Contribution to the SBT/IEEE International Telecommunications Symposium,
Rio de Janeiro, Brazil, September 3–6, 1990

A DISCRETE TIME ANALYSIS OF THE DQDB ACCESS PROTOCOL WITH GENERAL INPUT TRAFFIC

Th. Stock, P. Tran-Gia

Institute for Computer Science, University of Würzburg, Am Hubland, D-8700 Würzburg, F.R.G.

Abstract *We present an approximate performance study of the DQDB (Distributed Queue Dual Bus) medium access protocol, which is an emerging IEEE 802.6 standard for local and metropolitan area networks. The main objective of this paper is to study the influence of general traffic process characteristics to the protocol performance. Due to the consideration of general arrival and service processes, the analysis is derived in discrete-time domain. The algorithm is based on a decomposition approach of the medium access delay, using the technique of embedded G/G/1 models. Using the analysis method, the whole distribution of delays like medium access delay and other system characteristics can be properly calculated. Validations with simulation results show that the approximation technique developed in this paper is appropriate for a wide range of protocol parameters.*

1 Introduction

The Distributed Queue Dual Bus (DQDB) access protocol is a candidate for emerging high-speed local area and metropolitan area network standards, e.g. as being defined in IEEE 802.6. Attentions are devoted to this medium access scheme in some recent studies, both from technological and protocol performance viewpoints.

There is a number of simulation studies [3, 7, 18] and approximate analyses [19, 20, 21] dealing with performance aspects of various successive releases of the standardization process [2, 8, 11, 12, 13]. In Wong [18] attentions are devoted to the protocol behavior under saturated traffic conditions. The study gives analytical insight into the station-based traffic discrepancy and the relationship between the overload performance and the initial system state prior to the overload period. A comparative study is given in Huber et al. [7] dealing with the delay performance of FDDI and QPSX/DQDB in high-speed networks. Zukerman [19, 20, 21] studied various aspects of DQDB performance using approximate queueing analyses.

Since the DQDB medium access protocol is dedicated for use in high-speed metropolitan area networks and large local area networks, the number of stations to be considered in performance investigations should be chosen large enough to reflect the real system environments. This choice and the according number of events needed in simulation studies may lead to excessive simulation time. To investigate sufficiently large configurations with varying parameter ranges, analytical investigation methods are required.

The aim of the analysis method developed in this paper is to give closed-form solutions, which should be simple to evaluate but have a sufficient approximation accuracy over a realistic range of parameters. The analysis is composed by standard basic models of discrete-time G/G/1 type. The compound model for the evaluation of the medium access delay contains nested models, whereby the service time of the next model level is composed by the waiting time of the previous modeling level and station-dependent random processes. We refer to this as the concept of embedded models.

The paper is organized as follows. In Section 2 the main properties of the DQDB medium access mechanism are summarized. Section 3 gives an outline of modeling steps and details of the analysis. Some numerical results for system configurations will be presented in Section 4.

2 DQDB access mechanism

The basic logical structure of a DQDB access system is depicted in Fig. 1. In the following we will briefly outline those characteristics of DQDB which are relevant in the performance modeling context. More details can be found e.g. in the draft versions of the protocol standard [8].

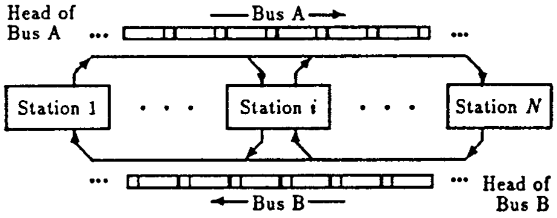

Figure 1: *DQDB system structure*

A DQDB system is based on a dual pair of slotted unidirectional buses flowing in opposite directions and operating in parallel at the MAC layer. Bus A is used for downstream and bus B for upstream payload traffic while a part of the corresponding control information is transferred on the opposite bus, respectively. Each station is attached to both buses and observes control (and payload) data passing to provide a local information record about the global queueing state of the system. This record is used by the station to participate in a distributed queueing scheme that is applied to the overall system. This shall help to achieve a fair system behavior, i.e. a behavior that approaches the case of a global FIFO service discipline.

