Properties of the Timed Token Protocol

Biao Chen and Wei Zhao
Department of Computer Science
Texas A&M University
College Station, TX 77843-3112

October, 1992

Technical Report 92-038

1 Introduction

Timed token medium access control protocol has been incorporated into many network standards, including FDDI, HSDB/HSRB and SAFENET. Many embedded real-time applications use them as backbone networks. The basic idea of the timed token protocol was presented by Grow [3] and then studied by Ulm [9], considering only two classes: synchronous and asynchronous messages. When a network is initialized, the stations decide the value of a target token rotation time (TTRT). Moreover, the maximum time a station is allowed to transmit synchronous messages (i.e. synchronous capacity allocated to a station) is set so that the available bandwidth is not exceeded. A station receiving the token is allowed to transmit synchronous frames up to the maximum time, and then asynchronous message frames can be transmitted only if the time elapsed from the previous token departure from the same station is less than TTRT.

This protocol is suitable for real-time applications not only because of its use in high bandwidth networks but also due to its special timing properties. The upper bounds on the maximum and average token rotation time have been studied by Johnson and Sevick [8]. A formal proof that the timing requirements of the FDDI token ring protocol are satisfied is provided in [5]. It was shown that the token rotation time can not exceed twice the value of TTRT, while the average rotation time is not greater than TTRT.

Although the prerequisite of "bounded token rotation time" property is indispensable, it is not sufficient for guaranteeing message deadlines. A node with inadequate synchronous capacity may be unable to complete the transmission of a synchronous message before its deadline. On the other hand, allocating excess amounts of synchronous capacities to the nodes could increase the token rotation time, which may also cause message deadlines to be missed. Therefore, guaranteeing message deadlines, bandwidth allocation and token rotation time properties are all closely related in the timed token medium access control protocol. In this report we will derive some properties of token rotation time and then use these properties to yield a close formula for the minimum total synchronous message transmission time a node can have during any message period.

2 Background and Definitions

2.1 Network Model

We consider the network topology as consisting of m nodes connected by point-to-point links forming a circle i.e., the token ring. A special bit pattern called the *token* circulates around ring (from node i to nodes $i+1$, $i+2$, ...until node m, then to nodes $1, 2, \ldots$), helping to determine which node should send a frame of message among the contending nodes.

We denote the latency between a node i and its upstream neighbor[1] by θ_i. This delay includes the node bit delay, the node latency buffer delay, the media propagation delay, etc. The sum total of all such latencies in the ring is known as the *ring latency* Θ, i.e., $\sum_{i=1}^{n} \theta_i = \Theta$. Thus, the ring latency Θ denotes the token walk time around the ring when none of the nodes in the network disturb it.

2.2 Message Model

Messages generated in the system at run time may be classified as either *synchronous messages* or *asynchronous messages*. We assume that there are n streams of synchronous messages, $S_1, S_2, \ldots, S_n$ in the system which form a synchronous *message set, M,* i.e.,

$$M = \{S_1, S_2, \ldots, S_n\}.$$

The characteristics of messages are as follows:

1. Synchronous messages are *periodic*, i.e., messages in a synchronous message stream have a constant inter-arrival time. We denote P_i to be the period length of stream S_i ($i = 1, 2, \ldots, n$).
2. The *deadline* of a synchronous message is the end of the period in which it arrives. That is, if a message in stream S_i arrives at time t, then its deadline is at time $t + P_i$.
3. Messages are independent in that message arrivals do not depend on the initiation or the completion of transmission requests for other messages.
4. The *length* of each message in stream S_i is C_i which is the maximum amount of time needed to transmit this message.
5. Asynchronous messages are non-periodic and do not have a hard real-time deadline requirement.

In the following discussion we assume that there is one stream of synchronous messages on each node (i.e., $m = n$). Since under certain transformation, an arbitrary token ring network where a node may have zero, one, or more streams of synchronous messages can be transformed into a logically equivalent network with one stream of synchronous messages per node. Hence, this assumption of one stream per node simplifies the analysis without loss of generality. We also assume that the network is free from hardware or software failures.

