A Modified Dynamic Weighted Round Robin Cell Scheduling Algorithm

Ji-Young Kwak, Ji-Seung Nam, and Doo-Hyun Kim

In this paper, we propose the modified dynamic weighted round robin (MDWRR) cell scheduling algorithm, which guarantees the delay property of real-time traffic and also efficiently transmits non-real-time traffic. The proposed scheduling algorithm is a variation of the dynamic weighted round robin (DWRR) algorithm and guarantees the delay property of real-time traffic by adding a cell transmission procedure based on delay priority. It also uses a threshold to prevent the cell loss of non-real-time traffic that is due to the cell transmission procedure based on delay priority. Though the MDWRR scheduling algorithm may be more complex than the conventional DWRR scheme, considering delay priority minimizes cell delay and decreases the required size of the temporary buffer. The results of our performance study show that the proposed scheduling algorithm has better performance than the conventional DWRR scheme because of the delay guarantee of real-time traffic.

Manuscript received Apr. 12, 2001; revised Aug. 7, 2002.

Ji-Young Kwak (phone: +82 42 860 4807, e-mail: jiyoung@etri.re.kr) and Doo-Hyun Kim (e-mail: doohyun@etri.re.kr) are with Mobile Collaboration Research Team, ETRI, Daejeon, Korea.

Ji-Seung Nam (e-mail: jsnam@chonnam.ac.kr) is with Chonnam National University, Gwangju, Korea.

I. INTRODUCTION

The communication protocols for multimedia traffic have received a great deal of attention in the past few years. Since multimedia traffic must support various types of traffic simultaneously, it is crucial to process data according to its characteristics. Thus, protocol designers have to grasp the characteristics of traffic and select a processing method suitable for the performance requirements. For instance, real-time audio traffic in a voice service requires rapid transmission, but the loss of a small amount of audio information is tolerable. On the other hand, the transfer of a text file should guarantee 100% reliable transfer; real-time delivery is not of primary importance in this case. Real-time video service, such as video on demand (VOD), requires not only rapid transfer but also high reliability. When a piece of video information is lost, its quality of service (QoS) is degraded. Therefore, multimedia communication protocols should be designed to provide the performance requirements of a wide range of multimedia services [1].

Asynchronous transfer mode (ATM) was proposed in the 1980s as a step in the evolution of public networks to support broadband services. Six different ATM service categories have already been defined. These service categories relate traffic characteristics and QoS requirements to network behavior. The constant bit rate (CBR) and real-time variable bit rate (RT-VBR) are intended for real-time applications, while the non-real-time variable bit rate (NRT-VBR), the available bit rate (ABR), the unspecified bit rate (UBR), and the guaranteed frame rate (GFR) are intended for non-real-time applications [2]. In an ATM network, information is transported over virtual connections (VCs) in fixed size cells. The primary advantage of the ATM is its ability to allocate bandwidth flexibly and dynamically. To be useful, ATM networks, which are conceived to carry heterogeneous traffic streams, must be able to guarantee diverse QoS requirements, such as diverse upper bounds on the end-to-end cell loss ratio and cell delay variation on a per-VC basis. However, there are no intrinsic mechanisms in ATM networking to guarantee specified end-to-end QoS requirements on a per-VC basis [3]. Furthermore, due to statistical multiplexing, more than the allocated number of cells are allowed to enter ATM networks. This causes congestion in the network [4]. Therefore, appropriate traffic control of call admission and congestion is required [5], [6]. One mechanism that can be employed to prevent congestion and guarantee QoS requirements is cell scheduling at the intermediate nodes. A scheduler simply decides which cell to send next, that is, it schedules network resources among the applications of various classes. The simplest service scheduling scheme is the first-in-first-out (FIFO) scheme in which cells are served in the order of their arrival times. However, the FIFO scheme does not provide adequate isolation among traffic streams, and hence the QoS experienced by a traffic stream is influenced by the behavior of other traffic streams. This problem is generated in ATM networks by the presence of heterogeneous traffic streams with diverse QoS requirements. For this reason, a service scheduling scheme that provides adequate isolation among competing traffic streams is desirable [7]. From this viewpoint, per-VC queuing is a good alternative. In per-VC queuing, a packet switch schedules cells to be transmitted on a link based on their VC [8]. A number of scheduling schemes for per-VC queuing have been proposed. These schemes include the virtualclock [9], weighted fair queuing (WFQ) [10], and self-clocked fair queuing (SCFQ) [11], [12]. Among the existing scheduling schemes, a simple scheme with fairly good performance is the weighted round robin (WRR) [13]. This scheduling algorithm uses a fixed weight for each channel. This means that the requirements for only CBR traffic sources are considered. After the WRR scheduling algorithm was proposed, the dynamic weighted round robin (DWRR) was proposed to support VBR and CBR traffic sources by assigning additional dynamic weighted value to each source. However, the DWRR scheduling algorithm focuses on the performance improvement of VBR sources. It does not consider the ABR traffic source, which is a major traffic source for data service. In addition, it does not consider UBR traffic or the cells that violate the service requirements for the channel [14]. To support multiple classes of traffic with varying delays and loss requirements, a scheduling scheme that guarantees diverse QoS requirements is needed. In this paper, we propose the modified dynamic weighted round robin (MDWRR) cell scheduling algorithm, which guarantees the delay property of real-time traffic, a factor that was not considered in the DWRR algorithm. It also considers the service of ABR traffic. We evaluate the performance of the proposed scheduling algorithm through computer simulation.

II. CONVENTIONAL SCHEDULING METHODS

1. Weighted Round Robin

The weighted round robin scheduling mechanism multiplexes cells from every virtual channel connection (VCC) with different priority levels. It is an extension of round robin scheduling based on the static weight. It services each VCC link in turn. Each VCC link can transmit one cell in its turn when there are cells to transmit. Each class queue has a counter that specifies the number of cells that can be sent. The counter value is initially equalized to the weight value assigned to that class. Cells from various classes are sent in a cycle from the head of these class queues while counter values are greater than zero. After sending a cell, the counter value of the class is reduced by one. When the counter value or the queue length has reached zero in all classes, all counters are reset to their weight values. Figure 1 shows an example of WRR cell scheduling, where weights of 2, 1, and 3 have respectively been allocated to queues 1, 2, and 3 [15].

