \left\{t_k^{adv}\right\}_{k\geq 1}$.

The following proposition is important for the formulation of the proposed queueing model for the tagged DQDB station.

<u>Proposition 1</u>

The states of processes $\left\{(REQ\text{-}CNTR)_j\right\}_{j\geq 0}$ and $\{b_j\}_{j\geq 0}$ are identical at the slots in which a change occurs at the HoQ of the station. That is,

$$(REQ\text{-}CNTR)_{\hat{t}} = b_{\hat{t}} \quad , \ \forall\, \hat{t} \in \left\{\hat{t}_k\right\}_{k\geq 1} \tag{4}$$

The proof may be found in the Appendix. □

Since $\left\{t_k^{adv}\right\}_{k\geq 1} \subset \left\{\hat{t}_k\right\}_{k\geq 1}$ the above proposition implies that whenever a packet advances to the HoQ, the value (state) of the balance process, defined in (3), is identical to the value of the REQ-CNTR which is downloaded to the CD-CNTR. Thus, the values of the balance process at these time slots determine the initial values of the CD-CNTR, as determined by the DQDB protocol. As a result, the balance process provides the information about the state of the distributed queue that is utilized by the tagged station of the DQDB network.

In this paper, the packet arrival process to the (tagged) DQDB station is assumed to be an independent and identically distributed (i.i.d.) discrete-time process defined on the slot boundaries; the station's queueing capacity is assumed to be infinite. Based on Proposition 1 and the DQDB protocol operation, the queueing behavior of the DQDB station can be modeled in terms of a single server queueing system. The packet service (transmission) time is deterministic and equal to one time slot. The service policy of the single (centralized) queue that captures exactly the availability of the server (network capacity) to the tagged station - as determined by the tagged station's perception about the state of the distributed global queue - is described below, in terms of server vacations or the effective packet service time.

According to the DQDB protocol the server will be unavailable (on vacation) for a period w_k, following the time instant at which the k^{th} packet advances to the HoQ (of the tagged station), t_k^{adv}. Over this period, the pending requests $(REQ\text{-}CNTR)_{t_k^{adv}}$ from the downstream stations will be satisfied and some service will be provided to the upstream users. w_k depends on $(REQ\text{-}CNTR)_{t_k^{adv}}$ and the evolution of the empty-slot process; that is, $w_k = w_k\langle (REQ\text{-}CNTR)_{t_k^{adv}}, \{e_j\}_{j=t_k^{adv}+1}^{\infty}\rangle$. In fact, the future evolution of $\{e_j\}_{j\geq 0}$ after t_k^{adv} and up to a stopping time - defined as the time of the $\left((REQ\text{-}CNTR)_{t_k^{adv}}+1\right)^{st}$ empty slot leaving the tagged station after t_k^{adv} - is more than sufficient, but these details are not important at this point. In view of Proposition 1, $w_k = \langle b_{t_k^{adv}}, \{e_j\}_{j=t_k^{adv}+1}^{\infty}\rangle$.

An alternative way to describe the service policy is based on the concept of the effective packet service time. Instead of assuming a vacation of length w_k, it may be assumed that the server remains at the queue but the packet service time is increased to the effective service time $s_k = w_k + 1$. Note that s_k depends on $b_{t_k^{adv}}$ and $\{e_j\}_{j=t_k^{adv}+1}^{\infty}$ and it is, in general, different for different packets. The determination of the sequence of effective service times $\{s_k\}_{k\geq 1}$ would require knowledge of the sequences $\{e_j\}_{j\geq 0}$ and $\{r_j\}_{j\geq 0}$ so that $b_{t_k^{adv}}$ and $\{e_j\}_{j=t_k^{adv}+1}^{\infty}$ be determined for any $t_k^{adv} \in \left\{t_k^{adv}\right\}_{k\geq 1}$. Such a requirement would result in a service policy which would determine an analytically intractable queueing system. In Section IV, processes $\{e_j\}_{j\geq 0}$ and $\{r_j\}_{j\geq 0}$ are approximated in a way that the evolution of the balance process $\{b_j\}_{j\geq 0}$ after a time instant $t_k^{adv} \in \left\{t_k^{adv}\right\}_{k\geq 1}$, depends only on its state $b_{t_k^{adv}}$. Furthermore, the effective service time of the k^{th} packet turns out to depend only on $b_{t_k^{adv}}$, that is $s_k = s_k\langle b_{t_k^{adv}}\rangle$, for all $k \geq 1$.

In the next section a general queueing system is formulated and studied, to be applied to the queueing analysis of the DQDB station in Section IV.

III Queueing analysis under a Markov service policy

In this section the general queueing system is formulated and the queue occupancy distribution at the packet departure instants is derived. Then, the queue occupancy distribution at an arbitrary time can be derived by using Markov renewal arguments. In this paper the queue is assumed to contain all packets present in the system, including the one under service.

