Abstract: Medium access control (MAC) protocol is one of the key components for providing quality of service (QoS) in wireless ATM (W-ATM) networks. In this paper, we propose a hierarchical scheduling scheme coupled with fair queuing algorithms with adaptive weights. This scheme is intended to be applicable to a TDMA/TDD based MAC protocol. Specifically, the performance of the fair-queueing algorithm using fixed weights and adaptive weights are investigated. Simulation results show that the proposed hierarchical fair-queueing scheduling with adaptive weights (HAW) can yield a lower cell transfer delay and higher channel utilization while maintaining fairness among multiple users.

I. Introduction

Wireless ATM (W-ATM) has been proposed as a potential solution for supporting future wireless multimedia communications [1]. One of the most challenging and important issues in W-ATM is the medium access control (MAC) protocol, which has a significant impact on the performance of W-ATM networks [2, 3]. The W-ATM MAC protocol needs to support different ATM services (e.g., CBR, VBR, ABR, and UBR) with differentiated QoS (quality of service) requirements, such as cell loss ratio, maximum cell transfer delay, and cell delay variation, etc., over wireless channels while achieving a high channel utilization. To realize this goal, the MAC protocol must support the isolation of different traffic flows such that multiple users will appreciate fairness in sharing the available resources of W-ATM networks [4].

In this paper, we consider a centralized W-ATM architecture where the base-station (BS) acts as a central controller which is responsible for the traffic management of the remote mobile terminals (MTs). Moreover, a TDMA/TDD (Time Division Multiple Access/Time Division Duplex) access scheme is employed in order to achieve a higher multiplexing gain, especially under asymmetric traffic conditions, i.e. different uplink and downlink bandwidth [5, 6]. Due to the requirement of an accurate control of the remote mobile terminals, the base-station and the mobile terminals must exchange connection status information. As these updates consume valuable bandwidth, i.e. overhead, which could otherwise be used for data transmission, their frequency should be kept as small as possible.

In order to minimize the overhead, Sigle et al proposed a hierarchical scheduling scheme coupled with fair-queueing algorithms [4]. In this paper we further extend the algorithm by the use of adaptive weights. Numerical results indicate that the hierarchical fair-queueing scheduling with adaptive weights can yield a lower cell transfer delay and a higher channel utilization while maintaining fairness among multiple users.

The remainder of this paper is organized as follows. In Section 2, we introduce a general framework of the MAC protocol in W-ATM networks. In particular, the detail of the TDMA/TDD frame structure is described. Section 3 presents the proposed hierarchical fair-queueing algorithm. Simulation results and performance comparison are given in Section 4. Finally, the conclusions are drawn in Section 5.

II. Medium Access Control in Wireless ATM

The main task of an advanced MAC protocol is to control the allocation of radio resources in a fair and efficient manner. To achieve this goal, a centralized access architecture is employed, for which the most important techniques

* This work is supported in part by the DAAD/RGC Joint Research Scheme under the project no. G-HK96/97.EG03
being considered are TDMA schemes with dynamic slot assignment. This mechanism allows a flexible accommodation
of bandwidth by allocating more or less time slots depending on current traffic conditions. TDMA can operate in either
TDD or FDD (Frequency Division Duplex) mode. Since TDD does not need separate uplink and downlink bands but
operates with a single carrier, a higher multiplexing gain can be achieved which leads to a high channel utilization
especially in the case of asymmetric traffic from uplink and downlink. In other words, the TDMA/TDD MAC protocol
is able to efficiently and dynamically allocate the available bandwidth to connections based on their current need and
traffic load through reservation and/or allocation cycles [3, 5, 6].

A. Fair-Queueing Scheduling Algorithm

As discussed above the scheduling algorithm is one of the most important mechanisms in a wireless MAC protocol.
In order to provide guaranteed end-to-end delay and fairness among multiple users with different traffic flows, fair-
queueing scheduling is suggested for packet switched networks [7, 8]. Fair-queueing scheduling is based on the fluid
flow model of the generalized processor sharing (GPS) mechanism. In GPS a weight $\phi_k$ is assigned to each session $k$
and the available resources are shared among all backlogged sessions according to these weights. It is noted that various
fair-queueing strategies, although they were originally developed for fixed packet switched networks, also play an
important role in wireless networks [4, 9] to ensure an efficient and fair sharing of the limited resources.

For GPS, fairness is defined as follows. For any arbitrary time interval $[t_1, t_2]$ with session $i$ and session $j$ being
backlogged continuously,

$$\frac{W_i(t_1,t_2)}{\phi_i} = \frac{W_j(t_1,t_2)}{\phi_j},$$

(1)

1 Normally, one dataslot contains one ATM cell and its cyclic redundancy code (CRC).
where $W_i(t_1, t_2)$ is the amount of session $i$ traffic served in the interval $[t_1, t_2]$. In other words, the server serves all backlogged sessions simultaneously, in proportion to their service weights. The weights are assigned during connection setup depending on the source parameters of the connections and their QoS requirements.