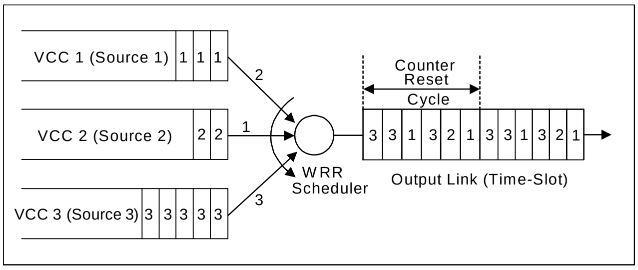

Fig. 1. Weighted Round Robin.

For a short time interval, a class may have more cell arrivals than its weight value can support. This burstiness of input traffic becomes a major factor creating delay. In this case, the counter value will reach zero before all the cells in the queue have been sent, and cell scheduling for this class will be suspended until the next counter reset [16].

In WRR, differential cell transmission capability for input VCCs is supported on the basis of the static priority level. However, this scheme is static and only the CBR service is considered because of the fixed weighted priority level for each VCC link [17].

2. Dynamic Weighted Round Robin

DWRR was proposed to support VBR and CBR traffic sources by assigning an additional dynamic weighted value to each source. In this algorithm, the peak cell rate (PCR) and average cell rate (ACR) characteristics are considered. This scheduling algorithm calculates the cycle size on the basis of VBR and CBR traffic sources. In this algorithm, ABR and UBR traffic are not considered. In addition, valid ABR cells may not be serviced because of invalid VBR and CBR cells that violate the established QoS parameters [16].

The ATM output port can be thought of as a train of time-slots. The time-slots in the ATM output are structured into a sequence of fixed-length time intervals called cycles. Each cycle contains a fixed number of time-slots. In each cycle, the time-slots are further divided into several rounds, as shown in Fig. 2. Each round may contain a variable number of time-slots. A circular scan on the traffic sources is done within every round. When a source is visited, a cell of this source is allowed to be transmitted by one of the time-slots in this round.

As shown in Table 1, the peak cell rate of a traffic source, B_P, and the average cell rate of a traffic source, B_M, have a special relation to the characteristics of each class of traffic; in AR traffic, B_M represents the minimum cell rate of a traffic source.

For CBR sources, the value of B_P is set to the same value as B_M. For VBR and ABR sources, the value of B_P is set to a value larger than B_M. For best-effort sources, the values of B_P and B_M are set to 0.

As shown in Fig. 3, the DWRR algorithm handles the VBR and CBR and the best-effort traffic sources dynamically. To assure QoS, VBR and CBR sources have to capture a guaranteed portion of the time-slots in every fixed time interval. Then, the remaining time-slots are for the best-effort traffic. The cells from each source are assumed to arrive at a random time slot within a cycle. Cells that do not arrive at the time of their assumed arrival are stored in the temporary buffer. However, they should be transmitted as soon as possible in the next round's scan if the sources are VBR or CBR services.

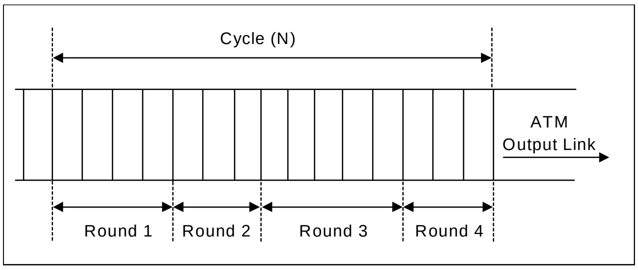

Fig. 2. The cycle and rounds in the ATM outgoing time-slots.

Table 1. The relation between B_P and B_M according to the characteristics of traffic.

The class of traffic sources	The relation between B_P and B_M
CBR traffic	$B_P = B_M$
VBR traffic	$B_P \geq B_M$
ABR traffic	$B_P \geq B_M$
UBR traffic	$B_P = B_M = 0$

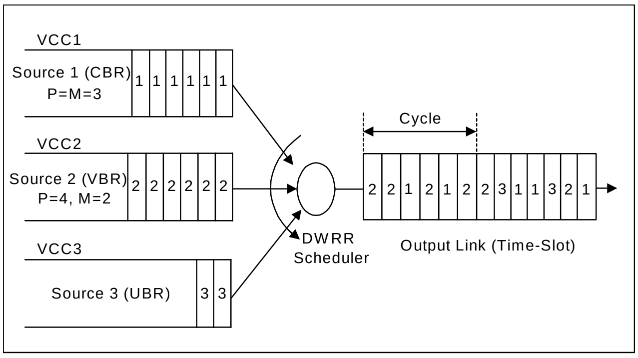

Fig. 3. DWRR cell scheduling.

The concept of the DWRR algorithm is stated as follows. Each VBR or CBR source is associated with a pair of counters, namely P and M. At the beginning of each cycle, the counter P is set to the peak cell rate $B_P(s)$ of the corresponding source s and the counter M is set to the average cell rate $B_M(s)$. The counters P and M are decreased by 1 when the source is visited and a cell is transmitted successfully. In each round, each source falls into one of the following states:

- State 1: $P > 0$, $M > 0$ and there are cells stored in the temporary buffer.
  In this case, the source is visited in the next round.
- State 2: $P > 0$, $M \leq 0$ and there are cells stored in the temporary buffer.
  In this case, the source is not visited as long as there are other sources staying in State 1. The arriving cell is pushed into the temporary buffer.
- State 3: $P \leq 0$, $M \leq 0$ or no cell arrival at the moment.
  In this case, the source will not be visited in the subsequent rounds.

The state of each source is examined at each time-slot of the cycle to ensure that each source can perform the required state transition upon the new arrival of a cell. The algorithm selects the sources which are currently in State 1 at the beginning of the time-slot and forms a round-robin scan in the cycle. It then waits until the next time-slot and performs the same action. If there is no source in State 1, then the algorithm performs the same steps and forms a round. If there is no source in States 1 and 2, then one of the sources belonging to the best-effort type is visited and a round is formed. To ensure a fair bandwidth among the best-effort sources, the visits in a cycle are scheduled in a round-robin manner [14].