Consider a discrete-time, single server, infinite capacity queueing system with i.i.d. packet (customer) arrivals and effective packet service time dependent on the state of an underlying Markov chain $\{c_j\}_{j\geq 0}$ at the time when the packet advances to the HoQ; $\{c_j\}_{j\geq 0}$ is assumed to be independent of the packet arrival process. Let $S_c = \{0, 1, 2, \ldots, N_c\}$ denote the state space and $p_c(i,j)$, $\pi_c(i)$, $i,j \in S_c$, the transition and steady state probabilities, respectively, of the underlying Markov chain $\{c_j\}_{j\geq 0}$. Given the state i of $\{c_j\}_{j\geq 0}$ when a packet advances to the HoQ, its effective service time $s(i)$ depends only on i and it is given by a general distribution $p_{s_c}(i,k)$, $i \in S_c$, $k \geq 1$. Consider the following definitions in addition to those presented in Section II:

x_j : Denotes the queue occupancy at slot j.

s_k : Denotes the effective service time of the k^{th} packet, defined as the length of time that the packet stays in the HoQ; $s_k = t_k^d - t_k^{adv}$. According to the packet service policy, s_k depends on $c_{t_k^{adv}}$, that is, $s_k = s_k\langle c_{t_k^{adv}}\rangle$.

a_j : Denotes the number of packet arrivals at slot j.

a^l : Denotes the generic random variable for the number of packet arrivals over l slots, $l \geq 1$; let $p_{a^l}(n)$ denote the probability that n packets arrive over l slots, $n \geq 0$; $p_{a^l}(n)$ can be determined as the l-fold convolution of $p_{a^1}(n)$.

τ : Denotes the generic random variable for the number of slots until the first packet arrival to the queue that was left empty by the last packet departure; its probability distribution is easily seen to be given by $p_\tau(n) = [p_{a^1}(0)]^{n-1}\,[1 - p_{a^1}(0)]$, $n \geq 1$.

τ_k : Denotes the length of time that the HoQ remains empty after the $(k-1)^{st}$ packet departure. That is, $\tau_k = t_k^{adv} - t_{k-1}^d$ (see Fig. 2). Note that $\tau_k = 0$ if $t_k^{adv} \equiv t_{k-1}^d$ which occurs when the departure of the $(k-1)^{st}$ packet leaves the queue non-empty ($x_{t_{k-1}^d} > 0$); in this case the k^{th} packet moves to the HoQ instantaneously. Otherwise, that is, if $x_{t_{k-1}^d} = 0$, τ_k is distributed identically to τ. The conditional probability distribution of τ_k given the buffer occupancy state, is given by $p_{\tau/x}(\tau_k = 0/x_{t_{k-1}^d} > 0) = 1$ and

$p_{\tau/x}(\tau_k = n/x_{t^d_{k-1}} = 0) = p_\tau(n),\ n \geq 1$. Note that $\tau_k = \tau_k\langle x_{t^d_{k-1}}\rangle$.

d_k : Denotes the (interdeparture) time between the $(k-1)^{st}$ and the k^{th} packet departures; $d_k = t^d_k - t^d_{k-1} = \tau_k + s_k$.

To derive the queue occupancy distribution upon packet departure, $\left\{x_{t^d_k}\right\}_{k\geq 1}$, an appropriate Markov chain is embedded on the time instants of packet departures, $\left\{t^d_k\right\}_{k\geq 1}$. Note that $\left\{x_{t^d_k}\right\}_{k\geq 1}$ evolves as follows

$$x_{t^d_k} = x_{t^d_{k-1}} + a^{d_k} - 1 \tag{5}$$

where a^{d_k} denotes the number of packet arrivals over d_k slots. From the definition of d_k, τ_k and s_k it is established that

$$d_k = \tau_k + s_k = \tau_k\langle x_{t^d_{k-1}}\rangle + s_k\langle c_{t^{adv}_k}\rangle \tag{6}$$

Since $t^{adv}_k = \tau_k\langle x_{t^d_{k-1}}\rangle + t^d_{k-1}$ it turns out that

$$c_{t^{adv}_k} = c_{t^{adv}_k}\langle c_{t^d_{k-1}}, x_{t^d_{k-1}}\rangle \tag{7}$$

From the above it is established that

$$x_{t^d_k} = x_{t^d_k}\langle x_{t^d_{k-1}}, c_{t^d_{k-1}}\rangle \tag{8}$$

Since $c_{t^d_k} = c_{t^d_k}\langle c_{t^d_{k-1}}, d_k\rangle$, the above imply that

$$c_{t^d_k} = c_{t^d_k}\langle c_{t^d_{k-1}}, x_{t^d_{k-1}}\rangle \tag{9}$$

From (8) and (9) it is established that $\left\{x_{t^d_k}, c_{t^d_k}\right\}_{k\geq 1}$ is a 2-D Markov chain embedded on the packet departure time instants $\left\{t^d_k\right\}_{k\geq 1}$. Let $S_{xc} = Z^+_0 \times S_c$ denote its state space, where Z^+_0 denotes the set of nonnegative integers; let $q_{xc}(i,m)$, $p_{xc}(i_1, m_1, i_2, m_2)$ and $\tilde{P}_{xc}$ denote its steady state probabilities, transition probabilities and transition matrix, respectively. By identifying $x_{t^d_k}$ and $c_{t^d_k}$ as the level and phase of Markov chain $\left\{x_{t^d_k}, c_{t^d_k}\right\}_{k\geq 1}$, respectively, its transition matrix $\tilde{P}_{xc}$ may be written in the form

$$\tilde{P}_{xc} = \begin{bmatrix} \tilde{B}_0 & \tilde{B}_1 & \tilde{B}_2 & \tilde{B}_3 & \cdots \\ \tilde{A}_0 & \tilde{A}_1 & \tilde{A}_2 & \tilde{A}_3 & \cdots \\ 0 & \tilde{A}_0 & \tilde{A}_1 & \tilde{A}_2 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}. \tag{10}$$