As the fluid flow model of GPS can not directly be applied to packet switched networks, packet-by-packet GPS (PGPS, also called weighted fair queueing) is proposed in [10]. In PGPS, the notion of virtual time is introduced for scheduling purposes. The virtual time is used to emulate the GPS model within the packet environment. Packets are served in increasing order of their service tags which are assigned to each packet upon its arrival at the queue. Specifically, the service tags can be calculated by the following equation:

$$F_k^i = \frac{L_k^i}{\phi_k} + \max(F_k^{i-1}, v(a_k^i))$$  \hspace{1cm} (2)

where

$F_k^i$ = service tag of the $i$-th packet from session $k$,
$L_k^i$ = length of the $i$-th packet from session $k$,
$a_k^i$ = arrival time of the $i$-th packet from session $k$,
$v(t)$ = virtual time function,
$\phi_k$ = weight of connection $k$.

Since the virtual time function of PGPS requires frequent recalculation, a modified algorithm called self-clocked fair queueing (SCFQ) [11], was proposed. In SCFQ, the $v(t)$ in Eqn. (2) is defined as the service tag of the packet in service at time $t$. Throughout, SCFQ will be chosen as the basis for the development of our proposed effective hierarchical fair-queuing scheme.

B. Hierarchical Fair-Queueing Scheduling with Adaptive Weights

As mentioned above, hierarchical fair-queueing scheduling with proper design can potentially minimize the MAC overhead and guarantee reasonable fairness. A model of hierarchical scheduling is depicted in Fig. 2.

![Fig. 2 Hierarchical fair queueing scheduling strategy.](image)

In this approach, a two-level fair queueing scheduling is implemented in the BS and MTs. We assume that the terminal $MT_i$ has $K_i$ sessions with queue lengths $q_{i,1}, q_{i,2}, ..., q_{i,K_i}$ and service weights $r_{i,1}, r_{i,2}, ..., r_{i,K_i}$. Likewise, in BS, the downlink traffic may have $B_j, B_{j2}, ..., B_{jn}$ sessions with service weights $r_{B_j}, r_{B_{j2}}, ..., r_{B_{jn}}$. The BS scheduler serves MTs based on their weights $\phi_i, i=1,2, ..., n$. In the second level, the MT scheduler decides which connection inside the station will be served on the basis of the weights $r_{i,j}$. We consider the traffic in mobile terminals. Let $W_{MT_i}(t_1, t_2)$ denote the amount of traffic served by $MT_i$ in the interval $[t_1, t_2]$. That is
\[ W_{MT_i}(t_1, t_2) = \sum_{j=1, d_{i,j} = 0}^{K} W_{ij}(t_1, t_2), \]  

(3)

where \( W_{ij}(t_1, t_2) \) represents the amount of session \( j \) traffic served in the interval \([t_1, t_2]\) in \( MT_i \).

In [4], the weights \( \phi_i \) are assigned once during connection setup (we refer to this mechanism as hierarchical scheduling with fixed weights, HFW). If not all sessions of one MT are backlogged, these sessions receive more service compared to the case when all connections are served individually. This may lead to a discrimination of other sessions. Therefore, we propose to adapt the weight \( \phi_i \) of a MT depending on the set of backlogged sessions. That is,

\[ \phi_i = \sum_{j=1, d_{i,j} = 0}^{K} r_{i,j}. \]  

(4)

In order to realize this scheme, the MT announces \( \phi_i \) together with the aggregated queue length in its data and request slots. It is worth noting that hierarchical fair-queueing operates in a frame-based manner. In practical cases such as with the TDMA/TDD access scheme, scheduling method may have an impact on fairness guarantee. Recall that the queue state and the current weight of the MTs can only be updated by the scheduler in the BS at the beginning of next frame. On the other hand, in the MTs, the scheduler works based on the instantaneous queue information. As a result, when some queues in MTs change their state from empty to non-empty or vice versa, the real schedule weight of a MT could be smaller or larger than the weight assigned to the MT by the base station at the beginning of the frame. In order to minimize the impact of this effect, the weight of each MT should be updated quickly. Therefore, the frame length must be bounded. As a consequence, the sessions in MTs will be served in terms of their service weights adaptively while maintaining reasonable fairness. In the following section, simulation results are presented which will demonstrate that hierarchical scheduling with adaptive weights has a better property of fairness than that with fixed weights.