Since the access mechanism is symmetrical for both directions of payload data transfer, our description will focus on one direction. We consider the downward data transfer on bus A and the corresponding request transfer on bus B. The traffic of requests forms a part of the control information that is exchanged between the stations and will be described later in this section. In the context of a discrete-time analysis described in this paper we restrict to one priority level of data traffic, while devoting attention to the influence of the input traffic characteristics on the overall protocol performance.

All information that is transferred on the buses is carried in slots, which consist of a one Byte access control field (ACF) and a 52 Byte segment area for isochronous and non-isochronous (asynchronous) traffic (cf. Fig. 2). This segment area is further subdivided into a segment header and a payload area in case of a non-isochronous segment

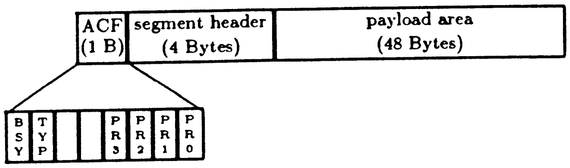

Figure 2: *Slot Format*

Two access control modes are provided to handle these different types of traffic. The pre-arbitrated (PA) access mode is reserved for isochronous services, i.e. those services that require a fixed bandwidth, like in the case of voice and video services. This mode is controlled by the slot generators, which mark the PA-slots by switching the BUSY and the SL_TYPE bit in the ACF to 1. Slots that can be used for non-isochronous services are identified by a SL_TYPE bit set to 0. Access to these slots is controlled by the stations themselves according to the queued-arbitrated (QA) access mode (→ QA-slots). As this access mode is investigated in our analysis we will describe it in more detail.

The main principles of the DQDB access mechanism can be summarized by means of the following principles, considering a station accessing bus A to transmit queued-arbitrated data segments:

- broadcast access request to all upstream stations
- keep track of access requests generated by downstream stations
- access bus A when all requests prior to its own are satisfied

Following these main mechanisms, if a station wants to transfer a non-isochronous segment downstream using bus A, it notifies this wish to all stations upstream by sending a request on bus B. This is done by using the request bit on the opposite bus. In parallel the station continuously takes notice about all requests flowing by on bus B. While the station has several separate queues for segments waiting to be transferred on both buses and different priority levels, the station schedules only one segment per bus. In other words, each station has one schedule position facing to each bus for each priority level, but only one of them can be active. The scheduled segment waiting to be transmitted in the station may not be sent before all preceding requests which were observed on bus B are served. To do this, the station has to wait until the corresponding observed number of free segments has passed by on bus A.

Considering only data transfer on bus A and one priority, a station can be in the following two states: IDLE and COUNTDOWN (see Fig. 3). We consider in the following the station i. For each bus and priority level the station has to maintain different counters, in particular the request counter (RQ) and the countdown counter (CD).

