A Mean Value Analysis for Throughputs and Waiting Times of the FDDI Timed Token Protocol

Michael Tangemann

Institute of Communications Switching and Data Technics
University of Stuttgart, Seidenstr. 36, D-7000 Stuttgart 1, FRG
Telefax: +49 711 121-2477, E-Mail: Tangemann@DSOIND5.Bitnet

The Fibre Distributed Data Interface (FDDI) is a HSLAN protocol which employs a Timed Token Protocol for the media access of its packet switched traffic classes. Although the Timed Token Protocol is well understood and many performance studies based on simulation or analytical techniques have been published, some fundamental equations describing the performance of the FDDI Timed Token Protocol have not been found yet. This paper deals with a mean value analysis of an FDDI system with isochronous, synchronous and asynchronous traffic over the whole load range. Although the analysis derived here is simple, it considers the main effects that determine the performance of the MAC protocol, which is derived in terms of throughputs and mean waiting times of asynchronous as well as synchronous packets. Some examples are discussed, and the results are validated by detailed simulations.

1 Introduction

During the last years, several proposals for High Speed Local Area Networks (HSLANs) have been developed, which provide larger bandwidths than LANs (typically 100 Mbps or more), cover larger geographical areas and allow the integration of different traffic types. Two of these proposals, FDDI and DQDB, have become international standard proposals in the meantime.

FDDI (Fibre Distributed Data Interface) was standardized by the American National Standards Institute X3T9 committee. It is based on a dual fibre optic ring with a maximum ring length of 100 km and a transmission rate of 100 Mbps. Every station connected to the ring may support up to three different traffic classes. An initial version (FDDI-I) of the standard provides for two packet switched (PS) traffic classes, namely asynchronous traffic with 8 priorities and synchronous traffic, where connections with guaranteed bandwidth and delay can be established. This first version has been extended to a second one (FDDI-II), which additionally provides isochronous, i.e. circuit switched (CS) channels. It is based on a 125 µs cycle, which is partitioned into 16 so-called Wide Band Channels (WBCs). Each of these WBCs can be used for either CS or PS traffic.

The access to the medium in FDDI-I and for the PS traffic classes of FDDI-II is controlled by the so-called Timed Token Protocol, which was developed by Grow [9] and Ulm [21] and finally standardized as the Media Access Control (MAC) protocol for the IEEE 802.4 Token Bus and for FDDI [6]. It belongs to the class of Token Passing Protocols and allows the integration of time-critical (synchronous) and non time-critical (asynchronous) traffic. The maximum transmission time for synchronous packets at a station is negotiated during connection set-up according to the bandwidth requirements of the synchronous connection. The asynchronous queues may basically be served exhaustively with a token rotation time dependent upper limit. Exhaustive service provides very good delay and throughput characteristics compared to other service disciplines such as gated or limited service (cf. e.g. [17, p. 123]). However, as the total load of the system increases, some of the stations start hogging the ring bandwidth, which leads to very large token rotations. Therefore, a timer based mechanism that forces overloaded stations to pass the token to the next station must be added in order to improve the delay behaviour.

This timer mechanism works as follows: Every station measures the token rotation time with a Token-Rotation Timer (TRT). When the token arrives at station \( i \), the token rotation time is copied into the Token Holding Timer (THT), which counts upwards during asynchronous packet transmissions and expires upon reaching a threshold called Target Token Rotation Time (TTRT). The TTRT was negotiated during the initialization of the ring. After the expiry of the timer, no more packets may be transmitted and the token must be passed to station \( i + 1 \). Three cases are possible: 1. If the load of the system and especially of station \( i \) is light, its queues are emptied before the timer elapses, i.e. station \( i \) is served exhaustively. 2. If station \( i \) is heavily loaded, its service is limited by the THT. 3. If the total load of the ring is large, the token rotation time can be larger than TTRT. In this case the token is said to be late, and no packets may be transmitted on token arrival.

Using this timer mechanism, the Timed Token Pro-
This protocol guarantees upper bounds for mean and maximum cycle times, e.g., the mean token rotation time is always smaller than $TTRT$. These bounds have been proved by Sevcik and Johnson [16, 10], and they are the basis for the guaranteed performance of the synchronous traffic, which is given in terms of minimum bandwidth and maximum delay. Multiple asynchronous priorities can be introduced by defining the order of service inside the stations and different threshold values $T_{Pr,i} \leq TTRT$ for each priority.