[1] The upstream neighbor of node i is node $i-1$ if $i > 1$ else node m if $i = 1$.

2.3 Timed token medium access control protocol

2.3.1 Protocol parameters

The timed token protocol uses the following parameters and variables for its operation.

1. *Target Token Rotation Time (TTRT).* When the network is initialized, the value of the $TTRT$ is determined, which gives the expected value of the token rotation time. It is selected to be sufficiently small to support the response time requirements of the messages at all the nodes in the network. Since the time elapsed between two consecutive visits of the token at a node can be as much as $2 \cdot TTRT$ [5], a node may not be able to transmit any message in this interval.

   Recall that the synchronous messages have their deadlines as the end of their periods. Hence, in order to meet message deadlines it is necessary to select $TTRT$ such that, for $1 \leq i \leq n$,

   $$TTRT \quad \leq \quad \frac{P_i}{2} \tag{1}$$

   where P_i is the period of synchronous message stream S_i. We assume that (1) holds throughout this paper. Any P_i may therefore be represented as a linear function of $TTRT$. That is,

   $$P_i = m_i \cdot TTRT - \delta_i, \tag{2}$$

   where $m_i = \lceil \frac{P_i}{TTRT} \rceil \geq 2$ and $\delta_i = \lceil \frac{P_i}{TTRT} \rceil \cdot TTRT - P_i$. The above expression for P_i has been introduced as it will be useful in several proofs encountered later on. Note that if $m_i = 2$, then $\delta_i = 0$ and if $m_i \geq 3$ then $0 \leq \delta_i < TTRT$.

2. *Synchronous capacity of node i (H_i).* This parameter represents the maximum time for which a station is permitted to transmit synchronous messages every time the station receives the token. Note that each station can be assigned a different H_i value.[2] This paper will deal with the issue of appropriate allocation of these H_i values.

3. *Token Rotation Timer of node i (TRT_i).* This counter is initialized to equal TTRT, and counts down until it expires (i.e., $TRT_i = 0$) or until the token is received and the time elapsed since the previous token departure is less than TTRT. In either situation, the TRT_i is reinitialized to TTRT. After being reset, it continues the subsequent counting down cycles in the same manner as above.

4. *Token Holding Timer of node i (THT_i).* This (down) counter is used to control the amount of time for which the node i can transmit asynchronous messages.

5. *Late Counter of node i (LC_i).* This counter is used to record the number of times that TRT_i has expired since the last token arrival at node i.

[2]In FDDI stations, the assignment of H_i to station i is a function of the station management entity of the FDDI protocol.

2.3.2 Protocol operation and constraint

At ring initialization, the following parameters are initialized at all nodes:

1. $THT_i \leftarrow 0$;
2. $LC_i \leftarrow 0$;
3. $TRT_i \leftarrow TTRT$.

The TRT_i counter always counts down. When it reaches zero, the following actions take place:

1. $TRT_i \leftarrow TTRT$;
2. $LC_i \leftarrow LC_i + 1$.

The TRT_i then begins the counting down process again with LC_i being incremented by one at every expiration of TRT_i. Normally, if LC_i exceeds one, the ring recovery process is initiated [4].

A token is considered to arrive *early* at node i if $LC_i = 0$ at the time of its arrival. The token is *late* if $LC_i > 0$.

When the token arrives *early* at node i, the following actions take place:

1. $THT_i \leftarrow TRT_i$;
2. $TRT_i \leftarrow TTRT$;
3. Synchronous frames (if any) can then be transmitted for a maximum time of H_i (i.e., the synchronous capacity at node i);
4. After transmitting synchronous frames (if any), the station enables counter THT_i (i.e., it starts counting down). The station may then transmit asynchronous frames as long as $THT_i > 0$ and $TRT_i > 0$.

When the token arrives *late* at node i, the following actions take place:

1. $LC_i \leftarrow 0$;
2. TRT_i continues to count down towards expiration. Note that it is *not* reset to $TTRT$ as in the case when the token is early;
3. Node i can transmit synchronous frames for a maximum time of H_i;
4. No asynchronous frame will be transmitted.