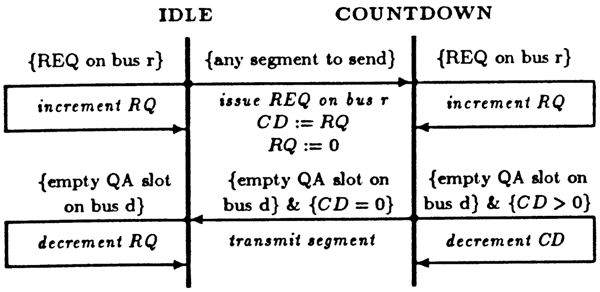

Figure 3: *A simplified state transition diagram*

1. IDLE-state: the station has nothing to send or was on immediate transition from state COUNTDOWN. The request counter maintains the number of requested transmissions sent by stations $i+1, ..., N$. This counter is decremented upon observing a free slot flowing by on bus A and is incremented upon seeing a request passing by on bus B.
2. COUNTDOWN-state: the station has data segments to transmit. A segment has been scheduled at time t_0 for transmission. The request counter indicates the number of request arrivals after t_0. The countdown counter maintains the number of requests which arrived prior to t_0 and have to be served before the scheduled segment. In this state, the countdown counter is decremented by observing a free slot flowing by on bus A while the request counter is incremented upon arrival of a new request on bus B.
3. State transitions: A state transition from IDLE to COUNTDOWN is processed as follows. The station enqueues a request to the local request queue, sets the countdown counter to the current value of the request counter and then resets the request counter. The station always takes over from COUNTDOWN to IDLE after sending a segment. This is followed immediately by a backward state transition from IDLE to COUNTDOWN if there are still segments waiting in the station.

Each time when a transition form IDLE to COUNTDOWN is executed, a request is generated and placed in the local request queue. This queue also is represented by a counter (RQC). Enqueueing and dequeueing is done by incrementing and decrementing RQC by one, respectively. It should be noted here that this request queue operates asynchronously to the above data segment queueing system.

3 Discrete-time model and analysis

3.1 System model and assumptions

We consider a network with N attached stations operating with the DQDB access protocol. The distance between station i and j is denoted by r_{ij}. The network carries both isochronous and non-isochronous traffic. The isochronous traffic (e.g., voice, video etc.) is preallocated slot-wise by the slot generator. As mentioned above, in the following we only consider the case of one priority level. Further, since we have a dual symmetrical bus system with decomposable traffic flows, it is sufficient to investigate only one data flow direction. The analysis of the other direction is analogous.

We devote now attention on the downstream data traffic on bus A and the corresponding upstream request traffic on bus B. The incoming traffic streams are characterized by discrete-time random processes by means of the discrete-time random variable (r.v.) A_{ij} with mean $EA_{ij} = \frac{1}{\lambda_{ij}}$ and coefficient of variation (c.v.) c_{ij}. Accordingly, the total traffic generated at station i to be transferred downstream on bus A is the random process A_i, which is a compound process represented by a superposition of the processes $A_{i,i+1}, \dots, A_{i,N}$.

We denote by p_I and $(1-p_I)$ the percentages of the isochronous and the remaining bandwidth available to non-isochronous traffic, respectively. With τ be the slot duration, the asynchronous bus utilization ρ_i of station i and the total asynchronous traffic ρ are

$$\rho_i = \sum_{j=i+1}^{N} \lambda_{ij} \cdot \tau \quad \text{and} \quad \rho = \sum_{i=1}^{N} \rho_i \,. \tag{1}$$

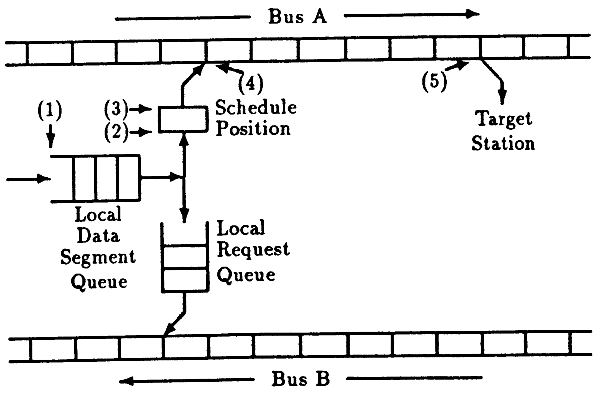

Figure 4: *Sending part and modeling concept*

In the following we observe a data segment, which is generated in station i and passed across the medium access control unit. It is then to be transmitted to station j, where $j > i$. The segment itself will be transferred on bus A and its corresponding request on bus B. As depicted in Fig. 4, we take into account the following time instants, which are significant for the calculation of the segment transfer time according to the DQDB access mechanism:

(1) arrival epoch of the segment

(2) time instant, at which the observed segment is scheduled for transmission on bus A. At this time a request is created and is to be sent on bus B. The segment is ready to be transmitted, but still has to wait according to the FIFO discipline in the globally distributed queue.

