Mobile Real-Time Communications in FDDI Networks*

Yibin Yang Ten-Hwang Lai Ming-Tsan Liu
Department of Computer and Information Science
The Ohio State University
Columbus, Ohio 43210-1277
E-mail: {yyang, lai, liu}@cis.ohio-state.edu

Abstract

We propose an architecture of FDDI-based mobile networks and address issues that arise in providing real-time communication services on such networks. A wide range of problems concerning synchronous bandwidth management and quality of service guarantee are identified. To solve these problems, we present a dynamic bandwidth management scheme, a source handoff protocol and two approaches to handling destination handoffs. These schemes make handoffs transparent to mobile users; no degradation in quality of service will be observed during handoffs. The proposed solutions are compatible with the FDDI standards.

1 Introduction

Wireless information networks have recently become a topic of intense interests. These networks are intended to provide mobile users with access to network resources and services anywhere at any time. An architecture of wireless networks, *cellular packet switch*, is proposed in [7]. It is based on the micro-cell structure and uses DQDB metropolitan area networks as infrastructure to connect wireless interfaces. Network control functions, such as call processing, mobility management and wireless resource management, are distributed among several interfaces in DQDB networks.

Our research is in a different direction. It is about extending existing FDDI networks with wireless terminals. FDDI, standing for fiber distributed data interface, is one of the most popular metropolitan area networks. It has been widely deployed. According to [9], the growth rate of FDDI networks was 80% in 1993 and close to 100% in 1994. We believe that the FDDI will continue to be popular and a mobile network based on it will be of great market value.

An ideal mobile system should provide mobile users with all services available to static (non-mobile) users. One of such services, real-time communication service [6, 11], is of particular importance in today's high speed networks. It can be used to support many applications such as manufacturing systems management, remote monitoring, robotic control, voice and video transmission. In [1, 14], the synchronous transmission capacity of FDDI networks is shown to be capable of supporting real-time communication service. It is desirable to extend this service to mobile users. One important application of this extension is in robotic communications. Communications between robots multiply their capabilities and effectiveness [4]. In a manufacturing environment, by providing mobile real-time communications between autonomous robots, the productivity is expected to increase. However, supporting real-time communications in a mobile network is a non-trivial task because a mobile user may move around. To cope with this situation, we first introduce a new architecture of mobile networks, then identify problems to support real-time communications in this architecture, and finally, design protocols and schemes to overcome these problems.

The rest of this paper is organized as follows. Section 2 is a preliminary of FDDI networks and real-time communications. Section 3 presents an architecture of FDDI-based mobile networks and discusses problems of real-time communications in this architecture. A dynamic bandwidth management scheme is proposed in Section 4. Solutions to problems caused by the mobile source and destination of a real-time connection are given in Sections 5 and 6, respectively. Finally, some conclusions are drawn in Section 7.

2 FDDI Networks and Real-Time Communications

In this section, we first briefly review the medium access control protocol (MAC) and the station management protocol (SMT) of FDDI networks. Then we introduce the concept of real-time channel and its extension to FDDI networks.

An FDDI network consists of a number of stations connected as a ring. It employs a *timed token MAC protocol.* A special control packet, *token*, circulates around the ring. Only the station possessing the token can transmit data. Data are classified into two categories: *synchronous* and *asynchronous*. A station transmits data according to the following rules: (1) each station is allocated a certain amount of bandwidth for transmitting synchronous data, h_i denoting the bandwidth allocated to station i;

---

*Research reported herein was supported in part by the Performance Analysis Lab of The Ohio State University and by U.S. Army Research Office, under contract No. DAAL03-92-G-0184. The views, opinions, and/or findings contained in this paper are those of the authors and should not be construed as an official Department of the Army position, policy or decision.

0-8186-7216-1/95 $04.00 © 1995 IEEE

(2) each time a station i receives the token, it is allowed to send synchronous data for up to h_i units of time; (3) if the token arrives early, the station is allowed to send asynchronous data for an amount of time that equals the earliness of the token arrival; otherwise, it is not allowed to send asynchronous data. The readers are referred to [2, 9] for details.

There are some important time properties for the FDDI MAC protocol. During the ring initialization, all stations agree on a value called the target token rotation time (TTRT). Let Θ be the sum of latencies between the stations and Δ denote the time to transmit a maximum-size asynchronous message. It has been shown in [10] that if $\sum_i h_i \leq TTRT - \Theta - \Delta$, each station is guaranteed to see the token at least once in every $2TTRT$ time. This result is generalized in [5] to the following theorem:

Theorem 1 *If* $\sum_i h_i \leq TTRT - \Theta - \Delta$, *the time elapsed between any* n *consecutive token visits to station* i *is bounded by* $nTTRT - h_i$.