$\tilde{A}_n$ and $\tilde{B}_n$, $n \geq 0$, are $(N_c+1) \times (N_c+1)$ matrices with elements $A_n(m_1, m_2)$ and $B_n(m_1, m_2)$, $m_1, m_2 \in S_c$, given by

$$A_n(m_1, m_2) = \Pr\left\{x_{t_k^d} = i_1 - 1 + n, c_{t_k^d} = m_2 / x_{t_{k-1}^d} = i_1, c_{t_{k-1}^d} = m_1\right\} \ , \ i_1 \geq 1 \tag{11}$$

$$B_n(m_1, m_2) = \Pr\left\{x_{t_k^d} = n, c_{t_k^d} = m_2 / x_{t_{k-1}^d} = 0, c_{t_{k-1}^d} = m_1\right\} \tag{12}$$

Note that $B_n(m_1, m_2)$, $m_1, m_2 \in S_c$, denotes the probability that the phase process $\left\{c_{t_k^d}\right\}_{k \geq 1}$ moves from state m_1 to state m_2 on two consecutive departure instants leaving the level process $\left\{x_{t_k^d}\right\}_{k \geq 1}$ in state n, under the assumption that the level process is in the zero state at the beginning of the transition. Similarly for $A_n(m_1, m_2)$ with the difference that n packet arrivals occur during this transition, under the assumption that the level process is in a non-zero state at the beginning of the transition. Let $\tilde{A}_n(l)$ $(\tilde{B}_n(l))$ be defined similarly to $\tilde{A}_n$ $(\tilde{B}_n)$ but imposing the additional requirement that the interdeparture time $d_k = \tau_k + s_k$ is equal to l, $l \geq 1$. That is, for $m_1, m_2 \in S_c$ and any $l \geq 1$,

$$A_n(m_1, m_2, l) = \Pr\left\{x_{t_k^d} = i_1 - 1 + n, c_{t_k^d} = m_2, s_k = l / x_{t_{k-1}^d} = i_1, c_{t_{k-1}^d} = m_1\right\}, \ i_1 \geq 1 \tag{13}$$

$$B_n(m_1, m_2, l) = \Pr\left\{x_{t_k^d} = n, c_{t_k^d} = m_2, \tau_k + s_k = l / x_{t_{k-1}^d} = 0, c_{t_{k-1}^d} = m_1\right\} \tag{14}$$

To facilitate the application of the formulae that follow to the DQDB study in the next section, let $\tilde{P}_{c/>0}$ $(\tilde{P}_{c/0})$ denote the one-step transition probability matrix of $\{c_j\}_{j \geq 0}$ over the periods when the queue is non-empty (empty); let $\tilde{P}_{c/>0}(l)$ and $\tilde{P}_{c/0}(l)$ denote the corresponding l-step transition probability matrices. Since $\{c_j\}_{j \geq 0}$ is independent from anything associated with the queue, $\tilde{P}_{c/>0}(l) = \tilde{P}_{c/0}(l) = [\tilde{P}_c]^l$, that is, it is equal to the l-step transition probability matrix of $\{c_j\}_{j\geq 0}$. Let $\tilde{P}_{s_c}(l) = \mathrm{diag}\,[p_{s_c}(0,l), p_{s_c}(1,l), \ldots, p_{s_c}(N_c,l)]$ denote the diagonal matrix of the effective packet service time probabilities associated with the states of $\{c_j\}_{j\geq 0}$. From (13) and (14) it is easily established that matrices $\tilde{A}_n(l)$ and $\tilde{B}_n(l)$ are given by

$$\tilde{A}_n(l) = p_{a^l}(n)\tilde{P}_{s_c}(l)\tilde{P}_{c/>0}(l) \quad , \quad l \geq 1 \tag{15}$$

$$\tilde{B}_n(l) = \sum_{i=1}^{l-1} [p_{a^1}(0)]^{i-1} \sum_{j=0}^{n} p_{a^1}(j+1)\tilde{P}_{c/0}(i)\tilde{A}_{n-j}(l-i) \quad , \quad l \geq 2 \quad , \quad \tilde{B}_n(1) = \tilde{0} \tag{16}$$

where $\tilde{0}$ is the $(N_c+1) \times (N_c+1)$ zero matrix. Finally, by using total probability arguments, $\tilde{A}_n$ and $\tilde{B}_n$ can be written as follows

$$\tilde{A}_n = \sum_{l=1}^{\infty} \tilde{A}_n(l) = \sum_{l=1}^{\infty} p_{a^l}(n)\tilde{P}_{s_c}(l)\tilde{P}_{c/>0}(l) \tag{17}$$

$$\tilde{B}_n = \sum_{l=1}^{\infty} \tilde{B}_n(l) = \tilde{P}_{c/0}\left(\tilde{I} - p_{a^1}(0)\tilde{P}_{c/0}\right)^{-1} \sum_{j=0}^{n} p_{a^1}(j+1)\tilde{A}_{n-j} \tag{18}$$

where $\tilde{I}$ is the $(N_c+1) \times (N_c+1)$ identity matrix.

The stationary probability distribution of the queue occupancy process upon departures, $\{q_x(i)\}_{i=0}^{\infty}$, - as determined by the stochastic matrix $\tilde{P}_{xc}$ - is derived based on the matrix analytic approach, as outlined in [9]; details on this approach may be found in [10]-[16]. The stationary probability distribution of the queue occupancy process at arbitrary time instants, $\{\psi(i)\}_{i=0}^{\infty}$, is then derived in terms of $\{q_x(i)\}_{i=0}^{\infty}$ by invoking Markov renewal theory arguments [9]; it is given by

$$\psi(0) = \lambda_t \left(1 - p_{a^1}(0)\right)^{-1} q_x(0) \tag{19}$$

$$\psi(i) = \left(1 - p_{a^1}(0)\right)^{-1} \left\{ \sum_{j=1}^{i} p_{a^1}(j)\psi(i-j) + \lambda_t \left(q_x(i) - q_x(i-1)\right) \right\} \quad , \quad i \geq 1, \tag{20}$$

where λ_t denotes the packet arrival rate.