(3) the segment is at the head of the global queue and is enabled to be sent, but still has to wait for a free slot flowing by on bus A.

(4) end of the transmission on the bus.

(5) the segment has arrived at the receiving station j.

This observation leads to a decomposition of the segment transfer time, where the following discrete time random variables are defined:

T_{12} : r.v. for the waiting time in the local queue in station i; each priority level has a separate local queue.

T_{23} : r.v. for the waiting time in the schedule position in station i. This waiting time is dependent on the state of the global queue, in conjunction with the distributed queueing scheme.

T_{34} : r.v. for the virtual transmission time (see Fig. 5).

T_{45} : propagation delay from station i to station j.

According to this observation, the medium access delay is T_{14} and the segment transfer time is T_{15}.

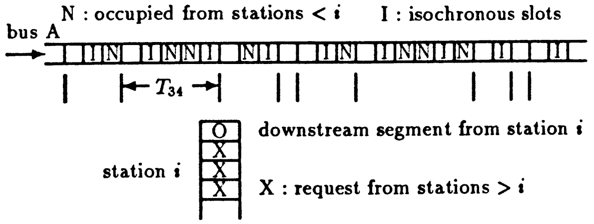

Figure 5: *Virtual transmission time* T_{34}

3.2 Embedded modeling and medium access delay

We will consecutively determine the distribution functions of T_{34}, T_{23} and T_{12}, which finally deliver the distribution of the medium access delay T_{14}.

The discrete time random variable T_{34} can be interpreted as the interval between free consecutive slots seen from the station i (cf. Fig. 5). Station i sees a slot stream on bus A, where two types of busy slots can be observed: i) isochronous slot patterns which are periodically allocated and ii) slots already occupied by non-isochronous traffic from stations $1, \dots, i-1$. The distribution of isochronous patterns on the slot stream is assumed to be uniform. Considering in this paper the special

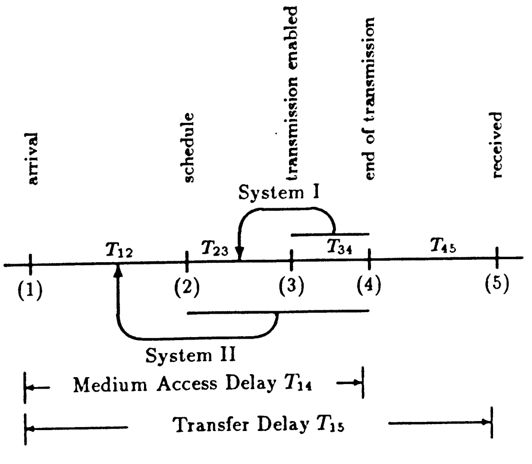

Figure 6: *Segment lifetime and analysis concept*

case of $p_I = 50\%$, we describe approximately the interval between free slots seen from station i with the following geometric distribution:

$$\Pr\{T_{34} = k \cdot \frac{1}{p_I} \text{ slots}\} \;=\; q_i^{k-1}(1-q_i)\,, \quad k = 1,2,\ldots$$
$$\text{with} \quad q_i = \sum_{j=1}^{i-1} \frac{\rho_j}{1-p_I}\,. \qquad (2)$$

From modeling point of view, T_{34} is the service time seen from all segments waiting for transmission, which have been noticed from station i, i.e. those from stations j with $j \geq i$. The waiting time T_{23} of segments in the schedule position (cf. Fig. 4) is approximated by the waiting time in a discrete-time G/G/1 system (system I of Fig. 6) with service time T_{34}. The arrival process of this system is the superposition of all downstream segment arrival processes $A_i, \ldots, A_N$ of stations $i, i+1, \ldots, N$. The calculation of the waiting time of this discrete-time G/G/1 system is done using the algorithm as described in appendix A. One important property of system I is that the mean service time increases while the arrival rate decreases with higher number i of the observed station.