Figure 1 shows an example of how TRT_i and LC_i (at some node i) vary with time t. At point B in the figure, the node receives the token early. At point F, the token is received late. Synchronous messages are transmitted in both cases, but asynchronous messages are transmitted only when the token arrives early.

Theoretically, the total available time to transmit synchronous messages, during one complete traversal of the token around the ring, can be as much as TTRT. However, factors such as ring latency Θ and other protocol/network dependent overheads reduce the total available time to transmit the synchronous messages. We denote the portion of $TTRT$ unavailable for transmitting synchronous messages by τ. That is, $\tau = \Theta + \Delta$ where Δ represents the protocol dependent overheads.[3] We define the ratio of τ to $TTRT$ to be α. The usable ring utilization available for synchronous messages would therefore be $(1 - \alpha)$ [9].

Thus, a protocol constraint on the allocation of synchronous capacities is that *the sum total of the synchronous capacities allocated to all nodes in the ring should not be greater than the available portion of the Target Token Rotation Time (TTRT)*, i.e.,

$$\sum_{i=1}^{n} H_i \leq TTRT - \tau. \tag{3}$$

In the rest of this paper, we assume that (3) holds.

3 Timing Properties and Minimum Total Transmission Time

In this section we prove the tight upper bound on the time elapsed between token's *l-th* and $(l+v)$-*th* visit to a node. Based on this we derive the formula for the minimum transmission time a node can get during any message period.

First we list some definitions of a few terms and lemmas which will be used in the following proofs.

3.1 Definitions of Terms

- $t_i(l)$, $(l = 1, 2, \ldots)$. It is the time when the token makes its *l-th* visit to node i.
- $R_i(l)$, $(l = 1, 2, \ldots)$. It is defined as follows:

$$R_i(l) = \begin{cases} t_i(l) + TTRT, & \text{if the token is early on its } l\text{-}th \text{ visit to node } i; \\ R_i(l-1) + TTRT, & \text{otherwise.} \end{cases}$$

  That is, $R_i(l)$ indicates the 'next expected arrival time' of the token at node i after the token's *l-th* visit. If the token is late on its $(l+1)$-*th* visit to node i, then $R_i(l)$ will be the time at which TRT_i expires and is reset to TTRT. Note that the definitions of $R_i(l)$ and $t_i(l)$ imply that

$$R_i(l) - t_i(l) \leq TTRT. \tag{4}$$

- The amount of time left, before the initiation of the ring recovery process by node i, can be expressed as a function of two parameters at that node — the TRT_i and LC_i. In order to simplify our proofs, we define a single parameter TRT_i', capturing the values of both TRT_i and LC_i within it, to indicate the amount of time left before the initiation of ring recovery process by node i. TRT_i' is formally defined as follows:

$$TRT_i' = TRT_i + (1 - LC_i) \cdot TTRT. \tag{5}$$

[3] For example, according to the FDDI standard, the protocol dependent overheads include the token transmission time, asynchronous overrun, etc. Refer to [1] for details.

Given the fact that $0 \leq TRT_i \leq TTRT$ and $0 \leq LC_i \leq 1$, it is clear that

$$0 \leq TRT_i' \leq 2 \cdot TTRT. \tag{6}$$

The physical meaning of TRT_i' is that when $TRT_i' > TTRT$, TRT_i has not expired since the last token arrival. A token arriving at this instant would be early. When $TRT_i' \leq TTRT$, $LC_i = 1$. Hence, TRT_i has expired once since the last token arrival. In either case, the amount of time left before node i initiates the ring recovery process is TRT_i'. In the event that TRT_i' becomes zero, the ring recovery process will be initiated.

3.2 Timing Properties

In the proofs of the lemmas and theorems that follow, $TRT_i(t)$, $THT_i(t)$, $LC_i(t)$, $TRT_i'(t)$ represent the values of TRT_i, THT_i, LC_i and TRT_i' at time t.