### III. Simulation Results

In computer simulations, we evaluated the delay characteristics and the fairness property of the proposed hierarchical fair queueing with adaptive weights (HAW), and compared it to the non-hierarchical (NH) case and the hierarchical scheduling with fixed weights (HFW). In particular, we assume an error free channel with a capacity of \( 2 \times 10^4 \) cells/sec and a variable frame length with maximum number of 20 dataslots including the frame header and minislots. Moreover, the transceiver turn around time is ignored and the guard time and physical layer preamble are included in the wireless header. The parameters for the TDMA/TDD frame structure are shown in Table 1.

<table>
<thead>
<tr>
<th>Definition</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Channel rate</td>
<td>2×10^4 cells/sec</td>
</tr>
<tr>
<td>Frame length</td>
<td>20 dataslots</td>
</tr>
<tr>
<td>Minislot length</td>
<td>1/4 dataslot</td>
</tr>
<tr>
<td>Frame header</td>
<td>2 dataslots</td>
</tr>
<tr>
<td>wireless header for uplink traffic</td>
<td>1/4 dataslot</td>
</tr>
</tbody>
</table>

Table 1. Parameters of TDMA/TDD frame structure

We consider a scenario with four MTs and three bidirectional connections between each MT and the BS, two rtVBR (real-time variable bit rate) and one UBR (unspecified bit rate) connections which are modeled as on-off and Poisson sources, respectively. The traffic parameters and the corresponding weights are shown in Table 2. It is noted that the bursty traffic has a higher weight due to its tighter delay requirement.

<table>
<thead>
<tr>
<th>Type</th>
<th>Average rate (cells/s)</th>
<th>Peak rate (cells/s)</th>
<th>Mean Burst Size (cells)</th>
<th>Delay Limitation (ms)</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>On-Off</td>
<td>500</td>
<td>2000</td>
<td>20</td>
<td>25</td>
<td>10</td>
</tr>
<tr>
<td>Poisson</td>
<td>1000</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>4</td>
</tr>
</tbody>
</table>

Table 2. Traffic Parameters

Fig. 3 shows the complementary delay distribution function of the uplink and downlink. It is observed that the HAW scheduling outperforms the NH for both uplink and downlink traffic. Although the HFW performs best for the uplink traffic, it has a very poor performance which is even worse than that of NH for downlink traffic. This is because in HFW the weight of each mobile is equal to the sum of weights of all connections in the mobile, regardless of whether the queues for the connections are empty or not. Hence, the uplink traffic can get more resources than the downlink traffic.
traffic when some queues of the MT are not backlogged. This unfair bandwidth allocation could lead to an even worse delay performance of downlink traffic when the traffic load of downlink traffic is much higher than the load of uplink traffic as in most client-server applications.

The delay performance for Poisson sources is shown in Fig. 4. It is noted that the HFW, although it has better delay performance for uplink traffic compared to HAW, suffers from a significantly increased delay for downlink traffic. In contrast, our proposed HAW can always balance well the delay performance between uplink and downlink.

The numerical results for evaluating the fairness of HAW and HFW are given below. In simulations, we use the same setting as before (i.e., each MT has 2 rt-VBR and 1 Poisson sources) and compare the Poisson sources only. We consider two sessions of Poisson sources, i.e. sessions $i$, $j$, with weights $r_i$, $r_j$, respectively. Moreover, sessions $i$ and $j$ are from uplink and downlink traffic, respectively. Let $\Delta W = \max \left( \frac{W_i(t_1, t_2)}{r_i} - \frac{W_j(t_1, t_2)}{r_j} \right)$, where $W_i(t_1, t_2)$ and $W_j(t_1, t_2)$ are the amount of traffic served for the backlogged sessions $i$, $j$. For different $\Delta t$ ($\Delta t = t_2 - t_1$) which is the time period that sessions $i$ and $j$ are continuously backlogged, $\Delta W$ is measured and plotted in Fig. 5. Clearly, for any value of $\Delta t$, HAW keeps the maximum value of $\left| \frac{W_i(t_1, t_2)}{r_i} - \frac{W_j(t_1, t_2)}{r_j} \right|$ quite small. But, with HFW,
the fairness becomes increasingly worse as $\Delta t$ gets larger. Hence, we conclude by noting that the fairness can be better sustained by the use of the proposed HAW scheduling algorithm.

![Fig. 5 Measured fairness performance of HAW and HFW.](image)

IV. Conclusions

This paper proposes a W-ATM MAC protocol which is based on a hierarchical fair-queueing scheduling. The application of our weight adaptation mechanism in a centralized TDMA/TDD access architecture is demonstrated and its performance is evaluated. Simulation and numerical results indicate that the proposed protocol can yield lower cell delay and higher channel utilization while maintaining fairness compared to the protocol using fixed weights.

In further study, channel errors will be considered in the protocol by reserving bandwidth for retransmission purposes.

References