As mentioned , the interval T_{24} can be seen as the virtual transmission time seen from those segments, which arrived at station i to be transferred on bus A. The distribution of T_{24} is simply the convolution of the distributions of T_{23} and T_{34}. We again describe the waiting process in the local queue (see Fig. 4) by means of a discrete-time G/G/1 system (system II in Fig. 6). The service process of this queueing system is modeled using the embedded modeling concept, i.e. the service time of system II is the sojourn time of system I, denoted by the r.v. T_{24}. The arrival process is the compound process of all arrival processes with r.v. $A_{ij}, j = i+1, \ldots, N$. Again we apply the G/G/1 analysis mentioned above to calculate the distribution of T_{12} and derive T_{14} by summing T_{12} and T_{24}. The decomposition of the medium access delay as shown in Fig. 6 is not only a time decomposition, but contains nested intervals computed by different submodels.

To obtain the total transfer delay, the propagation delay T_{45}, which can be easily estimated from the station-to-station distance, r_{ij}, has to be added. Out of the above distributions values of interest like means and coefficients of variation of the medium access delay and the total transfer time can be derived.

4 Numerical results

We consider a metropolitan area network with $N = 49$ stations, which are equidistantly located on a dual bus system of length 100 kilometers and transmission capacity 136 Mbps each. The slot length is chosen at 53 Bytes (48 B segment payload, 4 B header, 1 B ACF). Although any percentage of isochronous traffic can be chosen, we assume a fixed amount of $p_I = 50\%$. In the diagrams shown in this section, we normalized the asynchronous traffic to the available bandwidth for non-isochronous traffic streams as $\rho^* = \rho/(1-p_I)$. Delays are given in μsec.

To obtain a parametric representation of the arrival processes, we consider here A_{ij} having a negative binomial distribution given by the mean EA_{ij} and the c.v. c_{ij} :

$$a_{ij} = \binom{y+k-1}{k} q^y (1-q)^k\,, \quad 0 \leq q < 1\,, \quad y \text{ real}, \qquad (3)$$

where

$$q = \frac{1}{EA_{ij} \cdot {c_{ij}}^2}\,, \quad y = \frac{EA_{ij}}{EA_{ij} \cdot {c_{ij}}^2 - 1}\,, \quad EA_{ij} \cdot {c_{ij}}^2 > 1\,. \qquad (4)$$

Fig. 7 depicts the mean access delay of stations 1, 13 and 40 as a function of the asynchronous traffic intensity for different input process coefficients of variation. The medium access delay is measured for one transmission direction (bus A) only. The arrival processes A_{ij} for traffic from stations i to stations j $(i < j)$ are negative binomially distributed with identical parameters, i.e. the mean $EA_{ij} = \frac{1}{\lambda}$ and the c.v. c_a. λ is given as $\lambda = \rho \cdot c^* \cdot \frac{2}{N(N-1)}$, with c^* be the bus speed (in $\frac{slots}{sec}$). Comparisons with simulation results show that the analysis is sufficiently accurate for dimensioning purposes.

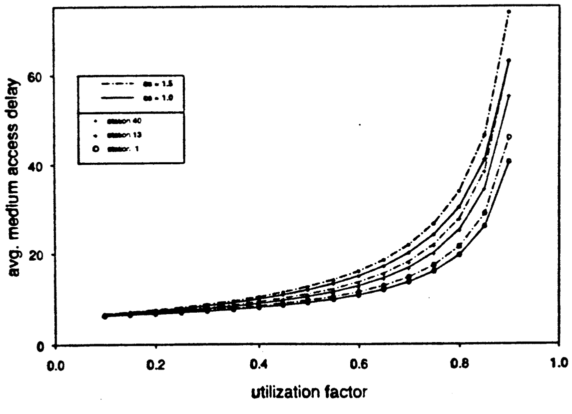

Figure 7: *Medium access delay vs asynchronous traffic*

It can be seen in Fig. 8 that the mean medium access delay is station-dependent, according to the often mentioned unfairness behavior of the DQDB protocol (cf. [7, 17, 18]). The first station has the smallest, while the station 40 has the largest access delay. Behind this station the mean access delay starts to decrease slightly. The same effect is observed for different input process parameters. It should be noted that the capacity limit of the entire system is defined by the station with the largest access delay.