According to the station management protocol [3], the bandwidth management in FDDI networks is centralized and static. There is at least one synchronous bandwidth management process (BMP) in an FDDI network. For each FDDI station, there is a station management module. The bandwidth allocation is through a request/response frame exchange between the BMP and the SMT module of a station. When a station wants to allocate or release some synchronous bandwidth, it sends a resource allocation frame (RAF) request to the BMP. Upon receiving this request, the BMP sends back a RAF response indicating the success or failure of the request. For the bandwidth allocation, a station cannot increase its amount of bandwidth until a positive response is received from the BMP.

The concept of real-time channel was first proposed in [6]. It is a simplex virtual connection with the quality of service (QOS) guarantee. This concept is further extended in [14] to FDDI networks by the following definition.

Definition 1 A *real-time channel* in an FDDI network is characterized by a 5-tuple, $RC = (T, C, s, d, h)$, where s is a source station generating a sequence of packets; T is a lower bound on the interval between any two consecutive packets; C is the maximum packet length, measured by transmission time; d is the maximum delay on each packet (a packet must be transmitted in d units of time after its generation); and h is the amount of bandwidth allocated to s.

A real-time channel $RC = (T, C, s, d, h)$ is said to be *feasible* if all packets generated in this channel can be delivered within the delay constraint d. The feasibility of a real-time channel depends on the values of h, $TTRT$ and their relations to T, C, and d. Given $TTRT$, T, C, and d, [14] provides a scheme to calculate the bandwidth required for a real-time channel to be feasible.

3 FDDI-based Mobile Networks

We describe in this section an architecture of FDDI-based mobile networks and point out issues that must be solved before real-time communication service can be supported in these networks.

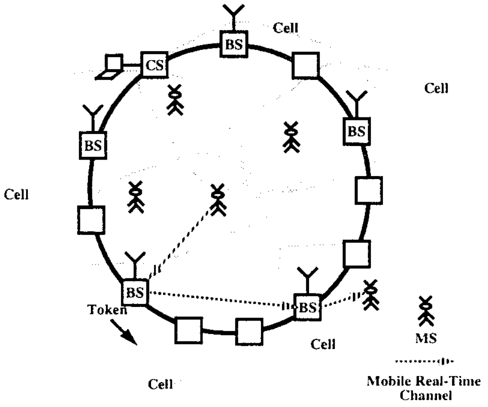

Figure 1: The architecture of FDDI-based mobile networks

3.1 Architecture for FDDI-based Mobile Networks

In this section, we propose an architecture of mobile systems based on FDDI networks. Its layout is depicted in Figure 1. Some components in this architecture are presented as follows. A mobile *control station* (CS) is a station that centrally manages the setups of mobile connections. A *mobile station* (MS) is a host that can move around while retaining its network connections without disruption. It is carried by a mobile user and serves as the user's interface to a mobile network. A *base station* (BS) is a wireless interface that connects to a wired network, in this case an FDDI network, as well as to the MSs within its wireless transmission range. Each BS is associated with a *cell*, which is the geographical area covered by its wireless transmission. Two cells are *overlapped* if they have a common area. Within this common area, an MS can communicate with two BSs. At any time, each MS has only one *local* BS, which is its primary data exchange interface. However, it may be able to exchange data with other BSs for a short interval.

When an MS moves from the cell of one BS into the cell of another, it will changes its local BS. This phenomenon is known as *handoff*. During a handoff, the path of data flow will change. A handoff time, t_h, is associated with a handoff. All packets generated before t_h are transmitted from the old local BS and all those generated after t_h are transmitted from the new local BS. According to [8], there are three types of handoffs: *hard*, *seamless*, and *soft* handoffs. For a seamless or soft handoff, the MS can simultaneously connect with two base stations for a while. This time period is called *degradation interval* and can last as long as several seconds [13]. Since soft handoffs are employed in the U.S. CDMA standard (IS-92) and considered in the European UMTS standard, we will assume soft handoffs in this paper.

3.2 Mobile Real-Time Communications

We propose to support real-time communications in an FDDI-based mobile network via *mobile real-time chan-*