III. THE MDWRR CELL SCHEDULING ALGORITHM

In future broadband high-speed networks, ATM packet switches should be able to support diverse applications, such as voice, data, image, video, and even unknown future services, which have different traffic characteristics and required QoS. If each class of traffic is treated equally in a network, the network must maintain the most stringent QoS in order to satisfy the required QoS of all types of traffic. Consequently, the utilization of packet switches must be kept small or large buffers used to prevent overflows. These restrictions limit the flexibility and cost-effectiveness of integrated ATM networks. Thus, some type of priority control scheme which treats each class of traffic according to its QoS requirements is needed to increase the utilization of packet switches and to satisfy the QoS of each traffic type [18].

The representative QoS requirement of an existing ATM service mainly depends on two parameters, the end-to-end cell transfer delay and the cell loss probability. Thus, it is necessary to distinguish packets and assign priorities to them based on their delay and loss constraints in the networks. One of the best methods to provide these traffic control capabilities is the priority queuing strategy. Priority queuing efficiently accommodates diverse QoS requirements according to their delay and loss constraints in a switching system. The first constraint is traditionally referred to as the time priority (or service priority) discipline and can typically be employed among sessions with different service requirements. The second constraint, called the space priority (or loss priority) discipline, discriminates between service classes without changing their relative time ordering and can also be used within the same session or application. The priority function of cell loss control may be provided by the user and can be included in the usage parameter control, such as the virtual leaky bucket algorithm. On the other hand, the priority function of cell delay control can be included in the ATM switch to satisfy the cell delay requirements efficiently.

The time priority scheme is suitable for satisfying the cell delay requirements of different traffic types in a switching system, and the delay property is critical to the real-time traffic of multimedia, because packet loss may occur because of delay bound violation. Our proposed MDWRR scheduling algorithm guarantees the delay property of real-time traffic, which was not considered in the DWRR algorithm. The MDWRR uses the delay priority in the cell transmission procedure, and this algorithm sets the threshold to non-real-time traffic for the purpose of preventing the cell loss of non-real-time traffic due to the cell transmission procedure based on delay priority.

Among the various categories of ATM service, ABR, UBR and GFR support data traffic such as transmission control protocol (TCP) packets. The ABR service relies on a rate-based scheme that requires complicated rules for source behavior and uses a special resource management (RM) cell to indicate the current state of congestion. The UBR service is the lowest priority service and does not guarantee cell loss or cell delay performance. The GFR service, which is proposed to fill the gap between the UBR and ABR, provides a minimum rate guarantee to VCs at the frame level. It also allows for the fair usage of any additional bandwidth left over from higher priority connections. The GFR is a frame-based service with a minimum guaranteed rate while all other ATM service categories are cell-based services. Hence, the GFR explicitly requires the end systems to transmit frames and also requires the ATM switches to be aware of the frame boundaries. This means that congested ATM switches should normally discard entire frames instead of individual cells. Thus, in the MDWRR algorithm, the GFR service is not supported, because a complex method for the frame-based service is required.

Figures 4 and 5 show the service mechanism of the DWRR algorithm and the MDWRR algorithm. As the figures show, the service order in the MDWRR algorithm is similar to that in the DWRR algorithm, but with the difference that the MDWRR algorithm serves ABR traffic, which was not considered in the DWRR algorithm to the same degree as NRT-VBR traffic. The MDWRR scheduling algorithm serves with the highest priority CBR traffic and RT-VBR, NRT-VBR, and ABR traffic below the minimum cell rate (MCR). RT-VBR, NRT-VBR, and ABR traffic above the MCR and below the PCR have the second priority. UBR traffic has the lowest priority. The algorithm provides high priority for NRT-VBR

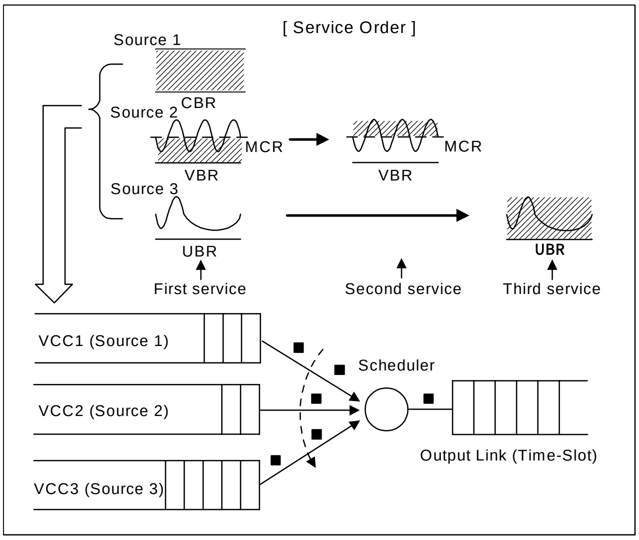

Fig. 4. The service mechanism of the DWRR cell scheduling algorithm.