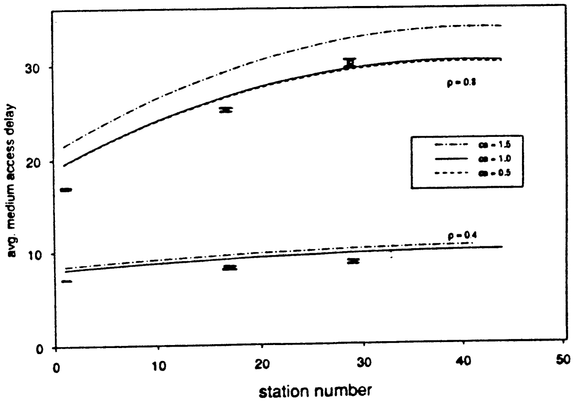

Figure 8: *Dependence of access delay on station number*

As expected, it is clearly shown that for a given total traffic intensity the system capacity limit depends strongly on the c.v. c_a. It is revealing that the results for $c_a < 1.0$ do not differ significantly from those for $c_a = 1.0$. This holds for the analytical results as well as for the simulation. The 95 % confidence intervals of the simulative results for $c_a = 0.5$ and $c_a = 1.0$ are depicted in Fig. 8 and cannot be destinguished.

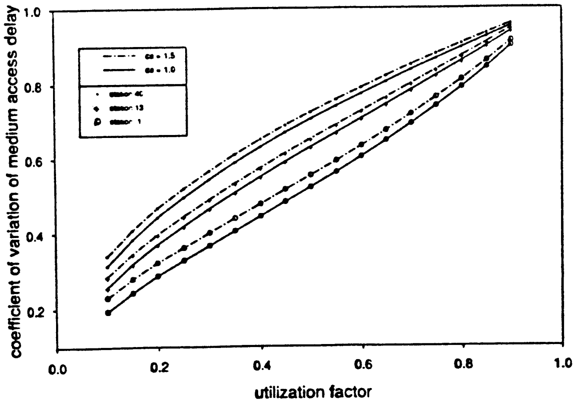

Figure 9: *Coefficient of variation of medium access delay*

Since the entire distribution of T_{14} can be obtained, higher moments of the medium access delay can be numerically determined. One such result is shown in Fig. 9, were the coefficient of variation vs. the normalized asynchronous traffic is plotted. It can be seen, that even highly varying arrival processes A_{ij} do not produce excessive variation at T_{14}. This is due to the fact that the superposition of arrival processes A_{ij} with $c_{ij} = c_a > 1.0$ results in a coefficient of variation $c < c_a$.

5 Conclusion and outlook

We presented an approximate performance study of the DQDB medium access protocol using discrete-time queueing models. The novel model component mainly observed in this paper is the investigation of the influence of general traffic processes on the protocol performance. This leads to a more realistic modeling of traffic streams in computer network environments. A percentage of preassigned isochronous traffic in the system is taken into account, which also can be chosen arbitrarily. Due to the consideration of general arrival and service processes, the analysis is derived in discrete-time domain. The analysis is based on a decomposition approach of the medium access delay, using embedded modeling technique. The main results obtained are approximate expressions for various delays in the system like the medium access delay given in the form of Z-transforms. From these basic relationships, further measures of interest like the mean and the coefficient of variation of the mean access delay can be derived. As shown in comparisons with simulations, the accuracy of the approximation is sufficient for a wide range of protocol parameters.