We start with the time bound between token's l-th visit and $(l+v)$-th visit.

LEMMA 3.1 *For any integers $l > 0$, $v > 0$ and any node i $(1 \leq i \leq n)$,*

$$R_i(l+v) - R_i(l) \quad \leq \quad v \cdot TTRT. \tag{7}$$

The lemma can be easily proved by an induction argument. Whether the equality can be achieved depends on the bandwidth allocation.

LEMMA 3.2 *(Johnson's Lemma [5]) After ring initialization, the TRT' values of all operational stations will be monotonically increasing in the downlink direction, up to and including the station which last received the token.*[4]

The reader is referred to [5] for the proof of the above lemma.

The next lemma gives an upper bound on the token's lateness.

LEMMA 3.3 *For any $l > 0$ and any node j, if the token is late on its $(l+1)$-th visit at node j, then*

$$\begin{aligned} t_j(l+1) &\leq R_j(l) + \sum_{h \neq j} H_h + \tau \\ &\leq R_j(l) + TTRT - H_j. \end{aligned}$$

Proof: Let us first define the phrase "token is at node i" to mean that the token is being held by node i or is on its way to node i from its upstream neighbor (node $i-1$ if $i > 1$, else node n if $i = 1$).

Recall that $TRT_i'(t)$ is defined as

$$TRT_i'(t) \quad = \quad TRT_i(t) + (1 - LC_i(t)) \cdot TTRT.$$

If the token is late on its $(l+1)$-th visit to node j, the token must be at some node i at time $R_j(l)$. Assume that the token arrives at node i at time T $(T \leq R_j(l))$. We have two cases to consider.

[4] This result is known as the 'TRT alignment' in [5]. By the FDDI MAC standard, the ring initialization phase aligns the TRT' values. That is, the TRT' values monotonically increase in the downlink direction of the ring. The proof of Johnson's Lemma assumes that this alignment holds during normal ring operation and so do we.

Case 1: The token arrives late at node i. In this case node i will only transmit its synchronous messages for at most H_i time. Hence the token will leave node i no later than $T + H_i + \theta_i \leq R_j(l) + H_i + \theta_i$.

Case 2: The token arrives early at node i. Therefore, $LC_i(T) = LC_j(T) = 0$. By Johnson's Lemma (Lemma 3.2) we have

$$TRT_i'(T) - TTRT \quad \leq \quad TRT_j'(T) - TTRT.$$

From (5), we get

$$TRT_i(T) \leq TRT_j(T).$$

Thus,

$$THT_i(T) \;=\; TRT_i(T) \quad \leq \quad TRT_j(T) \;=\; R_j(l) - T.$$

Hence, node i can transmit asynchronous messages for at most $R_j(l) - T$ time and transmit synchronous messages for at most H_i time. In this case too, the token will leave node i before

$$T + (R_j(l) - T + H_i) + \theta_i \;=\; R_j(l) + H_i + \theta_i.$$

That is, if the token is at node i when TRT_j expires, then the token will leave node i no later than $R_j(l) + H_i + \theta_i$.

Now consider the nodes on the way from node i to node j. Let them be labeled as $n_1, n_2, \ldots, n_k$. According to Johnson's Lemma, the token will be late on its visit to each of these k nodes on the way to node j. Hence, these nodes will transmit their synchronous messages only. That is, the token will arrive at node j no later than

$$\begin{aligned} R_j(l) + H_i + \theta_i + \sum_{h=1}^{k}(H_{n_h} + \theta_{n_h}) + \theta_j + \Delta \quad &\leq \quad R_j(l) + \sum_{h \neq j} H_h + \sum_{h=1}^{n} \theta_h + \Delta \\ &= \quad R_j(l) + \sum_{h \neq j} H_h + \tau \end{aligned}$$

where Δ represents the protocol dependent overheads. By (3), we have

$$t_j(l+1) \leq R_j(l) + \sum_{h \neq j} H_h + \tau \leq R_j(l) + TTRT - H_j.$$

Q.E.D.