First performance results concerning throughput limits have been presented by Ulm [21]. A procedure to estimate the throughput of each asynchronous traffic class has been presented by Dykman and Bux [4, 5], and an estimation of the cycle time and the node throughput has been derived by Pang and Tobagi [14], who consider the deterministic behaviour of the Timed Token Protocol with synchronous and asynchronous traffic under heavy load. This method has been extended by Va­len­z­ano, Mon­tus­chi and Cim­i­niera [22], who de­vel­oped approx­i­mate ex­pressions for the mean cycle time and the asynchronous throughput under heavy load.

Some recent publications are concerned with the deriv­a­tion of delay approximations. Takagi [18] and Nakamura et. al. [13] considered single buffer FDDI systems and obtain results for throughputs and mean waiting times. Based on an $M/G/1$ server vacation model, Gen­ter and Vastola [8] derived approximate expressions for the delay. Karve­las and Leon-Garcia [11] derived mean waiting time approximations for FDDI from exhaus­tive service polling models with extended switchover times, whereas Yue and Brooks [23] used a polling model with limited service, where the asymptote is shifted according to the maximum possible throughput. Lam­aire and Spiegel [12] finally condensed the latter two articles and provide some extensions.

In [19, 20], we developed an analysis of the FDDI MAC protocol for one asynchronous priority level, which is based on a detailed model and therefore uses a complex iterative algorithm, from which throughput and mean waiting time values can be obtained. The purpose of the present paper is to provide a mean value analysis for throughputs and waiting times of synchronous and asynchronous packets in a general FDDI (I or II) system for any given load scenario. Since only major effects limiting the performance are considered, the analysis is easy to understand and can be performed on a pocket calculator, but it still provides sufficiently accurate results.

This paper is organized as follows: After a model description, the first part of the analysis is presented. It deals with the analysis of the cycle (token rotation) time and the throughput of each queue. Based on these results, an approximation for the mean waiting times of the packets will be derived in section 4. Finally, some examples will be given in section 5.
traffic at station $i$ are defined in the same way. Throughout this paper, we assume that $\rho_s$ is not larger than the limit specified in the FDDI MAC standard [6] and that therefore no synchronous overload can occur.

Station latency and transmission delay of the medium are included in the switchover times $r_i$. Dynamic overheads such as token transmission and transmitter idle times of the stations, which increase the switchover time if the token is used for packet transmissions, as well as the protocol overrun, i.e., finishing the packet transmission after THT expiration, are neglected here since they are dominated by other effects.

3 Mean Cycle Time and Throughput Calculation

In this section, we want to estimate the throughput of each station and priority level. Since system, station and queue throughputs depend strongly on the cycle time and the priority thresholds, we first perform a mean value analysis of the cycle time, which is defined as the time between two successive token arrivals at a station. For reasons of symmetry, the mean cycle time $c$ is the same for all stations. Once $c$ is known, the individual throughputs can be expressed in terms of $c$.

In order to determine $c$, it will be decomposed into the cycle time of the empty system $c_0 = \sum_{i=1}^{N} r_i$ and synchronous and asynchronous queue service times $s_i$ and $a_i$. In a first approach, we assume that every station has two queues, one for synchronous traffic and one for asynchronous of a single priority. An extension to the case of multiple asynchronous priorities at a station will be given at the end of this section.

In our analysis, each asynchronous queue can be in one of three possible states. If the total load of the system is light, every queue will be served exhaustively and no packet losses occur. Queues that have no packet loss are said to be in state 'l' (light load). Their contribution to the cycle time is

$$a_i = a_{i,i} = \rho a_i \cdot c \quad (8)$$

Under high load, some of the queues suffer packet loss but still have a throughput greater than zero, which is denoted by the queue state 'o' (overload). The service time of an overloaded queue is determined according to the Timed Token Protocol

$$a_i = a_{o,i} = T_{pri,i} - T_{RT} = T_{pri,i} - c \quad (9)$$

independently of the load of the queue. In (9) the value of the Token Rotation Timer (TRT) upon token arrival is assumed to be equal to the cycle time, which is exactly correct if the Token is never late. Under overload, $T_{RT} \approx c$ is used as an approximation.