Acknowledgement

The authors would like to thank Rainer Dittmann for the valuable helps and programming efforts.

References

[1] M. H. Ackroyd, "Computing the Waiting Time Distribution for the G/G/1 Queue by Signal Processing Methods", *IEEE COM-28*, 1980, 52–58.

[2] Z. L. Budrikis et al., "QPSX: A Queue Packet and Synchronous Circuit Exchange", *ICCC '86*, Munich, 1986, 288–293.

[3] P. Davids, Th. Welzel, "Performance Analysis of DQDB Based on Simulation", *Third IEEE Workshop on Metropolitan Area Networks*, San Diego, March 28 – 30, 1989.

[4] B. T. Doshi, A. A. Fredericks "Visual Modeling and Analysis With Application to IEEE 802.6 DQDB Protocol", *ITC Specialist Seminar*, Adelaide, 1989.

[5] J. Filipiak, "Access Protection for Fairness in a Distributed Queue Dual Bus Metropolitan Area Network", *ICC '89*, San Diego, March 1989.

[6] P. Henrici, "Fast Fourier Methods in Computational Complex Analysis", *SIAM Review*, vol. 21, 1979, 481–527.

[7] M. N. Huber, K. Sauer, W. Schödl, "QPSX and FDDI-II Performance Study of High Speed LANs", *EFOC/LAN 88*, Amsterdam, June 29 – July 1, 1988.

[8] IEEE Working Group, *Proposed IEEE Standard 802.6 - Distributed Queue Dual Bus (DQDB) - Metropolitan Area Network (MAN)*, Draft versions: June 1988, November 1988, Spring 1989.

[9] L. Kleinrock, "*Queueing Systems, Volume I: Theory, Volume II: Computer Applications*", John Wiley & Sons, New York, 1975/76.

[10] D. V. Lindley, "The Theory of Queues with a Single Server", *Proc. of the Cambridge Philosophical Society*, vol. 48, 1952, 277–289.

[11] J. F. Mollenauer, "Standards for Metropolitan Area Networks", *IEEE Communications Magazine*, 26, April 1988, 15–19.

[12] R. M. Newman, J. L. Hullett, "Distributed Queueing: A Fast And Efficient Packet Access Protocol for QPSX", *ICCC '86*, Munich, 1986, 294–299.

[13] R. M. Newman, Z. L. Budrikis, J. L. Hullett, "The QPSX MAN", *IEEE Communications Magazine*, 26, April 1988, 20–28.

[14] P. Potter, M. Zukerman, "A Discrete Shared Processor Model for DQDB", *ITC Specialist Seminar*, Adelaide, 1989.

[15] K. Sauer, W. Schödl, "Performance Aspects of the DQDB Protocol", *ITC Specialist Seminar*, Adelaide, 1989.

[16] P. Tran-Gia, "Discrete-Time Analysis for the Interdeparture Distribution of GI/G/1 Queues", *Proc. Seminar on Teletraffic Analysis and Computer Performance Evaluation*, June 1986, Amsterdam.

[17] P. Tran-Gia, T. Stock, "Approximate Performance Analysis of the DQDB Access Protocol", *ITC Specialist Seminar*, Adelaide, 1989, to appear in *Computer Networks and ISDN systems.*

[18] J. W. Wong, "Throughput of DQDB Networks under Heavy Load", *EFOC/LAN 89*, Amsterdam, June 14 – 16, 1989.

[19] M. Zukerman, "Queueing Performance of QPSX", *12th International Teletraffic Congress*, Torino, June 1–8, 1988, paper 2.2B.6.

[20] M. Zukerman, "Approximations for Performance Evaluation of the Packet Access Queue in QPSX: the IEEE 802.6 Evolving MAN Standard", *Australian Telecommunication Research*, 22, No. 2, 1988, 53–62.