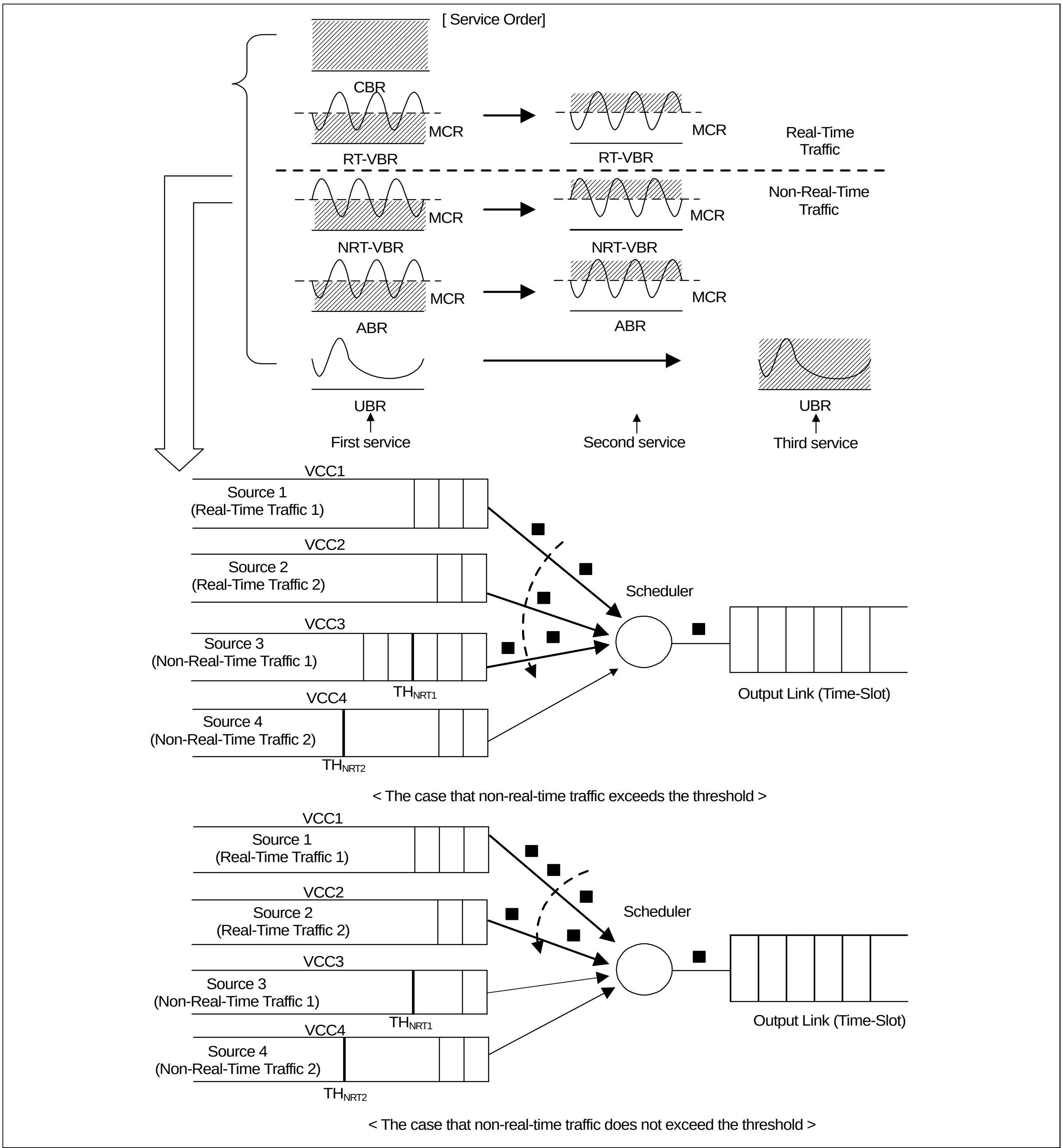

Fig. 5. The service mechanism of the MDWRR cell scheduling algorithm.

traffic by assigning a smaller threshold to NRT-VBR than ABR traffic. The scheduling is performed in this order in each cycle.

The service flow of the MDWRR algorithm is defined by the pseudo-code in Fig. 6 The algorithm in Fig. 6 uses a few sets of sources to control the state transition of the sources: sets S_1, S_2, and S_3 record the sources that are currently in State 1, 2, and 3; set S_0 records the sources that are the best-effort; sets S_{1R} and S_{2R} record the sources of real-time traffic (CBR, RT-VBR) that are in State 1 and 2; and set S_{NR} records the sources of non-real-time traffic (NRT-VBR, ABR). Sets S', S_0', S_1', S_2' and S_{NR}' are for temporary usage only. QLs_a and Ths_a represent the queue length and threshold of source s_a. Each CBR, RT-VBR, NRT-VBR, and ABR source is associated with a pair of counters, namely, P and M. At the beginning of each cycle, counter P is set

```
                    <variable definition >
  ┌N : the length of a cycle, in units of time-slots
  │S : the traffic source
  │Bp (S): peak cell rate (PCR) of traffic source s
  └BM (S): average cell rate (ACR) of traffic source s
          ( ABR traffic : minimum cell rate (MCR) )

                    < state definition >
  ┌S1 (State 1): Bp(S) > 0, BM(S) > 0
  │          →the source should be visited in the next round.
  │          ( CBR traffic and VBR, ABR traffic <=MCR )
  │S2 (State 2): Bp(S) > 0, BM(S) <= 0
  │          →the source is not visited as long as there are
  │             other sources
  │          staying in State 1. ( VBR traffic > MCR and ABR
  │          traffic <=PCR )
  └S2 (State 3): Bp(S) <= 0, BM(S) <= 0 or no cell arrival at the
                 moment
             →the source will not be visited in the subsequent
                 rounds.

                    < algorithm >
  /* The cycle length is N and a set of n sources {s1, s2, … ,sn}
     is given */
  step 1 : Let Pi = Bp(si) and Mi=BM(si), for all 1 <=i <=n
  step 2 : Let S0={best-effort sources}, S1={s1, s2, …,sn}–S0
           S2=S3=S0'=S2R= ϕ
           S1R={ the sources of real-time traffic (CBR, RT-VBR)
                in State 1 }
           SNR={ the sources of non-real-time traffic
                 (NRT-VBR, ABR) }
  step 3 : i =0, S0' = S0;

  while( i < N ) do {
      if( S1 ≠ϕ ) {
          S1 '=S1;
          while (S1 '  ≠ϕ ) do{
              SNR '=SNR;
              while( SNR '  ≠ϕ ) do{
                  select a source sa from SNR ';
                  if( SlR == ϕ )
                      THsa = O;
                  else
                      THsa = threshold value of sa;
                  if( ( S1' ∌ sa )&&(QLsa ≥THsa ) )
                      S1'= S1' + { sa };
                  if( ( S1' ∋ sa )&&(QLsa < THsa ) )
                      S1'= S1' – { sa };
                  SNR'= SNR'– { sa };
              }
              select a source sk from S1';
              visit sk;
              S1'= S1' – { sk }, i= i + 1;
              Pk= Pk – 1, Mk = Mk – 1;
              check the new state of sk;
              If the state transits into State 2,
              Then move sk from S1 to S2;
              If the state transits into State 3,
              Then move sk from S1 to S3;
          }
          goto round;
      }else if(S2 ≠ϕ ){
          S2' = S2;
          while(S2 ' ≠ϕ ) do{
              SNR' = SNR;
              while(SNR '  ≠ϕ ) do{
                  select a source sa from SNR ';
                  if(S2R == ϕ )
                      THsa = 0;
                  else
                      THsa = threshold value of sa;
                  if( ( S2' ∋ sa )&&(QLsa ≥THsa ) )
                      S2'= S2' + { sa };
                  if( ( S2 ' ∋ sa )&&(QLsa < THsa ) )
                      S2'= S2' – { sa };
                  SNR'= SNR'– { sa };
              }
              select a source sk from S2 ';
              visit sk;
              S2'= S2' – {sk}, i= i + 1;
              Pk= Pk – 1;
              check the new state of sk;
              If the state transits into State 3,
              Then move sk from S2 to S3;
          }
          goto round;
      }else{
          select a best effort source sk from S0;
          visit sk;
          S0' = S0' – {sk}, i = i + 1;
      }
      round :
      if(S0 ' == ϕ ), then {S0 ' = S0;}
      check the new state of each source in S3 for the
      next round;
  }
```