In [5] Johnson proves that $2TTRT$ is an upper bound on token rotation time. With the above lemmas we can give a tighter bound on the token rotation time.

THEOREM 3.1 *(Johnson and Sevcik's Theorem [5, 8]) For any integer $l > 0$ and any node j $(1 \leq j \leq n)$,*

$$t_j(l+1) - t_j(l) \quad \leq \quad TTRT + \sum_{h \neq j} H_h + \tau.$$

Proof: If the token is not late at its $(l+1)$-th visit to node j, then

$$t_j(l+1) - t_j(l) \;\leq\; TTRT \;\;\leq\;\; TTRT + \sum_{h\neq j} H_h + \tau$$

Otherwise, from Lemma 3.3 we have

$$\begin{aligned}
t_j(l+1) - t_j(l) &\leq R_j(l) + \sum_{h\neq j} H_h + \tau - t_j(l) \\
&\leq \sum_{h\neq j} H_h + \tau + (R_j(l) - t_j(l)) \\
&\leq \sum_{h\neq j} H_h + \tau + TTRT \qquad \text{(by (4))}.
\end{aligned}$$

Q.E.D.

This simply says that the maximum time that can elapse between two consecutive token arrivals at node j is bounded by $2 \cdot TTRT - H_j$. This result was first proved in [5].

The next theorem provides an upper bound on the time elapsed between token's l-th visit and $(l+v)$-th visit.

THEOREM 3.2 *(Generalized Johnson and Sevcik's Theorem) For any integer $l > 0$, $v > 1$ and any node j ($1 \leq j \leq n$),*

$$t_j(l+v-1) - t_j(l) \;\;\leq\;\; (v-1)\cdot TTRT + \sum_{h\neq j} H_h + \tau \tag{8}$$

Furthermore, equality holds in the worst case.

Proof: We prove the theorem by induction on v. For $v = 2$, by Theorem 3.1 we have

$$t_j(l+1) - t_j(l) \leq TTRT + \sum_{h\neq j} H_h + \tau. \tag{9}$$

Hence, the theorem holds for $v = 2$.

Assume that for $v = k$, (8) holds, i.e.,

$$t_j(l+k-1) - t_j(l) \leq (k-1)\cdot TTRT + \sum_{h\neq j} H_h + \tau. \tag{10}$$

Now we consider for $v = k+1$. We have two cases:

Case 1: The token arrives early on its $\mathbf{v'}$-th visit to node j ($\mathbf{l < v' < v}$). That is,

$$R_j(v'-1) \geq t_j(v'). \tag{11}$$

Hence,

$$\begin{aligned}
t_j(v') - t_j(l) &= (t_j(v') - R_j(l)) + (R_j(l) - t_j(l)) \\
&\leq (R_j(v'-1) - R_j(l)) + (R_j(l) - t_j(l)) \qquad \text{(by (11))}.
\end{aligned} \tag{12}$$

By Lemma 3.1 and (4), we have

$$t_j(v') - t_j(l) \;\leq\; (v'-1-l)\cdot TTRT + TTRT \;=\; (v'-l)\cdot TTRT. \tag{13}$$

Now

$$t_j(l+(k+1)-1) - t_j(v') \;=\; t_j(v'+(l-v'+k+1)-1) - t_j(v'). \tag{14}$$

By the induction hypothesis (10) and (14), we have

$$t_j(l+(k+1)-1) - t_j(v') \;\leq\; (l-v'+k)\cdot TTRT + \sum_{H\neq j} H_h + \tau. \tag{15}$$

Adding (13) and (15) on both sides, we have

$$\begin{aligned} t_j(l+(k+1)-1) - t_j(l) \;&\leq\; (v'-l)\cdot TTRT + (l-v'+k)\cdot TTRT + \sum_{h\neq j} H_h + \tau \\ &=\; k\cdot TTRT + \sum_{h\neq j} H_h + \tau. \end{aligned}$$