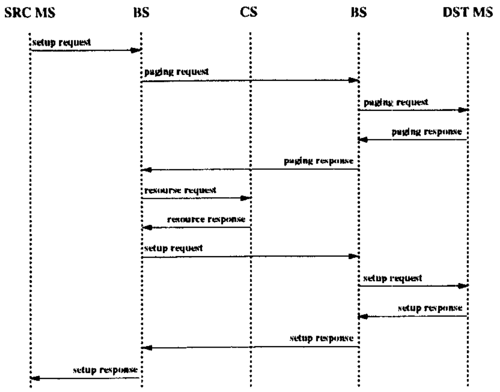

Figure 2: Real-time channel setup

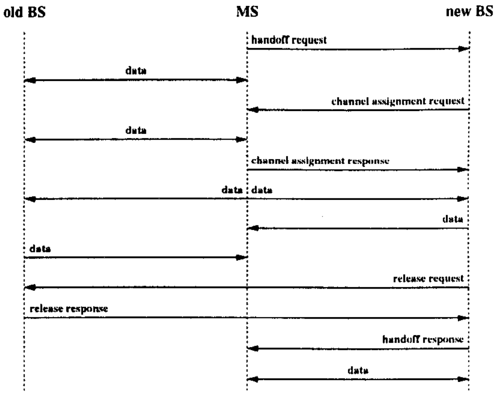

Figure 3: Real-time channel handoff

nels. A mobile real-time channel may have MSs as its source and destination. The data in a mobile real-time channel are first sent from the source MS to its local BS through a wireless channel. Then the BS transmits the data through the FDDI network to the local BS of the destination MS. Finally, this local BS delivers the data to the destination MS. By extending Definition 1, we give the following definition of mobile real-time channel.

Definition 2 A *mobile real-time channel* in an FDDI-based mobile network is characterized by a 5-tuple (T, C, s, d, h), which is defined as in Definition 1 except that 1) s is a mobile station, 2) the generation time or finish time of a packet is the time when its last bit reaches or leaves the local BS of s, respectively, and 3) h is the bandwidth allocated to the local BS of s.

For a mobile real-time channel, there are two importance stages of control: *setup* and *handoff.* We first show the setup of a real-time channel with MSs as both its source and destination. (See Figure 2.) When a mobile station wants to set up a channel, it sends to its local BS a *setup request* containing the destination MS's address and the channel's parameters T, C, d. The local BS broadcasts a *paging request*, trying to determine if the destination MS is in existence or if its power is on. The *paging request* is received by each BS, which broadcasts via the wireless medium to ask if the destination MS is in its cell. If the MS's power is on, it replies with a *paging response.* The response is broadcasted by the local BS to all other base stations via the FDDI backbone. Upon receiving the *paging response*, the local BS of the source MS sends a *resource request* to the CS with the channel's parameters. The CS decides whether to accept or reject the request depending on availability of resources. After the decision, the CS sends back (by broadcast) a *resource response*, with a channel ID if the request is accepted. If the *resource response* is positive, the BS of the source MS sends the *setup request* together with the channel ID to the destination MS, of course by way of a local BS. The destination MS replies with a *setup response*, which is delivered all the way to the source MS and the channel is ready. We do not exclude the possibility of the source or destination MS, or both, being moving to another cell during the setup process. With all messages between base stations being sent by broadcast, the above setup procedure is invulnerable of handoffs. Of course we need to assume that information regarding the requested channel will be passed from the old BS to the new BS during a handoff.

A handoff in these mobile networks is distributedly controlled, involving only the old local BS and the new local BS. It is also flexible, permitting the involved MS to exchange data with two BSs for a short period. The handoff process is depicted in Figure 3 and described as follows. When an MS detects that the neighboring BS's signal is stronger, it sends a *handoff request* to this BS. If this new BS can allocate a wireless channel within a certain interval, it sends a *channel assignment request* to the MS. Upon receiving the request, the MS sends back a *channel assignment response.* Afterward, the MS will exchange data with two BSs. Once the *channel assignment response* is received by the new BS, it sends a *release request* to the old BS; upon receiving a *release response* from the old BS, it sends a *handoff response* to the MS. Now the handoff is over. The MS has changed its local BS and will communicate with other stations through the new local BS.

3.3 Problems in Supporting Mobile Real-Time Channels

In the previous section, we showed the interactions between the MS and BSs during a handoff. However, an important question is left unanswered: how the data of the real-time channel will be handled in the old and new BSs so that the channel's QOS requirements can be satisfied. In this section, we first present a straightforward approach to the data handling during a source handoff of a real-time channel (T, C, s, d, h), then we identify some problems with this approach.

First consider the relationship of the handoff time and the token arrivals to the old and new BSs during a source handoff as depicted in Figure 4. Suppose a *handoff request* is received by the new BS and a wireless channel is properly assigned so that a *release request* is sent by the new BS at the token arrival $\bar{t}_{u-M+1}$. At the token arrival t_{u-M+1}, the old BS will send a *release response*

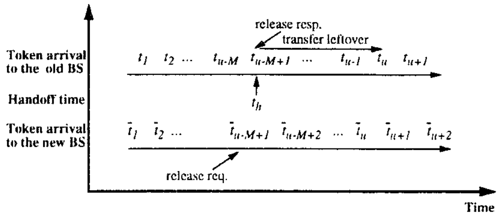

Figure 4: Relationship between time instances

to the new BS. The old BS will not transmit any data generated after t_{u-M+1}, which is regarded as the handoff time, t_h. However, some previously generated real-time packets, called the *leftover*, still have to be transmitted from the old BS. Let the number of token arrivals needed to transmit the leftover be M ($M \geq 1$) so that t_u is the last token visit for the old BS to transmit the data of the channel in question.