IV The DQDB station queueing analysis

In this section the general queueing system studied in Section III is considered for the study of the queueing behavior of the DQDB station. This section is organized as follows. At first approximations on $\{e_j\}_{j\geq 0}$ and $\{r_j\}_{j\geq 0}$ are presented. Then the relevance of the queueing model of Section III to the DQDB station queueing behavior is discussed and the identified application peculiarities are addressed.

As it was shown in Section II, the state of the balance process $\{b_j\}_{j\geq 0}$ (defined by (3)) when the k^{th} packet advances to the HoQ affects its effective service time. Equation (3) may be written as follows:

$$b_0 = 0, \qquad b_j = [b_{j-1} + r_j - \hat{e}_j + 1_j^d]^+ \tag{21}$$

where $1_j^d = 1$ if a (tagged) packet departure occurs at time j and it is zero otherwise; $\{\hat{e}_j\}_{j\geq 0}$ denotes the empty-slot arriving process defined by

$$\hat{e}_j = \begin{cases} 1 & \text{if the } j^{th} \text{ slot on the forward bus } \underline{\text{arriving}} \text{ at the tagged station is empty} \\ 0 & \text{otherwise.} \end{cases} \tag{22}$$

Clearly, $\hat{e}_j = e_j + 1_j^d$, where $\{e_j\}_{j\geq 0}$ is the empty slot process leaving the tagged station (Section II). Processes $\{\hat{e}_j\}_{j\geq 0}$ and $\{r_j\}_{j\geq 0}$ are complex, since they reflect the cumulative activities of the upstream and the downstream stations, respectively, as modulated by the DQDB protocol and the stations' perception about the state of the system. An exact characterization of these processes is difficult. To develop a tractable model for the balance process $\{b_j\}_{j\geq 0}$ and apply the analysis of the general queueing system (Section III), processes $\{\hat{e}_j\}_{j\geq 0}$ and $\{r_j\}_{j\geq 0}$ are assumed to be Bernoulli with intensities $p_{\hat{e}}$ and p_r equal to $1 - \lambda_{up}$ and λ_{down}, respectively; λ_{up} (λ_{down}) denotes the packet traffic rate generated by the upstream (downstream) stations.

The impact of the Bernoulli traffic approximation on the queue occupancy distribution is not significant, as indicated by the matching of the derived analytical results with simulations. This matching could be attributed to the 'smoothening' effect of the queueing process and, specifically for this queueing system, to the fact that the cumulative number of events (requests or empty slots) over some time horizon determine the evolution of the queue occupancy and not, in general, isolated events. The latter constitutes an additional filtering of the Bernoulli approximation, for the specific queueing system under study. Finally, it should be noted that a correlated (2-state Markov) model for the request and busy slot processes can be easily accommodated by the developed model at the expense of increasing the computational complexity. Such a model has not been considered here, though, for several reasons: (a) the accuracy achieved with the Bernoulli model was considered to be satisfactory; (b) an approach for the determination of the parameters of the 2-state Markov approximation would have to be developed, which is beyond the scope of this paper, particularly in view of (a); (c) in view of (a) and the model's applicability under correlated busy slot traffic, such a consideration would not increase substantially the contribution of this work.

The following proposition will be useful for the study of the DQDB station.

<u>Proposition 2</u>:

Under the Bernoulli assumption for $\{\hat{e}_j\}_{j\geq 0}$ and $\{r_j\}_{j\geq 0}$ the balance process in (21) is not Markov, but satisfies the following:

(a) $b_{t_k^{adv}}$ is (probabilistically) determined by $\{x_{t_{k-1}^d}, b_{t_{k-1}^d}\}$, i.e., $b_{t_k^{adv}} = b_{t_k^{adv}}\langle b_{t_{k-1}^d}, x_{t_{k-1}^d}\rangle$,

(b) s_k is (probabilistically) determined by $b_{t_k^{adv}}$, i.e., $s_k = s_k\langle b_{t_k^{adv}}\rangle$,

(c) $b_{t_k^d}$ is (probabilistically) determined by $b_{t_k^{adv}}$, i.e., $b_{t_k^d} = b_{t_k^d}\langle b_{t_k^{adv}}\rangle$.

Conditions (a)-(c) are sufficient for $\left\{x_{t_k^d}, b_{t_k^d}\right\}_{k\geq 1}$ to be an embedded Markov chain on $\left\{t_k^d\right\}_{k\geq 1}$.