Case 2: The token is always late at node j between the $(l+1)$-th **visit and the** $(l+k)$-th **visit inclusive.**

Because the token is late, from Lemma 3.3, we have

$$t_j(l+k) \;=\; t_j(l+(k+1)-1) \;\leq\; R_j(l+k-1) + \sum_{h\neq j} H_h + \tau. \tag{16}$$

Therefore, by (4) and (16),

$$\begin{aligned} t_j(l+(k+1)-1) - t_j(l) \;&=\; (t_j(l+(k+1)-1) - R_j(l)) + (R_j(l) - t_j(l)) \\ &\leq\; (R_j(l+k-1) + \sum_{h\neq j} H_h + \tau - R_j(l)) + TTRT \\ &=\; R_j(l+k-1) - R_j(l) + TTRT + \sum_{h\neq j} H_h + \tau \\ &=\; (k-1)\cdot TTRT + TTRT + \sum_{h\neq j} H_h + \tau \qquad \text{(by Lemma 3.1)} \\ &=\; k\cdot TTRT + \sum_{h\neq j} H_h + \tau. \end{aligned}$$

Thus, the first part of the theorem is proved.

We now show that the equality holds in the worst case by giving an example. We consider a ring with $n+2$ nodes among which n are synchronous nodes which transmit only synchronous messages. We denote them by $node\_1, node\_2, node\_3 \ldots, node\_n$ in their order in the ring. $node\_a, node\_b$ are the asynchronous node which only transmits asynchronous messages. They are between $node\_n$ and $node\_1$, $node\_a$ follows by $node\_b$. For the simplicity, we assume that the token walking time and other overhead from $node\_n$ to $node\_1$ through $node\_a$ and $node\_b$ is 0. It is easy to modify the proof to suit the general case. Suppose at $t_n(0) = t_0$ token reaches $node\_n$ and none of the nodes has any message to send at that time. Therefore, the token will return to *node_n* at time $t_n(1) = t_0 + \tau$. Suppose the synchronous message arrives at *node_n* at time $t = t_0 + \tau + \epsilon$ where $\epsilon > 0$ is very small. We see *node_n* will pass token to *node_a* immediately. Since token arrives early on *node_a*, *node_a* can send its asynchronous message up to $TTRT - \tau$. Suppose *node_a* only sends $TTRT - \tau - \delta$ where $\delta > 0$ is very small and passes the token to *node_b*. The token is also early on the *node_b*. Hence the TRT on *node_b* is reset at $t = t_0 + TTRT - \delta$ and *node_b* can send asynchronous message up to δ. According to Lemma 3.2 when token leaves *node_b* all the LC counters on *node_1*, *node_2*, ..., *node_n* have been set to 1. Therefore, the token will be late on the way to *node_n*. *node_1*, *node_2*, ..., *node_n* can only send synchronous message up to their allocated synchronous capacity. So the token will return to *node_n* at time $t_n(2) = t_0 + TTRT + \sum_{i=1}^{n-1} H_i + \tau$. This time, *node_n* will use its bandwidth to send synchronous message up to its allocated synchronous capacity. The next node, *node_a*, which is a asynchronous node, gives up transmission since the token is late. Hence the token will be early on *node_b* at $t = t_0 + TTRT + \sum_{i=1}^{n} H_i + \tau$. Note that *node_b* can send asynchronous message up to $TTRT - \delta - (\sum_{i=1}^{n} H_i + \tau)$ which makes the token late on the way back to *node_n* again. Therefore, the token will return to *node_n* at time $t_n(3) = t_0 + 2TTRT + \sum_{i=1}^{n-1} H_i + \tau$. Generalize and apply this argument to the general case we have that for the token's *v-th* visit to *node_n*

$$t_n(v) - t_n(1) \quad = \quad (v-1) \cdot TTRT + \sum_{i=1}^{n-1} H_i + \tau$$

This conclude the proof of the theorem. Q.E.D.

If we let $\epsilon \to 0^+$ in the above example then the synchronous message arrives at time $t \to t_n(1)^+$. This observation gives the following result.