A straightforward approach to the handoff handling is that when a handoff happens, the new BS

1) Sends a request RAF of bandwidth h to the bandwidth management process at $\bar{t}_{u-M+1}$.

2) Starts data transmission from $\bar{t}_{u-M+2}$.

And the old BS

1) Stops receiving packets generated after t_{u-M+1}.

2) Continues to transmit the leftover until t_u.

3) Sends a release RAF of bandwidth h to the bandwidth management process at t_{u+1}.

However, simple as this approach is, it has some problems. In the rest of this section, we present these problems.

3.3.1 Overhead Bandwidth Problem

In the above approach, we can see that there is an overlapped time period when both the old and new BSs are allocated bandwidth h. For this time period, the real-time channel occupies bandwidth $2h$. The extra bandwidth of h is called *overhead bandwidth*, which is formally defined as follows:

Definition 3 For a real-time channel (T, C, s, d, h), if two base stations are respectively allocated an amount μ and μ' of bandwidth in a same round of token rotation, then the *overhead bandwidth for that round of token rotation* is $\mu + \mu' - h$. The *overhead bandwidth for the handoff* is defined as the maximum overhead bandwidth among all individual token rotations during the handoff period.

For the straightforward approach, the overlapped period is quite long, *i.e.* M token rotations. The amount is very large as well, *i.e.* h. The problems are how to make the period shorter and the amount smaller.

3.3.2 Bandwidth Management Problems

If the FDDI bandwidth management scheme described in Section 2 is employed, the straightforward approach has some problems:

1) *Heavy burden.* For each handoff, there is one bandwidth allocation and one bandwidth release; four control frames are exchanged. If the handoff rate is high, the burden on the bandwidth management process is very heavy. The problem is to find a bandwidth management scheme so that h can be reallocated from the old BS to the new BS without the BMP's interference.

2) *Slow response.* According to the FDDI's bandwidth management scheme, in the best case, the requested bandwidth can only be granted one token rotation after a request is sent. Consider the time sequence in Figure 4. The value of M is unknown to the new BS at $\bar{t}_{u-M+1}$ because a *release response* can only be sent from the old BS at t_{u-M+1}. If the new BS sends a bandwidth allocation request to the BMP at $\bar{t}_{u-M+1}$, the overlapped time period will be long for a large M. Otherwise, the new BS cannot transmit data at $\bar{t}_{u-M+2}$, which is $\bar{t}_{u+1}$ when $M = 1$; in this case, the channel's delay constraint may be violated. The problem is to find a bandwidth management scheme so that the requested bandwidth can be granted within one token rotation.

3) *Overhead bandwidth under-utilization.* One possible solution to the heavy burden and slow response problems is to pre-allocate a bandwidth h to each BS. This approach is obviously inefficient. Suppose there are 10 BSs in a mobile network and suppose statistically at most three handoffs may occur simultaneously in the whole network. Then, instead of pre-allocating $10h$ bandwidth, $3h$ will be sufficient. The problem is to find a bandwidth management scheme that allows global sharing of the pre-allocated bandwidth.

3.3.3 Ordering Problem

According to Figure 4, the real-time data transmitted from the new BS will be generated after $t_h = t_{u-M+1}$. If the new BS starts transmission at the next token arrival, $\bar{t}_{u-M+2}$, when $M > 1$, the old BS has not stopped transmitting data at t_{u-M+2}. Since the data transmitted by the old BS is generated earlier than those transmitted by the new BS, the data in the real-time channel will be disordered. If the new BS only starts transmission after the old BS has stopped, we face another problem as described in the following section.

3.3.4 Delay Problem

If the new BS holds the real-time data until the old BS has finished the leftover, then some data generated at the new BS may violate the delay constraint, as explained below. Suppose the number of token visits needed to transmit the leftover, M, is greater than 1. The total amount of leftover L satisfies $(M-1)h < L \leq Mh$. Suppose $L < Mh$. Suppose some data are sent by the source MS to the new BS before t_u. Consider the first part of these data which are no more than $Mh - L$. These data immediately follow those leftover at the old BS. Should no handoff have taken place, they would have been sent to the old BS and be transmitted from the old BS at t_u. But because of the handoff, these data are sent to the new BS instead, and will be held from transmission until $\bar{t}_{u+1}$. If the deadline for transmitting the mentioned data is t_u, the delay constraint is violated.

```
Overhead bandwidth management for each BS
IN CASE OF
• Bandwidth pool (BPF) control frame is received
    guard := the value in the BPF frame;
  IF overhead bandwidth $h_o$ requested by MAC
      IF $guard \geq h_o$
          $guard := guard - h_o$;
      ELSE reject the request
  ELSE IF overhead bandwidth $h_o$ released by MAC
      $guard := guard + h_o$;
  Send a BPF control frame with the new value of
    guard to the downstream BS;
```

Figure 5: Overhead bandwidth management

3.3.5 Destination Problem

When a handoff occurs to the destination MS of a real-time channel, there is a *rerouting* problem. Before the handoff, the destination MS receives data from the old BS; but afterward, it receives data from a new BS. In an FDDI-based mobile network, the source station of the real-time channel should be able to notice this change and route the data to the new BS of the destination MS. The problem is how to make the rerouting.