[21] M. Zukerman, "Circuit Allocation and Overload Control in a Hybrid Switching System", *Computer Networks and ISDN Systems*, 16, 1988/89, 281–298.

[22] M. Zukerman, P. Potter, "The DQDB Protocol and its Performance under Overload Traffic Conditions" *ITC Specialist Seminar*, Adelaide, 1989.

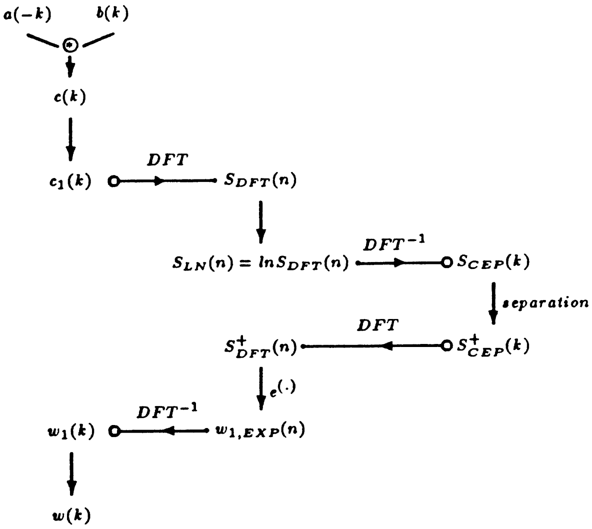

Figure 10: *The Cepstrum algorithm for G/G/1 analysis*

A Analysis of G/G/1 in transform domain using the Cepstrum concept

Given a stationary G/G/1 system with finite distributions of interarrival and service time $a(k)$ and $b(k)$, respectively, the system function is defined as the convolution

$$c(k) = a(-k) * b(k)\,. \tag{5}$$

The waiting time distribution is denoted by

$$w(k) = \pi_0(w(k) * c(k))\,, \tag{6}$$

where

$$\pi_0(x(k)) = \begin{cases} x(k) & \text{if } k > 0 \\ \sum_{i=-\infty}^{0} x(i) & \text{if } k = 0\,. \end{cases} \tag{7}$$

For the waiting time distribution function

$$W(k) = \sum_{i=-\infty}^{k} w(i) \tag{8}$$

an analogous form can be obtained

$$W^-(k) + W(k) = c(k) * W(k)\,, \tag{9}$$

where

$$W^-(k) = \begin{cases} 0 & \text{if } k \geq 0 \\ (c(.) * W(.))(k) & \text{if } k < 0\,. \end{cases} \tag{10}$$

Transforming into Z-domain one arrives at

$$W^-_{ZT}(z) \cdot \frac{1}{w_{ZT}(z)} = \frac{c_{ZT}(z) - 1}{1 - z^{-1}} = S_{ZT}(z)\,. \tag{11}$$

In this equation $W^-_{ZT}(z)$ has to be eliminated to obtain the Z-transform of the waiting time distribution $w_{ZT}(z)$.

The algorithm is illustrated in Fig. 10 and consists of the following major steps (see [1]):

1. Calculation of the transfer function $S_{ZT}(z)$ out of the system function $c(k) = a(-k) * b(k)$. Since $c(k)$ is of finite length, $S_{ZT}(z)$ can be equivalently represented by the discrete Fourier transform (DFT) $S_{DFT}(n)$.
2. Calculation of the complex Cepstrum $S_{CEP}(k) = DFT^{-1}(\ln[S_{DFT}(n)])$.
3. Separation of $S^+_{CEP}(k)$, which consists of non-negative components of $S_{CEP}(k)$. The function $S^+_{CEP}(k)$ is the Cepstrum of the unnormalized waiting time distribution $w_1(k)$.
4. Inverse transformation of $S^+_{CEP}(k)$ to get $w_1(k)$ and the normalization of $w_1(k)$ to obtain finally the waiting time distribution $w(k)$ of the G/G/1 system.