Fig. 6. The MDWRR cell scheduling algorithm.

to the peak cell rate $B_P(s)$ of the corresponding source s and counter M is set to the average cell rate $B_M(s)$. If source s is ABR traffic, counter M is set to the minimum cell rate $B_M(s)$. Counters P and M are decreased by 1 when the source is visited and a cell is transmitted successfully. In each round, each source falls into one of the three states (State 1, State 2, or State 3). The scheduling algorithm visits the sources in State 1 with first priority. The sources in State 2 have the second priority and best-effort sources have the lowest priority.

The schedule among traffic sources to be served in each round is as follows. The MDWRR scheduling algorithm checks whether non-real-time traffic exceeds the threshold. If the traffic does not exceed the threshold, the scheduling algorithm services only the real-time traffic based on the round robin (RR) method.

But, if the traffic exceeds the threshold, it services the non-real-time traffic exceeding the threshold and real-time traffic based on the RR method. The new arrival of cells may cause a transition of the state of the sources. The sources in State 3 are examined again in the beginning of each round so that the high-priority sources with new incoming cells will be visited in the next round. The best-effort sources have the lowest priority. They are visited only when there is no source in States 1 and 2. In order not to starve the future opportunities for CBR, RT-VBR, NRT-VBR, or ABR sources to be visited, at most one best-effort source is visited within each round. The schedule among several best-effort sources is round-robin. The MDWRR scheduling algorithm reduces the delay time of real-time traffic in this way. It can also reduce the cell loss of non-real-time traffic, which may occur by serving real-time traffic preferentially. The CBR traffic does not have to be served above the PCR, because it requires constant bandwidth for real-time images or voice traffic.

Though the MDWRR scheduling algorithm may be more complex than the conventional DWRR scheme because of the cell transmission procedure based on delay priority, considering the delay priority minimizes cell delay and decreases the required size for the temporary buffer.

IV. SIMULATION AND ANALYSIS OF THE RESULTS

Our simulation model consists of a queue for each of the five traffic sources and a scheduler (Fig. 7). Each queue is used to keep the traffic of diverse classes according to the VCC.

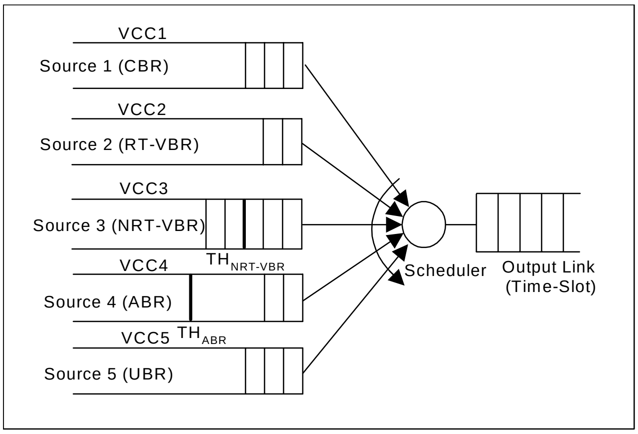

Fig. 7. Simulation model.

We used the simulation tool known as UltraSAN to build a simulation model and evaluate the performance of the model. The simulation environments are defined as follows. The transmission rate of one outgoing channel is set to 1.55 Mbps, and there are five input VCCs. Each VCC has a temporary buffer for accumulating cells. Input traffic consists of five diverse classes, but the traffic considered here is the CBR, RT-VBR, NRT-VBR, and ABR traffic. Each class of traffic is defined in Table 2. For CBR traffic, the amount of arrival traffic and length of the cell interval are constant. Thus, CBR traffic is modeled as a deterministic distribution process. For RT-VBR, NRT-VBR, and ABR traffic, the amount of arrival traffic and length of the cell interval are variable and have a bursty property. RT-VBR, NRT-VBR and ABR traffic are modeled as ON-OFF source models (Fig. 8), which consist of two states, namely, the ON state and the OFF state. In the ON state the source generates a Poisson flow of packets as rate λ, while in the OFF state the source generates no packets [19], [20]. Thus, the traffic source, modeled as an ON-OFF source, generates cells at the peak cell rate in the ON state and keeps silent in the OFF state. The source is thus characterized by alternating independent ON (burst) and OFF (silent) periods. The duration of the ON or OFF period can be characterized by any general distribution function. We assume that the duration of the ON period between two OFF periods is determined by a

Table 2. Traffic model of each source.

CBR	Deterministic Distribution, λ =0.013
RT-VBR	ON-OFF, λ = 0.016, $E[t_{on}]$ = 1.25, $E[t_{off}]$ = 0.8
NRT-VBR	ON-OFF, λ = 0.0125, $E[t_{on}]$ = 0.83, $E[t_{off}]$ = 1.25
ABR	ON-OFF, λ = 0.016, $E[t_{on}]$ = 0.5, $E[t_{off}]$ = 0.5

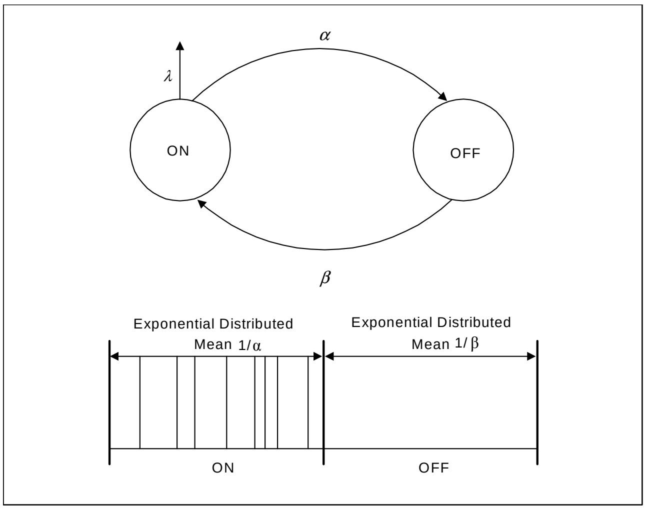

Fig. 8. ON-OFF source model.