4 Dynamic Bandwidth Management Scheme

In this section, we propose a *dynamic* bandwidth management scheme to solve the problems of Section 3.3.2. This scheme will be implemented in the SMT modules of BSs and the CS. It has following strengths: 1) bandwidth h can be reallocated directly from the old BS to the new BS without burdening the BMP; 2) the demands of overhead bandwidth can be satisfied immediately and some pre-allocated overhead bandwidth can be shared network-wide; 3) it is fully compatible with the FDDI bandwidth management scheme. Only BSs and the CS participate in this new bandwidth management scheme; all other FDDI stations and the BMP still use the original scheme.

The basic idea of this scheme is to make the CS a *proxy* of bandwidth management. Instead of requesting bandwidth by each BS individually, the CS requests bandwidth for all BSs. The bandwidth managed by the CS consists of two parts: the *normal* bandwidth used by the BSs off handoff periods and the *guard* bandwidth used as overhead bandwidth during handoffs. We show how to manage these two types of bandwidth as follows.

In order to share a pre-allocated bandwidth, a global variable *guard* is maintained distributedly by the BSs; it indicates the amount of *guard* bandwidth still available for use as overhead bandwidth. The most updated value of this variable is known by SMT modules of the BSs before each token arrival. When a BS needs some overhead bandwidth at a token arrival, it can test whether the needed amount is available. To implement this scheme, at each token arrival, a BS transmits a control frame, *bandwidth pool* (BPF), to the next BS downstream. This frame is very small, containing only the value of *guard* variable. When this frame is received by a BS, this value is examined and modified if some overhead bandwidth is needed. Then the new value is sent to the next BS in the downstream via another BPF frame. The detailed operations of this scheme are depicted in Figure 5.

```
Normal bandwidth reallocation in a BS
IN CASE OF
• Wanting to reallocate bandwidth to another BS:
    Send a BRF control frame to that BS;
    Decrease the bandwidth allocated to this station;
• Receiving a BRF control frame:
    Increase the bandwidth allocated to this station;
    Notify the MAC module of the reallocation;
Normal bandwidth reallocation in the CS
IN CASE OF
• Resource request received during a channel setup:
    Calculate the bandwidth needed for this channel;
    Send a RAF request to BMP for the bandwidth;
• RAF response received from the BMP:
    IF the requested amount is granted
        Send a BRF control frame to the requesting BS;
```

Figure 6: Bandwidth reallocation management

When a mobile real-time channel is set up, an amount of normal bandwidth is allocated to the CS by the BMP. The bandwidth is then reallocated to the requesting BS by the CS. During a handoff, this normal bandwidth is reallocated directly from the old BS to the new BS. This reallocation scheme alleviates most of the BMP's burden in managing BSs' bandwidth. To implement this scheme, a control frame, *bandwidth reallocate* (BRF), is transmitted from the station releasing the bandwidth to the station accepting the bandwidth. It has one data field, specifying the amount of the reallocated bandwidth. The bandwidth is reallocated once the accepting station receives this frame. The details of the operations are depicted in Figure 6.

5 Handoff Control for Mobile Source

In this section, we first present some solutions to the source handoff problems mentioned in Section 3.3. Then we propose a handoff protocol for implementation within the MAC layer of a BS. Finally, we revisit the QOS problems after some assumptions are dropped.

5.1 QOS Guarantee in Source Handoff

In Section 3.3, there are two problems concerning the QOS during a source handoff: ordering problem and delay problem. For the ordering problem, an easy solution is to make the destination capable of packets reordering. Real-time packets transmitted in FDDI frames should bear some sequence numbers and some buffers should be provided in the destination. If packets are received out of sequence, they are put into buffers until the preceding packets arrive. We first assume the destination has this reordering capability. Then in Section 5.3, we show the ordering problem can still be solved even if this assumption is dropped.

It is easy to see that the straightforward approach can guarantee the delay constraint. The packets transmitted by the new BS will be generated after $t_h = t_{u-M+1}$ and the new BS will be allocated bandwidth h at $\bar{t}_{u-M+2}$, the first token visit after the packets are generated. Since the original bandwidth h is so allocated that the real-time channel is feasible, we know the data transmitted from the new BS will not be delayed. The data transmitted from the old BS will not be delayed as well because the handoff does not affect their transmissions. However, the straightforward approach is not efficient because it may allocate an overhead bandwidth long before it is needed. The problem we will solve in the rest of this section is to find out the last token arrival for the new BS to start data transmission without delay. First we define the *latest cleanup time* in a real-time channel.