The third queue state will occur especially in systems with asymmetric load and different priorities, where the throughput of the queues with low priority may be decreased to zero. These queues are in state 'd' (deferred), and their contribution to the cycle time is

$$a_i = a_{d,i} = 0 \quad (10)$$

Queues change from state 'l' to 'o' as $a_{i,i} = a_{o,i}$. This is the load situation where they reach their peak loads, which is characterized by

$$c = c_{p,i} = \frac{T_{pri,i}}{1 + \rho a_i} \quad (11)$$

and queues enter state 'd' as $a_{o,i} = a_{d,i}$, which means that

$$c = c_{d,i} = T_{pri,i} \quad (12)$$

Therefore the following conditions determine the queue states:

Queue $i$ is in state

$$\{\begin{array}{ll} 'l' & \text{if } 0 < c < c_{p,i} \\ 'o' & \text{if } c_{p,i} < c < c_{d,i} \\ 'd' & \text{if } c_{d,i} < c \end{array} \quad (13)$$

Synchronous queues are assumed to be always under light load, i.e., no synchronous packets are lost and the synchronous queue service times are given by

$$s_i = \rho s_i \cdot c \quad (14)$$

Combining (8) - (10) and (14), the mean cycle time can be expressed by

$$c = c_0 + \sum_{i=1}^{N} a_i + \sum_{i=1}^{N} s_i = c_0 + \sum_{i \in o} T_{pri,i} \quad (15)$$

$$= \frac{c_0 + \sum_{i \in o} T_{pri,i}}{1 - \rho_s - \sum_{i \in l} \rho a_i + \sum_{i \in o} 1}$$

This equation can be solved in an iterative manner by assuming a set of queue states, verifying them by using (13) and correcting the queue states iteratively if necessary. Another possibility is calculating $c$ as a function of $\rho_s$ in a piecewise manner starting with $\rho_s = 0$. Here, none of the queues is in state 'o' nor 'd', and (15) yields an expression for $c$ which is valid up to the $\rho_s$ for which $c$ reaches the smallest value of the set $\{c_{p,i}, c_{d,i}\}$. Beyond this value, (15) again gives a unique solution for $c$, which is valid up to the next value of the set $\{c_{p,i}, c_{d,i}\}$. This method can be continued until the system is under heavy load and all asynchronous queues are either in state 'o' or 'd'.

The above results can now easily be extended to the case of more than one asynchronous queue at a station by inserting for priority $j$ at station $i$ in state 'o'

$$a_{o,ij} = T_{pri,ij} - (c + \sum_{k=1}^{j-1} a_{ij}) \quad (16)$$

where a smaller $j$ denotes a higher priority. Note that this is not correct for the IEEE 802.4 Token Bus, where due to different timer settings always $a_{o,ij} = T_{pri,ij} - c$.

As soon as the mean cycle time $c$ and hence the individual queue service times $a_{ij}$ are known, the normalized
information throughput $\gamma_{a,ij}^p$ of priority $j$ at station $i$ can be determined easily according to

$$\gamma_{a,ij} = a_{ij}/c ,$$

(17)

$$\gamma_{a,ij}^p = (1 - p_f) \cdot \gamma_{a,ij}$$

(18)

and

$$\gamma_{a,ij} = \gamma_{a,ij} - \frac{L_{a,ij}}{L_{a,ij} + L_o} .$$

(19)

Since synchronous queues are assumed to have no loss, always $\gamma_{a,i} = \rho_{a,i}$.

These very simple results need some further explanation. In general, cycle times of the Timed Token Protocol have very complex non-i.i.d. properties and may not only be represented by their mean values. However, by studying scenarios similar to those in [4, 5] under various load situations, we observed that the long-term behaviour of the system under overload is periodic with large numbers of different cycle times in a period.

Let us conclude this section with some remarks.

Throughout this section, no assumptions have been made about the distributions of the individual interarrival and packet service times. Therefore, the analysis can be applied to FDDI systems with general sources which are characterized only by their offered load.