random variable t_{on}, which is exponentially distributed. Similarly, the duration of the OFF period between two ON periods is determined by a random variable t_{off}, which is also exponentially distributed. Therefore, the expectation of the period duration is denoted as $E[t_{on}]$ for the ON period and $E[t_{off}]$ for the OFF period. Assume that the cell arrival probability in the ON state is determined by a Poisson random variable X with parameter λ, where E[X] = λ. Let ρ represent the utilization of the time–slot for a traffic source (the percentage of a slot being used). Then, we have

$$\rho = \frac{\lambda E[t_{on}]}{E[t_{on}] + E[t_{off}]}.$$

Given a cycle with length N and a traffic source with a claimed peak cell rate B_P and an average cell rate B_M, we have

$$B_P = \lambda N$$

and

$$B_M = \rho N = \frac{\lambda N E[t_{on}]}{E[t_{on}] + E[t_{off}]}.$$

The stochastic activity network (SAN) model of simulation consists of six subnets: CBR_traffic, rtVBR_traffic, nrtVBR_traffic, ABR_traffic, scheduler, and scheduler1 (Fig. 9).

The SANs consist of four primitive objects: places, activities, input gates, and output gates. Places are represented graphically as circles and are used to represent the state of the modeled system. Each place contains a certain number of tokens, which represent the marking of the place. Activities represent actions in the modeled system that take some specified amount of time to complete. They are of two types: timed and instantaneous. Timed activities are represented graphically as hollow ovals and represent actions in the modeled system that take time to complete. Instantaneous activities are represented graphically as vertical lines and represent actions that are completed in a negligible amount of time compared to the other activities in the modeled system. Activities can have case probabilities associated with them. Case probabilities, represented graphically as circles on the right side of an activity, represent uncertainty associated with the completion of that activity. Each case stands for a possible outcome. Input gates are represented graphically as triangles with their points connected to the activity they control. They are used to control the enabling of activities and define the marking changes that will occur when an activity is completed. Output gates are represented graphically as triangles with their flat side connected to an activity or a case. Like input gates, output gates are used to change the state of the modeled system upon the completion of an activity, but the only difference is that output gates are associated with a single case [21].

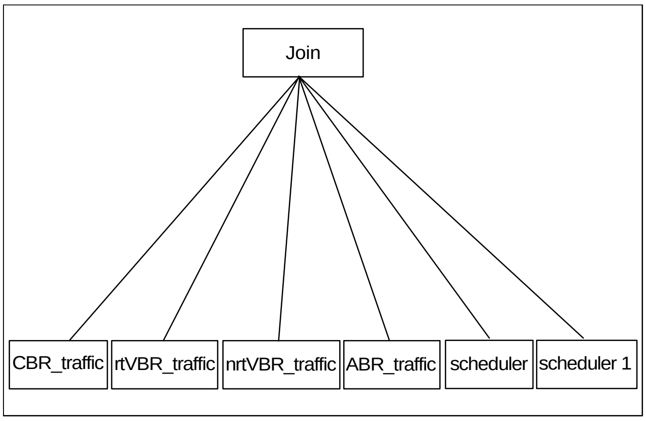

Fig. 9. The composed model.

Figure 10 depicts the model of a CBR_traffic subnet. The CBR_traffic subnet represents CBR traffic modeled as a deterministic distribution.

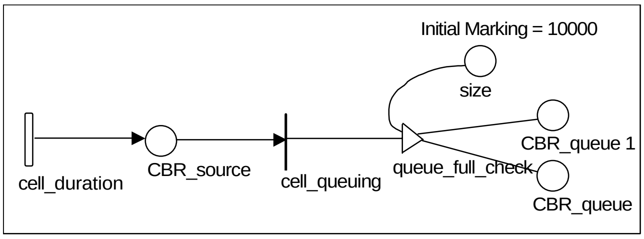

Fig. 10. A CBR_traffic subnet.

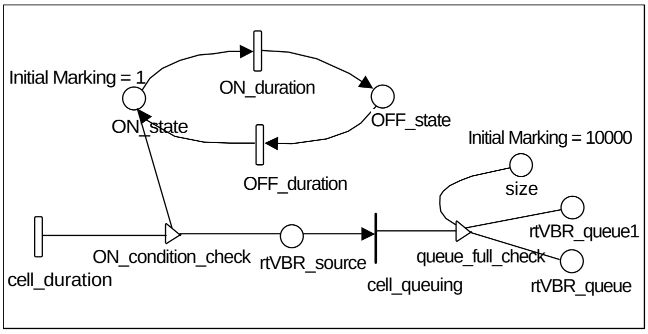

Fig. 11. An rtVBR_traffic subnet.

Figure 11 shows the model of an rtVBR_traffic subnet. The rtVBR_traffic subnet represents RT-VBR traffic modeled as an ON-OFF source.

The models of nrtVBR_traffic and ABR_traffic subnets are similar to those of the rtVBR_traffic subnet. Figures 12 through 14 show the model of a scheduler subnet divided into three modules. The scheduler subnet models the MDWRR scheduling algorithm. The MDWRR scheduling algorithm checks whether non-real-time traffic exceeds the threshold. If the traffic does not exceed the threshold, the scheduling algorithm serves only the real-time traffic based on the RR method, but if the traffic exceeds the threshold, it serves the non-real-time traffic exceeding the threshold and real-time traffic based on the RR method. The starting part of the scheduler subnet checks whether the cell is in a traffic queue. If it is, the scheduling service is begun. The scheduling part of the scheduler subnet serves cells to the MDWRR scheduling procedure. The average/peak cell check part of the scheduler subnet checks whether each of the served classes of traffic is in the state of average cell or peak cell.