Definition 4 For a mobile real-time channel (T, C, s, d, h), let $Q = \langle t_g, t_0, t_1, \ldots \rangle$, where t_g is the starting time of packet generation and $t_0, t_1, \ldots$ are the token visits to the local BS of s after t_g. A time instant $t_c \in Q$ is said to be a *clean-up time* for the channel if after the token visit of t_c, the local BS of s has delivered all packets generated before t_c. The *latest clean-up time* for time t, denoted by $LCT(t)$, is the latest among all the clean-up times that are not later than t. For convenience, regard t_g as a clean-up time so that $LCT(t)$ is meaningful for all $t \geq t_g$.

According to this definition, we can see that all token visits between $LCT(t)$ and t (not include them) will use up the allocated bandwidth h. Using this definition, we derive the following theorem to specify the last token arrival for the new BS to start transmission.

Theorem 2 *If the handoff time $t_h = t_{u-M+1}$ and the new BS is allocated h for its first data transmission, the latest token arrival for the new BS to start transmission without delay is $\bar{t}_{u+1}$ if $M = 1$ or $\bar{t}_u$ if $M > 1$.*

Proof. Let t_g be the packet generation starting time and t'_i $(i \geq 0)$ be the token arrivals to transmit these packets. Bandwidth h allocated by any scheme, such as in [1, 14], ensures that all packets will be transmitted without delay if

$$t'_i \leq t_g + (i+2)TTRT - h \qquad (1)$$

If $M = 1$, $t_h = t_u$. According to Theorem 1, we have $\bar{t}_{u+i+1} < t_{u+i+1} \leq t_u + (i+2)TTRT - h$ for any $i \geq 0$. By Eq.(1), we know data generated after t_h will not be delayed if transmitted at $\bar{t}_{u+1}$. Otherwise, if $M > 1$, let $LCT(t_h)$ be the latest cleanup time for t_h and let t_g be the generation time of the first packet transmitted by the new BS. Suppose there are M' token visits to the old BS after $LCT(t_h)$ (not include it) and before t_u (include it). It is easy to see $M' \geq 2$. According to Definition 4, all token visits between $LCT(t_h)$ and t_u (not include them) use up bandwidth h and all data transmitted by these token visits are generated after $LCT(t_h)$. According to [14], to satisfy the throughput requirement, h should be no less than $(TTRT/T)C$. Since the packets transmitted from the new BS are generated after those transmitted from the old BS, we have

$$\begin{aligned} t_g &\geq LCT(t_h) + (M'-1)h(T/C) \\ &\geq LCT(t_h) + (M'-1)TTRT \qquad (2) \end{aligned}$$

According to Theorem 1, we have

$$\bar{t}_{u+i} < t_{u+i} \leq LCT(t_h) + (M'+i+1)TTRT - h \qquad (3)$$

for any $i \geq 0$. From Eq.(2) and Eq.(3), it is easy to derive that $\bar{t}_{u+i} \leq t_g + (i+2)TTRT - h$. By Eq.(1), we know if the bandwidth h is allocated at $\bar{t}_u$, all packets generated at or after t_g will be transmitted without delay. ■

```
LOOP BEGIN
IN CASE OF
• Handoff request received from an MS
    Call the move-in module;
    Thereafter transmit real-time packets using
        bandwidth h at each token arrival;
• Release request received from another BS
    Call the move-out module;
LOOP END
```

Figure 7: The main program of the handoff protocol

5.2 Handoff Protocol for Mobile Source

Based on Theorem 2 of the previous section and the bandwidth management scheme of Section 4, we propose the following new approach to data handling in a source handoff. When a handoff occurs, the new BS do the following:

1) Receive a *release response* from the old BS and decide the value of M.

2) Request an overhead bandwidth $h_o = h$ out of the guard bandwidth and start transmission at $\bar{t}_{u+1}$ if $M = 1$ or at $\bar{t}_u$ if $M > 1$.

3) Release overhead bandwidth h_o and acquire the reallocated bandwidth h from the old BS at $\bar{t}_{u+2}$.

4) Transmit data using bandwidth h at and after $\bar{t}_{u+2}$.

The old BS do the following:

1) Stop receiving packets generated after t_{u-M+1}.

2) Continue to transmit the leftover until t_u.

3) Release bandwidth h for reallocation at t_{u+1}.

The main program of the handoff protocol to implement this approach is depicted in Figure 7. If no handoff happens, when the token arrives, the handoff protocol transmits data using bandwidth h as usual; when a source handoff takes place, depending on whether the source enters the cell of the BS (a *handoff request* received from the MS) or leaves the cell of the BS (a *release request* received from another BS), the main program calls either the *move-in* or the *move-out* module. The old BS's *move-out* module cooperates with the new BS's *move-in* module to ensure a smooth handoff.