<u>Proof</u>:

(a) If $x_{t^d_{k-1}} > 0$ then $t^{adv}_k \equiv t^d_{k-1}$ and, thus, $b_{t^{adv}_k} = b_{t^d_{k-1}}$. If $x_{t^d_{k-1}} = 0$ then $(t^d_{k-1}, t^{adv}_k] \neq \phi$. From (21) it is easily established that the evolution of the balance process over $(t^d_{k-1}, t^{adv}_k]$ is described by

$$b_j = [b_{j-1} + r_j - \hat{e}_j]^+ \,,\ j \in (t^d_{k-1}, t^{adv}_k] \,,\ k \geq 1 \tag{23}$$

where $t^d_0 \triangleq 0$. Under the Bernoulli assumption for $\{r_j\}_{j\geq 0}$ and $\{\hat{e}_j\}_{j\geq 0}$, (23) implies that $\{b_j\}_{j\geq 0}$ evolves over the non-zero intervals $(t^d_{k-1}, t^{adv}_k]$, $k \geq 1$, according to the Markov transition matrix $\tilde{P}_b$ given by (for $N_b = \infty$)

$$\tilde{P}_b = \begin{bmatrix} p_0 + p_1 & p_2 & 0 & 0 & \cdots \\ p_0 & p_1 & p_2 & 0 & \cdots \\ 0 & p_0 & p_1 & p_2 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{24}$$

where $p_0 = p_{\hat{e}}(1 - p_r)$, $p_1 = p_{\hat{e}} p_r + (1 - p_{\hat{e}})(1 - p_r)$ and $p_2 = (1 - p_{\hat{e}}) p_r$.

In view of the above, $\{b_j\}_{j\geq 0}$ evolves as determined by (24) after t^d_{k-1}, $k \geq 1$, for a time interval $\tau_k = t^{adv}_k - t^d_{k-1}$ which depends on $x_{t^d_{k-1}}$. Thus, $b_{t^{adv}_k}$ is determined as the τ_k - step transition of the Markov matrix (24), where the non-zero τ_k (when $x_{t^d_{k-1}} = 0$) is equal to the time until the first packet arrival following t^d_{k-1}.

(b) Let $T_{t^{adv}_k}(m)$ be defined to be the time interval until the occurrence of the stopping time of process $\{\hat{e}_j\}_{j\geq 0}$, defined as the time instant at which the m^{th} empty slot arrives at the tagged station following time instant t^{adv}_k. Clearly,

$$\sum_{i=1}^{T_{t^{adv}_k}(m+1)} \hat{e}_{t^{adv}_k + i} = m + 1 \tag{25}$$

From the definition of the effective packet service time, s_b, and the DQDB protocol it is easily established that if $b_{t_k^{adv}} = m,\ m \in S_b$, then

$$s_b(m) = T_{t_k^{adv}}(m+1) \tag{26}$$

and thus, under the Bernoulli assumption for $\{\hat{e}_j\}_{j\geq 0}$,

$$p_{s_b}(m,l) = \Pr\left\{T_{t_k^{adv}}(m+1) = l\right\} = \binom{l-1}{m} p_{\hat{e}}^m (1-p_{\hat{e}})^{l-1-m} p_{\hat{e}}\ ,\ l > m,\ m \in S_b \tag{27}$$

The above prove part (b) .

(c) It was shown in (b) that $s_k = s_k\langle b_{t_k^{adv}}\rangle$. Given that $s_k(b_{t_k^{adv}}) = l,\ l \geq 1$, $b_{t_k^d}$ is determined completely by the evolution of $\{r_j\}_{j\geq 0}$ over l slots. This is easily explained in terms of the DQDB protocol description. The content of the CD-CNTR decreases from $b_{t_k^{adv}}$ to zero during the effective service time interval $(t_k^{adv}, t_k^d]$ of length l and the state $b_{t_k^d}$ of the balance process at t_k^d reflects the new requests registered over the same interval. In view of this observation, the l-step probabilities of $\{b_j\}_{j\geq 0}$ over a busy sub-period $(t_k^{adv}, t_k^d]$, where $t_k^d - t_k^{adv} = s_k(b_{t_k^{adv}}) = l$, are easily determined in terms of the parameters of $\{r_j\}_{j\geq 0}$:

$$p_{b/>0}(m_1, m_2, l) = \binom{l}{m_2} p_r^{m_2}(1-p_r)^{l-m_2}\ ,\ m_1, m_2 \in S_b,\ l \geq m_2 \tag{28}$$

It is concluded from the above that $b_{t_k^d} = b_{t_k^d}\langle b_{t_k^{adv}}\rangle$. Note that given the length of the effective service time l, $b_{t_k^d}$ is independent of $b_{t_k^{adv}}$. This is also illustrated in (28) where the right hand side is independent from m_1.

In view of (a), (b) and (21), arguments similar to those in (5)-(9) determine that $x_{t_k^d} = x_{t_k^d}\langle x_{t_{k-1}^d}, b_{t_{k-1}^d}\rangle$, under the Bernoulli assumption for $\{r_j\}_{j\geq 0}$ and $\{\hat{e}_j\}_{j\geq 0}$. It is also clear, from (a) and (c), that $b_{t_k^d} = b_{t_k^d}\langle x_{t_{k-1}^d}, b_{t_{k-1}^d}\rangle$. Therefore, $\left\{x_{t_k^d}, b_{t_k^d}\right\}_{k\geq 1}$ is a Markov chain embedded on $\left\{t_k^d\right\}_{k\geq 1}$ and the proof of Proposition 2 is completed.