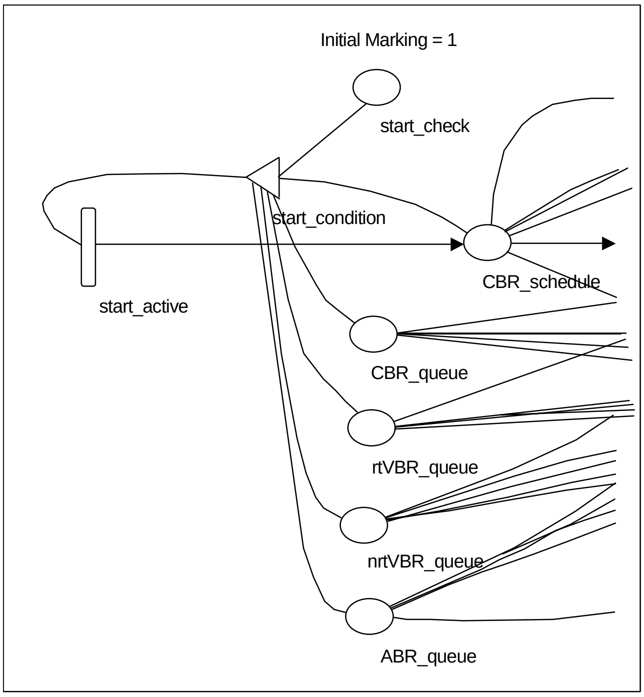

Fig. 12. The start part of the scheduler subnet.

The scheduler1 subnet models the conventional DWRR scheduling algorithm and the model of the scheduler1 subnet is similar to that of the scheduler subnet. The only difference is that the scheduler1 subnet uses the conventional DWRR algorithm for the scheduling procedure.

The results of our simulation are given in Tables 3 through 6 and Figures 15 through 18. Table 3 presents the average queue length of each traffic source in the DWRR and MDWRR scheduling algorithms, when the threshold of NRT-VBR and ABR traffic are set to 12 and 20, respectively. The queue length is related to the delay time, which is defined as the interval from the point of time at which the cell enters a queue to the point of time in which a transmission begins. A smaller queue length means a smaller time during which a cell stays in a queue. It means a smaller delay time. Thus, the MDWRR scheduling algorithm has a smaller delay time for processing than the DWRR algorithm in real-time traffic (Table 5). On the other hand, non-real-time traffic in the MDWRR scheduling algorithm has a longer queue length than the DWRR algorithm. Because some delay time is tolerable in non-real-time traffic, the MDWRR can keep more cells in the queue without cell loss occurring.

Figure 15 and Table 4 show the results that compare the delay time of the MDWRR scheduling algorithm with that of the DWRR algorithm in the RT-VBR traffic of real-time traffic (CBR, RT-VBR), when the threshold of the NRT-VBR and ABR traffic are set to 12 and 20, respectively. As the figure and table show, the MDWRR scheduling algorithm has a smaller delay time than the conventional DWRR algorithm in RT-VBR traffic. Incidentally, CBR traffic, which is one of the classes of real-time traffic, does not have to be above the PCR, because it requires a constant bandwidth for real-time images or voice traffic. Therefore, we compare the delay time of the MDWRR scheduling algorithm with that of the DWRR

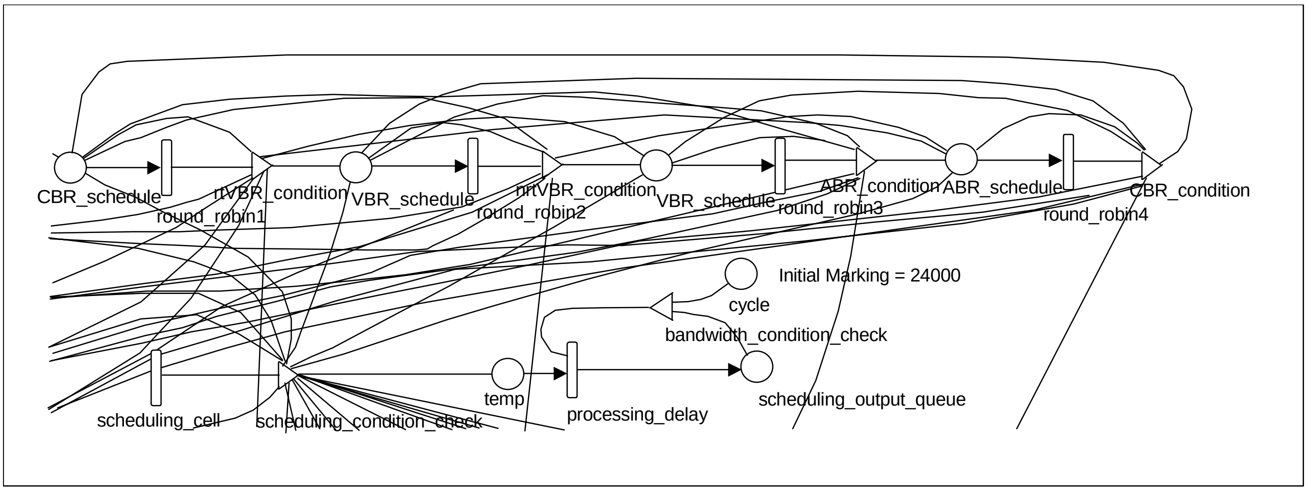

Fig. 13. The scheduling part of the scheduler subnet.