The interactions between the *move-out* and *move-in* modules are explained as follows. Immediately after it is evoked, the *move-in* module allocates a wireless channel and informs the MS of the new channel. Afterward, at the next token arrival, it sends out a *release request* to

```
LOOP BEGIN
• When token arrives
    IF this is the first token arrival after the handoff
        Stop receiving data from the source MS;
        $M := \max(\lceil(\text{the amount of leftover})/h\rceil, 1)$;
        $OH := Mh$ – the amount of leftover;
        Transmit a release response with the values of $M$,
            $OH$ and sequence number of the last received
            packet to the new BS;
    Transmit the leftover to the destination of the
        real-time channel using bandwidth $h$;
    Release the token;
    IF the leftover have been depleted
        Notify the SMT module to reallocate normal
            bandwidth to the new BS;
        Return to the main program;
LOOP END
```

Figure 8: The *move-out* module of the handoff protocol

the *move-out* module of the old BS. At the first token visit after the *release request* is received, the *move-out* module in the old BS stops receiving data from the MS and sends out a *release response* to the *move-in* module. The parameters in the *release response* specify when the new BS should start transmission and from which packet the new BS should transmit. Additional information such as channel ID and destination address is sent to the new BS from the MS via a *handoff request.* The *move-out* module continues to transmit the leftover at bandwidth h until they are depleted. Then before the next token arrival, it notifies the local SMT module to release normal bandwidth h. Based on the parameters of the *release response*, the *move-in* module in the new BS decides how many token visits should be passed before it starts transmission. Before the first transmission that will incur overhead bandwidth, the *move-in* module requests the local SMT module to allocate some overhead bandwidth out of the guard bandwidth. After bandwidth h is reallocated from the old BS, the *move-in* module asks the local SMT module to release the overhead bandwidth. Now the handoff is over and the control is returned to the main program.

5.3 Further Discussions

In this section, we will discuss what changes should be made to the handoff protocol when two assumptions, destination reordering and soft handoff, are dropped.

If the destination cannot reorder packets, the new BS cannot send packets directly to the destination until the old BS has depleted the leftover. However, the delay constraint may be violated as shown in Section 3.3.4. To overcome this dilemma, the handoff protocol should do *packet relay.* Before the leftover are depleted in the old BS, the new BS starts to relay some packets to the old BS, which will be sent later to the destination by the old BS. If $M = 1$, no relay is needed because the new BS starts after t_u. Otherwise, if $M > 1$, it is enough to relay data of OH units, the amount of data to fill up the last transmission of the old BS. By relaying these data, all the data transmitted by the new BS to the destination at $\bar{t}_{u+1}$ would be transmitted from the old BS at t_{u+1} should no handoff have taken place. Since $\bar{t}_{u+1} < t_{u+1}$, no packet transmitted by the new BS to the destination is late.

If the assumption of soft handoff is dropped, the MS cannot maintain data exchange with both the old and the new BS. Therefore, we cannot control the data flow and make $t_h = t_{u-M+1}$. However, t_h should satisfy $\bar{t}_{u-M} < t_h \le \bar{t}_{u-M+1}$. Similar to Theorem 2, it can be proved that the new BS should start transmission at $\bar{t}_u$ no matter $M = 1$ or $M > 1$.

```
Allocate a wireless channel;
Send a channel assignment request to the MS:
Wait until channel assignment response is received;
LOOP BEGIN
IN CASE OF
• Token arrival
    IN CASE OF
        • this is the first arrival after
                channel assignment response
            Send a release request to the old BS;
        • $M > 2$ and this is one of the $M - 2$ arrivals
                after release response
            Do nothing;
        • $M > 1$ and this is the $(M - 1)$th or $M$th arrival
                after release response
            Transmit data using $h_o$ if available;
        • $M = 1$ and this is the first arrival after
                release response
            Transmit data using $h_o$ if available;
    Release the token;
    IF $M > 2$ and this is the $(M - 2)$th arrival after
        release response
        Ask SMT module for overhead bandwidth $h_o$;
• Release response received from the old BS
    $M, OH$ := the values in release response;
    Discard the packets up to the sequence number in
        release response;
    IF $M = 1$ or $M = 2$
        /* transmission starts at next token visit */
        Ask SMT module for overhead bandwidth $h_o$;
• Notified by SMT module of bandwidth reallocation
    Ask SMT module to release overhead bandwidth $h_o$;
    Return to the main program;
LOOP END
```

Figure 9: The *move-in* module of the handoff protocol

6 Handoff Control for Mobile Destination

In this section, we give solutions to the problem caused by mobile destinations. Two approaches, multicasting and rerouting, are addressed and compared.