A careful consideration of the study of the general queueing system presented in Section

III reveals that the analysis is valid for a similar queueing system whose underlying process $\{b_j\}_{j\geq 0}$ satisfies (a) and (c) and whose effective packet service time satisfies (b), in Proposition 2. Equations (24), (27) and (28) determine the elements of matrices $\tilde{P}_{b/0}$, $\tilde{P}_{s_b}(l)$ and $\tilde{P}_{b/>0}(l)$ which - together with $\tilde{P}_{b/0}(l) = [\tilde{P}_b]^l$ - replace the corresponding matrices for process $\{c_j\}_{j\geq 0}$ in (15)-(18) and, finally, determine transition matrix $\tilde{P}_{xb}$ (as in (10)) for the DQDB station model. Due to the independence of $b_{t_k^d}$ from $b_{t_k^{adv}}$ given l, $\tilde{A}_n(l)$ can be derived as in (15) by replacing the $(N_b+1)\times(N_b+1)$-dimensional matrices $\tilde{P}_{s_b}(l)$ and $\tilde{P}_{b/>0}(l)$ by the (N_b+1)-dimensional vectors $\bar{P}_{s_b}(l)$ and $\bar{P}_{b/>0}(l)$ defined below.

$$\bar{P}_{s_b}(l) = [p_{s_b}(0,l),\ldots,p_{s_b}(N_b,l)]^T \text{ and } \bar{P}_{b/>0}(l) = [p_{b/>0}(0,l),\ldots,p_{b/>0}(N_b,l)] \qquad (29)$$

Note that the conditions in Proposition 2 are fully satisfied by an underlying process, $\{u_j\}_{j\geq 0}$, which evolves according to different stochastic matrices over the idle ($\tilde{P}_{u/0}$) and busy ($\tilde{P}_{u/>0}$) periods and satisfies condition (b). In this case, $u_{t_k^{adv}}$ is determined according to the elements of the τ_k-th power of the first transition matrix and $u_{t_k^d}$ according to the elements of the s_k-th power of the second matrix. The balance process is different from the above in the sense that it does not evolve according to a stochastic matrix over the busy periods, (21).

V Numerical results and discussion

Results for the queue occupancy distribution (at an arbitrary time instant) of DQDB stations are presented in this section, under various values of network parameters. They are calculated based on the queue occupancy distribution upon packet departures, which is derived as presented in this paper.

The traffic load of a station is assumed to be linearly dependent on its position on the bus (number of downstream stations). That is, the traffic load of station i is given by

$$\lambda_i = (M - i)\lambda_0 \ , \ \ \lambda_0 = \frac{2\lambda}{M(M-1)} \tag{30}$$

where M denotes the number of stations in the DQDB network and λ is equal to the total traffic load on the bus. Various models have been considered for the packet arrival process to the DQDB station: Bernoulli, (truncated) Poisson, independent and identically distributed with fixed size batches of lengths 3 (Batch-3) and 4 (Batch-4). Uniform traffic load has been considered only for the derivation of the results shown in Fig. 11.

The computational tractability of the analysis approach presented in Section III requires that the cardinality $N_b + 1$ of the state space S_b of $\{b_j\}_{j\geq 0}$ be finite and of small to moderate size. This is necessary for the solution of non-linear matrix equations required by the matrix-analytic approach [9]. Regarding the DQDB application, the latter requirement imposes a restriction on the maximum number of pending requests N_b at any time instant, as captured by the state of the balance process $\{b_j\}_{j\geq 0}$. The larger the network size in number of stations and physical length, the larger the maximum number of pending requests. A maximum value N_{max} on the number of pending requests has been considered for the derivation of the numerical results. In particular, the evolution of the balance process over the idle periods $\{(t^d_{k-1}, t^{adv}_k]\}_{k\geq 1}$, given by (23), is truncated and determined by

$$b_j = \min\left\{[b_{j-1} + r_j - \hat{e}_j]^+ \ , \ N_{max}\right\}, \ j \in (t^d_{k-1}, t^{adv}_k] \ , \ k \geq 1. \tag{31}$$

Thus, the transitions of the balance process over the idle periods are determined by the $(N_{max} +$

$1) \times (N_{max}+1)$ stochastic matrix

$$\tilde{P}_{b,\max} = \begin{bmatrix} p_0+p_1 & p_2 & 0 & \dots & 0 & 0 \\ p_0 & p_1 & p_2 & \dots & 0 & 0 \\ 0 & p_0 & p_1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & p_0 & p_1+p_2 \end{bmatrix}. \tag{32}$$

Note that the stationary probabilities of (32) and (24) are given, by

$$\pi_{b,max}(m) = \frac{1-p}{1-p^{N_{max}+1}} p^m,\ m \in S_{max} = \{0,1,\dots,N_{max}\} \tag{33}$$

$$\pi_b(m) = (1-p)p^m,\ m \in S_b = \{0,1,2,\dots,N_b\},\ N_b = \infty, \tag{34}$$

respectively, where $p = p_2/p_0$. For sufficiently large N_{max}, $p^{N_{max}+1} < \delta$ and $\pi_{b,max}(m) - \pi_b(m) < \epsilon(\delta, m)$ for $m \in S_{max}$. For $\delta = 10^{-7}$ it was observed that an increment of N_{max} beyond the minimum value specified by this value of δ, did not result in any significant change of the values of the computed queue occupancy distributions. This value of N_{max} was also used for the truncation of the state space containing the allowed transitions of $\{b_j\}_{j\geq 0}$ over the busy sub-periods $(t_k^{adv}, t_k^d]$, $k \geq 1$, determined by (28). Notice that the value of N_{max} depends on the location of the tagged station. Some plots of N_{max} as a function of the station location n are shown in Fig. 3, for $\delta = 10^{-7}$, various network sizes M and total traffic loads λ (assumed to be linearly distributed among the stations). N_{max} assumes the largest values for stations around the middle of the network, reflecting the expected large number of pending requests in these stations. The smaller number of pending requests for more downstream stations reflects the dominating impact of lighter request traffic. The smaller number of pending requests for more upstream stations reflects the dominating impact of lighter busy slot traffic. Finally, N_{max} increases with λ and M as expected.