An interesting effect of priority inversion can be observed by interpreting (11). If $c_{p,i} < c_{p,k}$ for any $i \neq k$ then $\gamma_{a,i,\text{max}} < \gamma_{a,k,\text{max}}$, i.e., queue $i$ has a lower throughput than queue $k$ even if $T_{\text{Pri},i} > T_{\text{Pri},k}$. However, this inversion of the priority levels can be avoided easily by dimensioning $T_{\text{Pri},i}$ and $T_{\text{Pri},k}$ such that for the maximum value of $\rho_{a,i}$ and the minimum value of $\rho_{a,k}$ the relation $c_{p,i} > c_{p,k}$ is still valid.

Another important and already well-known result is that for a symmetric system with one asynchronous priority. Here, we get

$$c = \begin{cases} c_0 / (1 - \rho_a) & \text{if } \rho_a \leq \gamma_{a,\text{max}} \\ c_0 + N \cdot \text{TTRT} / (N + 1) & \text{if } \rho_a > \gamma_{a,\text{max}} \end{cases}$$

(20)

and

$$\gamma_a = \sum_{i=1}^{N} \gamma_{a,i} = \begin{cases} \rho_a & \text{if } \rho_a \leq \gamma_{a,\text{max}} \\ \gamma_{a,\text{max}} & \text{if } \rho_a > \gamma_{a,\text{max}} \end{cases}$$

(21)

$$\gamma_{a,\text{max}} = \frac{\text{TTRT} - c_0}{\text{TTRT} + c_0 / N} .$$

(22)

A similar expression for $\gamma_{a,\text{max}}$ including overrun has already been found in [4], and again it can be seen clearly that Ulm’s result for $\gamma_{a,\text{max}}$ [21] is an upper limit and correct for large $N$. In [22] similar expressions for an overloaded single-priority system including synchronous traffic have been derived, which can now be obtained much more easily directly from (15).

It is also important to mention that according to (21) $\gamma_{a,\text{max}}$ does not depend on the transmission rate $\omega$ of the medium. That means that as long as the round trip delay $c_0$ (which is mainly determined by the ring length), the TTRT and the number of stations are held constant, the maximum relative throughput remains the same, even if $\omega$ is increased to 1 Gbps or more. This has been doubted recently in some publications. The derivation in [24], e.g., is based on the erroneous assumption that the stations are served according to the limited-1 service discipline, for which of course $\gamma_{a,\text{max}}$ decreases with an increasing transmission rate $\omega$. However, for the service discipline of the Timed Token Protocol, (21) holds.

### 4 Mean Waiting Times

For the purpose of a waiting time analysis in this section, the model will be changed slightly. Instead of considering a model with $N$ stations, where each station corresponds to an FDDI station and can have multiple queues, the queues are now treated separately, and a cyclic polling model consisting of $N$ queues is obtained. This model contains synchronous as well as asynchronous queues. Since deferred queues do not have any throughput at all, they are removed from the system.

Let us first assume that all queues are in state 1, i.e., every queue is under light load and will be served exhaustively. This was part of the definition of state 1 in the previous chapter. Therefore, the FDDI system can be modeled as a polling system with exhaustive service. If now Markov arrivals are assumed, from [17] the following result for the mean waiting time $w_k$ of packets in queue $k$ of such a polling system can be extracted by inserting (4.32a,b) into (4.35b)

$$w_k = w_{k,i} = (1 - \rho_k) \cdot \frac{c_k^{(2)}}{2c} .$$

Here, $c_k^{(2)}$ is the second moment of the cycle time experienced by queue $k$, which in the general case depends on the queue index $k$. Unfortunately, no exact closed-form expression for $c_k^{(2)}$ is available. This is the reason why we use an approximation based on the pseudo conservation law for the weighted sum of the waiting times of a polling system with exhaustive service [3, 17], which says that in our notation

$$\sum_{k=1}^{N} \rho_k \cdot w_k = \rho \cdot \frac{\sum_{k=1}^{N} \lambda_k \cdot \delta_k^{(2)}}{2(1 - \rho)} + \frac{c}{2} \cdot [\rho - \sum_{k=1}^{N} \rho_k^{(2)}} ,$$