If the destination of a real-time channel is an MS, we have to ensure that data still be received by the destination even if it has changed its local BS. Specifically, there are three requirements: 1) No packet is duplicatedly delivered from both the old and the new BS of the destination MS; 2) No packet is lost; 3) All packets are delivered to the destination MS once the BSs receive them, i.e. no delay in delivery. To meet these requirements, we may adopt one of two approaches: *multicasting* or *rerouting.* For the first approach, the source station multicasts the packets to all BSs. Upon receiving the packets, the BSs decide which packet should be delivered to their MSs. For the second approach, packets are only sent to the local BS of the destination MS. The BSs deliver all received packets to their MSs. The first approach does not incur extra traffic but it imposes more load on the BSs because they have to receive and process the packets not intended for them. Its main advantage is no need for rerouting. The implementations of the two approaches are presented as follows.

In the multicast approach, after a *release request* is received by the old BS of the destination MS, in the next token visit, it sends out a *release response* to the new BS and then stops delivering any packet received afterward. Upon receiving a *release response*, the new BS starts to deliver the packets received after this response. There is no packet loss or duplication because the packets transmitted from the source before the *release response* are delivered from the old BS and those transmitted afterward are delivered from the new BS. No packet is delayed in delivery to the destination MS as well because the new BS does not hold any packet received after the *release response.* In the rerouting approach, after a *release request* is received, in the next token visit, besides sending a *release response* to the new BS, the old BS also broadcasts a *destination reroute* control frame (DRF) with this channel's ID to all stations. Upon receiving this control frame, the source station starts data transmission to the new BS. No packet is duplicated, lost or delayed because any packet is transmitted to only one BS and delivered promptly. Since the DRF frames consume some bandwidth, when handoff rate is high, this approach is discouraged.

7 Conclusion

We have addressed two issues: (1) how to build a mobile network using the FDDI as a backbone and (2) how to provide real-time communication service on such a network. To the best of our knowledge, this is the first paper on these topics. In particular, we described an architecture of FDDI-based mobile networks, identified some problems in supporting real-time communications, and proposed possible solutions. Our solutions are compatible with the current FDDI standards in the sense that only base stations and the control station will need our schemes. Those stations having nothing to do with mobility will need no modification. In particular, the bandwidth management process remains unchanged. A complete version of this paper is available as a technical report [12], which includes an algorithm to minimize the overhead bandwidth.

References

[1] G. Agrawal, B. Chen, and W. Zhao, "Local synchronous capacity allocation schemes for guaranteeing message deadlines with the timed token protocol," in *Proc. INFOCOM*, pp. 186–193, March 1993.

[2] ANSI X3T9, *FDDI Media Access Control-2 (MAC-2) - draft proposed.* American National Standard, ANSI X3T9/92-120, 1992.

[3] ANSI X3T9, *FDDI Station Management (SMT) - draft proposed.* American National Standard, ANSI X3T9/92-067, 1992.

[4] T. Balch and R. C. Arkin, "Communications in reactive multiagent robotic systems," *Autonomous Robots*, vol. 1, pp. 27–52, 1994.

[5] B. Chen and W. Zhao, "Properties of the timed token protocol," Technical Report TAMU 92-038, Department of Computer Science, Texas A&M University, 1992.

[6] D. Ferrari and D. C. Verma, "A scheme for real-time channel establishment in wide-area networks," *IEEE Journal on Selected Areas in Communications*, vol. SAC-8, pp. 368–379, April 1990.

[7] D. J. Goodman, "Cellular packet communications," *IEEE Transactions on Communications*, vol. 38, no. 8, pp. 1272–1280, August 1990.

[8] B. Jabbari, G. Colombo, A. Nakajima, and J. Kulkarni, "Network issues for wireless communications," *IEEE Communications Magazine*, pp. 88–98, January 1995.

[9] R. Jain, *FDDI Handbook: High Speed Networking Using Fiber and Other Media.* Addison Wesely, 1994.

[10] M. J. Johnson, "Proof that timing requirements of the FDDI token ring protocol are satisfied," *IEEE Transactions on Communications*, vol. Com-35, no. 6, pp. 620–625, June 1987.

[11] D. D. Kandlur, K. G. Shin, and D. Ferrari, "Real-time communications in multi-hop networks," in *Proc. IEEE International Conference on Distributed Computing Systems*, pp. 300–307, 1991.

[12] T.-H. Lai, Y. Yang, and M. T. Liu, "Real-time communication in FDDI-based mobile networks," *Technical Report (postscript file available from the authors)*, 1995.

[13] S. Tekinay and B. Jabbari, "A measurement-based priorization scheme for handovers in mobile cellular networks," *IEEE Journal on Selected Areas in Communications*, vol. 10, no. 8, pp. 1343–1350, Oct. 1992.

[14] Q. Zheng and K. G. Shin, "Synchronous bandwidth allocation in FDDI networks," *to appear in IEEE Transactions on Parallel and Distributed Systems*, 1995.