Figs. 4, 5, 6 present the results obtained for stations No. 2, No. 5, and No 9, respectively, for $M = 10$, $\lambda = .7$ and packet arrival processes modeled as Bernoulli, Poisson and Batch-4. Simulation results are also presented for a DQDB network with identical parameters and interstation distance equal to one time slot. The analytically derived results are seen to be in good agreement with the simulations. Note that the results obtained for Station No. 9 under Bernoulli and Poisson arrivals almost coincide due to the fact that these processes almost coincide for small rates, which is the case for λ_9 in view of (30).

Fig. 7 (Fig. 8) presents the results for station No. 3 obtained for $M = 5$ and Bernoulli (Poisson) arrivals under traffic loads $\lambda = .7$, .8 and .9. As expected, the queue occupancy increases as λ increases. Both analytical and simulation results are shown.

Fig. 9 presents the analytical results obtained for station No. 2 obtained for $M = 5$, $\lambda = .7$ and under Bernoulli, Poisson, Batch-3 and Batch-4 packet arrival processes. Note that the queue occupancy increases with the batch size. For instance at the queue occupancy level with probability 10^{-7} the queue occupancy is below 15 under Bernoulli arrivals and over 30 under Batch-3 arrivals.

Fig. 10 presents the analytical results obtained for stations No. 2, 5 and 9 for $M = 5$, $\lambda = .7$ and under Poisson and Batch-4 packet arrivals. Notice that (for the traffic load distribution as in (30)) the larger the station number (more downstream) the smaller the queue occupancy and, thus, the smaller the required buffer size for certain packet loss probability. It should be noted that the previous does not reflect a 'reverse' unfairness, in the sense that the downstream users are favored by the protocol, but it is solely a consequence of the lighter traffic of the downstream stations under (30); thus, these stations are expected to require smaller buffers. The 'classical' unfairness problem of the DQDB protocol is, in fact, observed through the occupancy distribution as well, when uniform traffic load is considered, as shown in Fig. 11. Notice that under uniform traffic load the downstream stations have larger occupancy distributions, reflecting the unfairness property of the DQDB network.

Fig. 12 presents results obtained for station No. 3, for $M = 5$, $\lambda = .7$ under Poisson and Batch-3 arrivals. Simulation results are shown under interstation distances of 1 and 4 slots. The analytic results cannot capture the interstation distance. Notice that the buffer occupancy increases as the interstation distance increases under both arrival processes. Nevertheless, the analytical results seem to be in satisfactory agreement with the simulations, especially under Poisson arrivals.

VI Conclusions

The contribution of this paper may be summarized as follows. At first an interesting queueing system of potentially wide applicability has been formulated and studied. The service provided to a packet in the queue depends on the state of an underlying Markov chain at the time when the packet advances to the HoQ. Furthermore, it has been shown that the Markovian structure of the underlying process can be relaxed and weaker conditions are sufficient for the application of the developed analysis approach; such conditions are stated in Proposition 2 and the comment following its proof. The second contribution of this work is the development of the queueing model for the DQDB station by properly identifying (and approximating) the process which captures the station's perception of the state of the global distributed queue and, thus, determines its transmission (service) time. By applying the analysis of the presented queueing system, numerical results for the buffer occupancy distribution of the DQDB station have been obtained for the first time.

Appendix A

A Proof of Proposition 1

In this appendix, Proposition 1 is proven. Let a busy sub-period be defined as the time interval between the slot when a packet advances to the HoQ and the slot when this packet departs from the station. Let an idle period be defined as the time interval over which the station is idle. From the definition of $\left\{\hat{t}_k\right\}_{k\geq 1}$ it is easily established that this sequence contains exactly the boundaries of all the idle periods and busy sub-periods which may be observed. Thus, to prove the proposition it suffices to show that (4) holds at the boundaries of the idle periods and the busy sub-periods. The following two Lemmas will be needed.

Lemma 1: If (4) holds at the beginning of an idle period, it also holds at the end of this period.

Proof : According to the DQDB protocol, the evolution of $\left\{(REQ\text{-}CNTR)_j\right\}_{j\geq 0}$ during the idle periods is described by

$$(REQ\text{-}CNTR)_j = \left[(REQ\text{-}CNTR)_{j-1} + r_j - e_j\right]^+ \tag{A.1}$$

In view of (3), the evolution of the balance process, $\{b_j\}_{j\geq 0}$, and the request counter content process, $\left\{(REQ\text{-}CNTR)_j\right\}_{j\geq 0}$, are identical over an idle period. Thus, if the states (values) of the processes coincide at the beginning of an idle period they will coincide at the end of this period as well. $\square$

Lemma 2: If (4) holds at the beginning of a busy sub-period, it also holds at the end of this sub-period.

Proof : Let t_i and t_{i+1}, t_i, $t_{i+1} \in \left\{\hat{t}_k\right\}_{k\geq 1}$, denote the beginning and the end of a busy sub-period. By assumption,

$$b_{t_i} = (REQ\text{-}CNTR)_{t_i}\,. \tag{A.2}$$

According to the DQDB protocol, during the busy sub-period $(t_i, t_{i+1}]$ the REQ-CNTR is only incremented. Specifically, process $\left\{(REQ\text{-}CNTR)_j\right\}_{j=t_i+1}^{t_{i+1}}$ evolves according to the formula

$$(REQ\text{-}CNTR)_j = (REQ\text{-}CNTR)_{j-1} + r_j \ , \ \ t_i < j \leq t_{i+1} \tag{A.3}$$

where the initial value $(REQ\text{-}CNTR)_{t_i}$ is set equal to zero. Thus, the value of the REQ-CNTR at the end of the busy sub-period is given by

$$(REQ\text{-}CNTR)_{t_{i+1}} = \sum_{j=t_i+1}^{t_{i+1}} r_j \tag{A.4}$$

The balance process evolves according to (3). At the end of the busy sub-period its value is given by

$$b_{t_{i+1}} = b_{t_i} + \sum_{j=t_i+1}^{t_{i+1}} r_j - \sum_{j=t_i+1}^{t_{i+1}} e_j \tag{A.5}$$