(24)

where $\rho = \sum_{k=1}^{N} \rho_k$ and we used the fact that the switch-over times are constant. As in [2] (Assumption B) for limited-1 service, we now assume that $c_k^{(2)} = c^{(2)}$, i.e., independent of $k$. Therefore, $c^{(2)}$ can be calculated by inserting (23) into the conservation law (24), and $w_k$ can
be expressed as

\[ w_k = (1 - \rho_k) \cdot \frac{\sum_{k=1}^{\infty} \rho_k \cdot w_k}{\sum_{k=1}^{\infty} (1 - \rho_k) \cdot \rho_k} \quad , \quad (25) \]

where the weighted sum of the waiting times in the numerator is given by (24). This approximation can be used for general polling systems with exhaustive service. It has several interesting properties, e.g., it is exact for symmetric systems, and it fits into the pseudo conservation law.

If now the load is increased, some of the queues will enter state '0'. For the queues which are still in state '1', we propose to calculate \( w_k \) again according to (23), but \( c_k \) will now be estimated by providing lower and upper bounds. Certainly, \( c_k^{(e)} \geq c^* \) since the variance of the cycle time is greater than or equal to zero. On the other hand, increasing the load until the next value \( c^* \) of the set \( \{c_{k0}, c_{k1} \} \) is reached will yield a very small change of the cycle time variance, because all the queues in state '0' try to make full use of their transmission window and hence always produce the maximum possible cycle time. For the same reason, the variance of \( c^* \) will be very small. We therefore use \( c^{(e)} = c^* \) as an upper bound for \( c_k^{(e)} \). In practice, \( c_k^{(e)} \) is very close to \( c^* \), which can therefore also be used as an approximation.

Finally, queues under overload must be considered. The mean waiting time \( w_k = w_{k,o} \) of those packets that are not lost is determined in a completely different manner, which is also valid for non-Markovian arrivals. Since these queues are overloaded, their queue length reaches the maximum \( m_k \) at some instant, and the packets arriving afterwards are lost. When the token arrives, the queue will be served for the service time \( a_k \) with the service rate \( c_k = 1/b_k \), which corresponds to the full bandwidth available for PS traffic. In the meantime, new packets arrive at rate \( \lambda_k \). After token departure, the queue is filled at rate \( \lambda_k \) until again the maximum queue length \( m_k \) is reached. This pattern is repeated periodically with a frequency of \( 1/c \) and leads to the following approximate expression for the long term average of the queue length \( \Omega_k \)

\[ \Omega_k = \begin{cases} m_k - \frac{(1 - \rho_k)a_k^2}{2\rho_kb_k} & \text{if } \rho_k < 1 \\ m_k & \text{if } \rho_k \geq 1 \end{cases} \quad , \quad (26) \]

In order to apply Little's law, we need the mean arrival rate into the queue, which is equal to the average service rate \( \lambda_k = c_k \cdot a_k/c \). Hence,

\[ w_k = w_{k,o} = \frac{\Omega_k}{\lambda_k} = \begin{cases} \frac{m_k b_k c}{a_k} - \frac{(1 - \rho_k)a_k}{2\rho_k} & \text{if } \rho_k < 1 \\ \frac{m_k b_k c}{a_k} & \text{if } \rho_k \geq 1 \end{cases} \quad . \quad (27) \]

5 Examples

In this section, we study three examples of the analysis derived in the previous two sections. In case A, a completely symmetric FDDI system with \( N = 25 \) stations is considered. The ring length of 100 km and 64 bit station latency lead to \( c_0 = 524.5 \, \mu s \). For the TTRT, a value of 10 ms is assumed. Every station in this example has one queue asynchronous traffic, which can store up to \( m_i = 50 \) packets. Isochronous and synchronous traffic are not considered. A mean information part length of \( L_i = 1024 \) bit is assumed, and hence the mean packet service time including a packet overhead of \( L_o = 224 \) bit is \( b_i = b'_i = 12.48 \, \mu s \). With a negative exponentially distributed service time of the information part and a constant packet overhead, \( b_i^{(3)} = 260.608 \, (\mu s)^2 \) is obtained. The total amount of offered traffic \( \rho = \sum_{i=1}^{N} \rho_i \) is varied from 0 to 2.