COROLLARY 3.1 *Assume that at time t, a synchronous message arrives at node i $(1 \leq i \leq n)$. Let $t_i(l+k)$ be the time of token's k-th arrival after t. Then*

$$t_i(l+k) - t \quad \leq \quad k \cdot TTRT + \sum_{j \neq i} H_j + \tau$$

and equality holds in the worst case.

With Corollary 3.1, we can now derive the minimum total transmission time for each node to transmit synchronous message within any time interval, as stated below.

COROLLARY 3.2 *Let $X_i(t, t+I, \vec{H})$ be the minimum total transmission time available for node i to transmit its synchronous message within the time interval $(t, t+I)$ under bandwidth allocation $\vec{H} = (H_1, H_2, \ldots, H_n)^T$ then*

$$X_i(t, t+I, \vec{H}) = \lfloor \frac{I}{TTRT} - 1 \rfloor \cdot H_i + \max(0, \min(r_i - \tau - \sum_{j=1,\ldots,n, j \neq i} H_j, H_i)). \qquad (17)$$

where $r_i = I - \lfloor \frac{I}{TTRT} \rfloor \cdot TTRT$.

Proof: Let $t_i(l+k)$ be the time of the token's *k-th* arrival at node i after the message's arrival at time t. I can be represented as

$$I \quad = \quad m_i \cdot TTRT + r_i,$$

where $m_i = \lfloor \frac{I}{TTRT} \rfloor$ and r_i is defined in the corollary. By Corollary 3.1,

$$t_i(l+v-1) - t \leq (v-1) \cdot TTRT + \sum_{j=1,\ldots,n,j\neq i} H_j + \tau$$

Since

$$\begin{aligned} t_i(l+m_i-1) - t &\leq (m_i-1) \cdot TTRT + \sum_{j=1,\ldots,n,j\neq i} H_j + \tau \\ &\leq m_i \cdot TTRT - H_i \\ &\leq I - H_i \end{aligned}$$

That is,

$$t + I \geq t_i(l+m_i-1) + H_i.$$

Hence, by the end of the time interval (i.e., $t+I$) in the worst case the token will have made $m_i - 1$ visits to node i since time t. In each of these visits, node i can transmit its synchronous message for the allocated synchronous capacity H_i.

On the other hand, by Corollary 3.1, in the worst case

$$t_i(l+m_i) - t = m_i \cdot TTRT + \sum_{j=1,\ldots,n,j\neq i} H_j + \tau$$

which implies

$$\begin{aligned} t &= t_i(l+m_i) - m_i \cdot TTRT - \sum_{j=1,\ldots,n,j\neq n} H_j - \tau \\ &= t_i(l+m_i) - I + (r_i - \sum_{j=1,\ldots,n,j\neq i} H_j - \tau) \end{aligned}$$

Therefore,

$$t + I = t_i(l+m_i) + (r_i - \sum_{j=1,\ldots,n,j\neq i} H_j - \tau) \tag{18}$$

Now we have two cases to consider:

Case 1: $\mathbf{0 \leq r_i \leq \sum_{j=1,\ldots,n,j\neq i} H_j + \tau}$**.** By (18), within time interval $(t, t+I)$ the token only makes $m_i - 1$ visits to node i. According to the above discussion each time node i can transmit synchronous message upto H_i. Consequently, the total amount of the time for node i to transmit this synchronous message will be

$$\begin{aligned} X_i(t, t+I, \vec{H}) &= (m_i - 1) \cdot H_i \\ &= \lfloor \frac{I}{TTRT} - 1 \rfloor \cdot H_i. \end{aligned}$$

This agrees with (17) since in this case $\max(0, \min(r_i - \tau - \sum_{j=1,\ldots,n,j\neq i} H_j, H_i)) = 0$.