Notice that the total number of empty slots leaving the tagged station over the busy sub-period is equal to the pending requests from the downstream users at the beginning of this busy sub-period. Thus,

$$b_{t_i} - \sum_{j=t_i+1}^{t_{i+1}} e_j = 0 \tag{A.6}$$

The above establish that

$$b_{t_{i+1}} = (REQ\text{-}CNTR)_{t_{i+1}} \tag{A.7}$$

and the proof of the Lemma is completed. □

Assuming proper initialization of the two processes - $b_0 = (REQ\text{-}CNTR)_0 = 0$ - then Lemma 1 implies that $b_{\hat{t}_1} = (REQ\text{-}CNTR)_{\hat{t}_1}$ at the end of the first idle period $\hat{t}_1$, which coincides with the beginning of the first busy sub-period. Then Lemma 2 implies that $b_{\hat{t}_2} = (REQ\text{-}CNTR)_{\hat{t}_2}$, where $\hat{t}_2$ is the end of the first busy sub-period, which coincides with the beginning of the next interval which may be either the second idle period or the second busy sub-period. By reiterating this argument the proof of the proposition is evident in view of the above Lemmas.

References

[1] IEEE 802.6. *Distributed Queue Dual Bus (DQDB) Subnetwork of a Metropolitan Area Network (MAN)*, Dec. 1990.

[2] C. Bisdikian, "Waiting Time Analysis in a Single Buffer DQDB (802.6) Network," *IEEE J. Select. Areas Commun.*, vol. SAC-8, pp. 1565-1573, Oct.1990.

[3] B. Mukherjee and C. Bisdikian, "A Journey Through the DQDB Network Literature," *IBM Report RC 170166 (75478), 7/11/91*, T.J. Watson Research Center, Yorktown Heights.

[4] M. Conti, E. Gregori, and L. Lenzini, "On the Approximation of the Slot Occupancy Pattern in a DQDB Network," *Proc. IEEE INFOCOM'92*, pp. 518-526, Florence, Italy, May 4-8, 1992.

[5] W. Jing and M. Paterakis, "Message Delay Analysis of the DQDB (IEEE 802.6) Network," *Proc. IEEE INFOCOM'92*, pp. 527-535, Florence, Italy, May 4-8, 1992.

[6] C.Y. Chen, G. Makhoul, and D. Meliksetian, "A Queueing Approach to the Performance Evaluation of DQDB," *Proc. 11th Int'l Phoenix Conf. on Comp. & Commun.*, Scottsdale, AZ, April 1992.

[7] J. Liebeherr, I. Akyildiz, and A. Tantawi, " An Effective Scheme for Pre-Emptive Priorities in Dual Bus Metropolitan Area Networks," *Proc. ACM SIGCOMM'92*, Baltimore, MD, Aug. 17-20, 1992.

[8] I. Stavrakakis and R. Landry, "Delay Analysis of the DQDB Man Based on a Simple Model," *Proc. ICC'92*, pp. 154-158, Chicago, IL, June 15-18, 1992.

[9] I. Stavrakakis and S. Tsakiridou, "A Queueing System with Markov-Structured Service Availabilities," *Research Report, CSEE/92/06-03*, CSEE Dept., University of Vermont.

[10] M.F. Neuts, *Structured Stochastic Matrices of M/G/1 Type and Their Applications*, Marcel-Dekker, New York, 1989.

[11] M.F. Neuts, "A Queue Subject to Extraneous Phase Changes," *Adv. Appl. Prob.*, vol. 3, pp. 78-119, 1971.

[12] D.M. Lucantoni, K.S. Meier-Hellstern, and M.F. Neuts, "A Single-Server Queue with Server Vacations and a Class of Non-Renewal Arrival Processes," *Adv. Appl. Prob.*, vol. 22, pp. 676-705, 1990.

[13] D.M. Lucantoni, "New Results on the Single Server Queue with a Batch Markovian Arrival Process," *Commun. Statist. - Stochastic Models*, vol.7(1), pp. 1-46, 1991.

[14] V. Ramaswami, "The N/G/1 Queue and its Detailed Analysis," *Adv. Appl. Prob.*, vol. 12, pp. 222-261, 1980.

[15] M.F. Neuts, "Some Explicit Formulas for the Steady-State Behavior of the Queue with Semi-Markovian Service Times," *Adv. Appl. Prob.*, vol. 9, pp. 141-157, 1977.

[16] H. Ahmadi, R. Guérin, and K. Sohraby, "Analysis of a Rate-Based Access Control Mechanism for High-Speed Networks," *IEEE Trans. Commun.*, vol. COM-41, pp. 940–950, June 1993.

[17] V. Ramaswami, "A Stable Recursion for the Steady State Vector in Markov Chains of the M/G/1 type," *Commun. Statist. - Stochastic Models*, vol. 4(1), pp. 183-188, 1988.

[18] J. Daigle, *Queueing Theory for Telecommunications*, Addison Wesley, 1992.

[19] E. Çinlar, *Introduction to Stochastic Processes*, Prentice-Hall, Englewood Cliffs, N.J., 1975.