Case B is based on the same set of system parameters, but the offered traffic is now assumed to be asymmetric. Station 1, which can, e.g., stand for a gateway station, now contributes \( \rho_1 = \rho/2 \), and the remaining traffic is distributed equally among stations 2 through 25.

![Figure 1: Mean Cycle Time](image-url)

Fig. 1 shows the mean cycle time of the approximation as a function of \( \rho \). The corresponding throughput is given in Fig. 2. Case A follows directly equations (20), (21). At \( \rho = \rho_{\max} = 0.776 \) the system reaches overload with constant mean cycle time and throughput values. It can be seen clearly, according to [16, 10] the mean cycle time is always smaller than TTRT. In Case B, station 1 reaches overload earlier than the other stations, because \( c_{2,1} \) is smaller due to the higher load \( \rho_1 \). This leads to a throughput reduction for station 1 for \( \rho > 0.758 \), whereas stations 2 through 25 do not experience packet loss up to \( \rho = 1.49 \). Beyond \( \rho = 1.49 \), all stations receive the same throughput regardless of their offered traffic, which means very high loss especially for...
This is the reason why we introduced another case C, where the system and traffic parameters are the same as in case B, except that station 1 is given a higher priority in order to increase its guaranteed throughput. Under overload, we get using (15)
\[ C = T_{PRI,1} + (N - 1)T_{PRI,2} + c_0 \]  
(28)
\[ \gamma_{n,i} = \begin{cases} 
   NT_{PRI,1} - (N - 1)T_{PRI,2} - c_0 & \text{if } i = 1 \vspace{0.5em} \\
   2T_{PRI,2} - T_{PRI,1} - c_0 & \text{if } 2 \leq i \leq N 
\end{cases} \]  
(29)
In order to obtain \( \gamma_1 = \sum_{i=2}^{N} \gamma_i \), we choose \( T_{PRI,1} = TTRT \) and
\[ T_{PRI,i} = \frac{(2N - 1)T_{PRI,1} + (N - 2)c_0}{3(N - 1)} = 6.97ms \]  
(30)
for \( i = 2, 3, \ldots, N \).

As can be seen in Fig. 2, the throughput of station 1 under overload is now adapted to its percentage of offered traffic. However, the total information throughput is now decreased slightly from 77.6 to 75.8 Mbps. Since
\[ \gamma_{max} = \frac{c_{max} - c_0}{c_{max}} \]  
(31)
this can also be derived from the maximum cycle time in Fig. 2, which is now 6.84 ms instead of 9.64 ms.

Additionally, some simulation results are provided in order to validate the results of the approximation. The 95% confidence intervals are always smaller than the symbols. For the simulation, Markov arrivals are assumed. This leads to a very good accuracy of the approximation. However, the accuracy deteriorates if the coefficient of variation of the interarrival times is very large or if correlations are introduced (the impact of various source models is studied in a companion paper [15]).

In Fig. 3 and 4, the mean waiting times are depicted. Because the waiting time approximations are different for the individual station states, the waiting times are unsteady at the offered traffic values where state transitions occur. The approximation for light load is again very good, whereas upper bounds are obtained under overload.

It can be seen clearly, that under overload the small service rate of station 1 in case B leads to extremely large delays, which are given by \( m_1 x / \gamma_1 \) according to (17), (27). In case C, the mean waiting time of station 1 is decreased significantly by adapting \( \gamma_1 \) to the actual offered load, whereas \( m_1 \) remains constant. In contrast to this, the increase of the waiting times of sta-
tions 2 through 25 caused by their lowered priority level remains tolerable. Therefore, besides tuning the station throughput, varying the priority thresholds $T_{pr}$ can also be used to optimize the delays.

6 Conclusion

In this paper, a general FDDI system with isochronous, synchronous and asynchronous traffic has been considered. We derived simple approximate expressions for the mean cycle time and the throughput and mean waiting time of each queue in the system for any traffic scenario. Some examples show that though the analysis is approximate, its accuracy is still very good. It can therefore be used as a method for the performance evaluation as well as dimensioning of an FDDI system.

References