Case 2: $\mathbf{TTRT \geq r_i \geq \sum_{j=1,\ldots,n,j\neq i} H_j + \tau}$**.** By (18), the token will make m_i visits to node i within time interval $(t, t+I)$. Again it can transmit upto H_i in the first $m_i - 1$ visits. But for the m_i-th visit it can only transmit $\min(r_i - \tau - \sum_{j=1,\ldots,n,j\neq i} H_j, H_i)$. This also agrees with (17) since $r_i - \tau - \sum_{j=1,\ldots,n,j\neq i} H_j \geq 0$. Q.E.D.

References

[1] *FDDI Token ring media access control (MAC).* ANSI Standard X3.139, 1987.

[2] *FDDI Token ring station management*, draft proposed — ANSI Standard X3T9.5 rev. 5, May 10, 1989.

[3] R. M. Grow, "A timed token protocol for local area networks," *Proc. Electro/82, Token Access Protocols*, May 1982.

[4] M. J. Johnson, "Reliability mechanisms of the FDDI high bandwidth token ring protocol," *Computer Networks ISDN Systems*, Vol. II., No. 2, pp. 121-131, 1986.

[5] M. J. Johnson, "Proof that timing requirements of the FDDI token ring protocols are satisfied," *IEEE Trans. Commun.* Vol. COM-35. No. 6, pp. 620-625, June 1987.

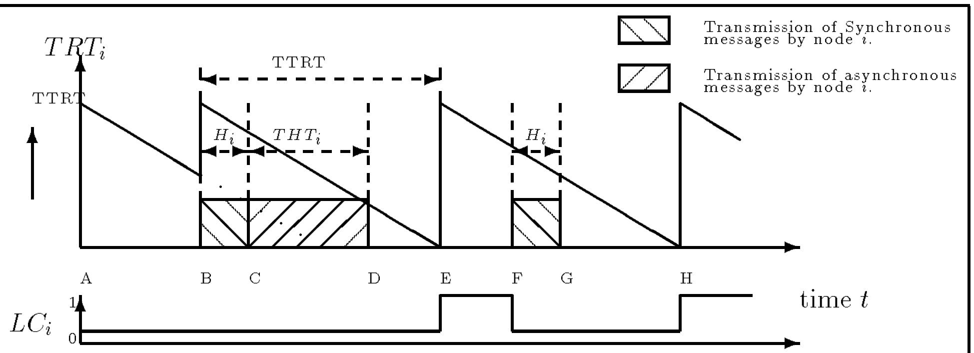
[6] F. E. Ross, "FDDI—A tutorial," *IEEE Commun. Mag.,* Vol. 24, No. 5, pp. 10-17, 1986.

[7] F. E. Ross, "An overview of FDDI: The fiber distributed data interface," *IEEE Journal on Sel. Areas in Comm.*, pp.1043-1051, Vol. 7, Sept. 1989.

[8] K. C. Sevcik and M. J. Johnson, "Cycle time properties of the FDDI token ring protocol," *IEEE trans. Software Eng.*, Vol. SE-13, No. 3, pp. 376-385, 1987

[9] J. N. Ulm, "A timed token ring local area network and its performance characteristics," *Proc. Conf. Local Computer Networks,* Feb. 1982, pp. 50-56.

[10] A. Valenzano, P. Montuschi, and L. Ciminiera, "Some properties of timed token medium access protocols," *IEEE Transactions on Software Engineering*, Vol. 16, No. 8, August 1990.



A: TRT_i is reset to TTRT. $LC_i \leftarrow 0$. Token is released.

B: Token is received early. $THT_i \leftarrow TRT_i$. $TRT_i \leftarrow TTRT$.

C: Synchronous messages transmitted in interval B-C. THT_i starts count down.

D: Asynchronous messages transmitted in interval C-D. Token released.

E: TRT_i expires and is reset to TTRT. $LC_i \leftarrow 1$.

F: Token received late since $LC_i = 1$. $LC_i \leftarrow 0$. Transmit synchronous frames only.

G: Allocated capacity for synchronous transmission expires. Token is released.

H: TRT_i expires again and is reset to TTRT. $LC_i \leftarrow 1$.

Figure 1: An Example of TRT_i and LC_i versus Time t