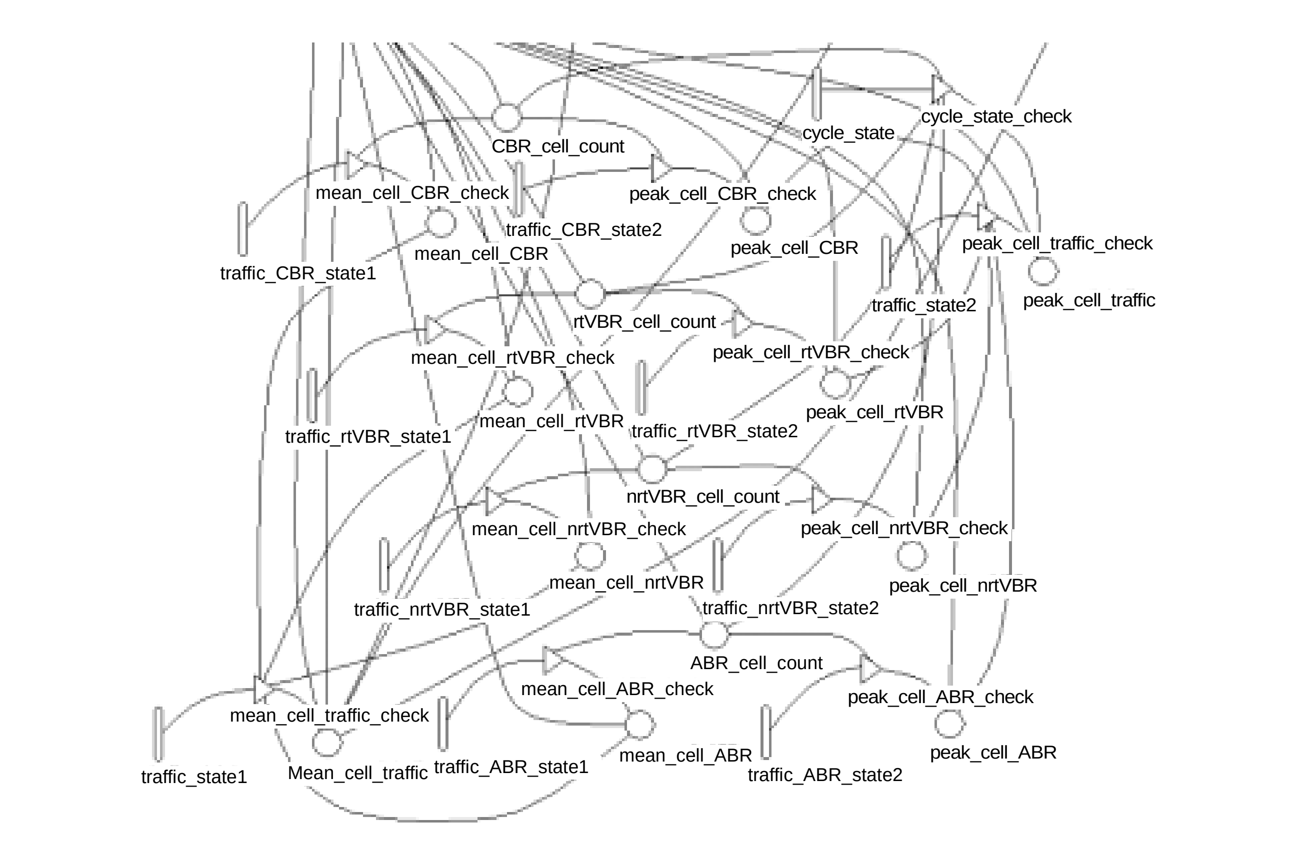

Fig. 14. The average/peak cell check part of the scheduling subnet.

Table 3. A comparison of the average queue length of each traffic source.

Cell scheduling procedure / Traffic class	MDWRR cell scheduling	DWRR cell scheduling
The average queue length of CBR traffic	16.19	18.06
The average queue length of RT-VBR traffic	8.05	13.49
The average queue length of NRT-VBR traffic	14.25	7.74
The average queue length of ABR traffic	20.03	14.32

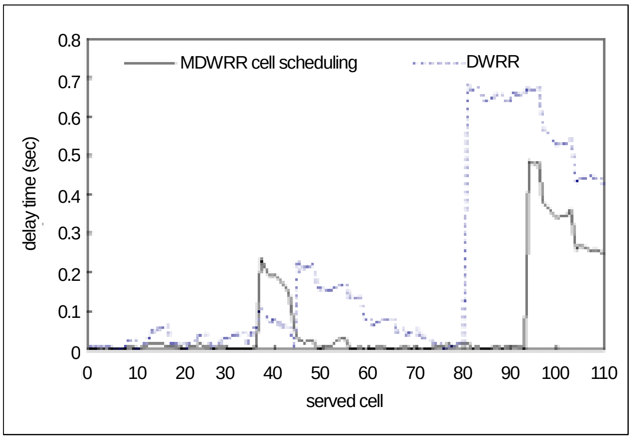

Fig. 15. The delay time of the served cell in the RT-VBR traffic.

algorithm in only the RT-VBR traffic.

Figures 16 through 18 depict a graph that compares the cell throughput when the threshold of NRT-VBR and ABR traffic are set to 12 and 20, respectively. Table 5 compares the average cell throughput of the MDWRR scheduling algorithm with that of the DWRR algorithm for each class of traffic. As these figures and table show, the cell throughput of RT-VBR traffic is higher than that in the DWRR algorithm, because

Table 4. A comparison of the average cell delay time of the RT-VBR traffic.

Cell scheduling procedure / Traffic class	MDWRR cell scheduling	DWRR cell scheduling
The average cell delay time of RT-VBR traffic	0.091726s	0.246275s

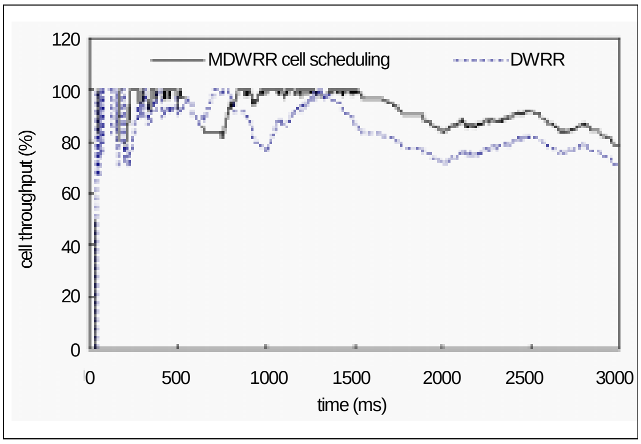

Fig. 16. The cell throughput of time in the RT-VBR.

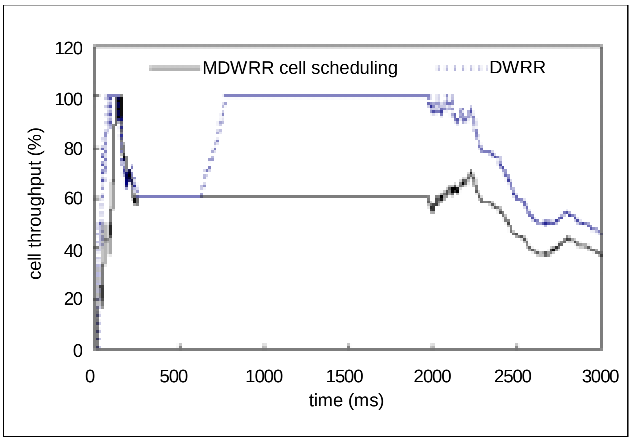

Fig. 17. The cell throughput of time in the NRT-VBR.

the MDWRR scheduling algorithm first serves RT-VBR traffic and then serves non-real-time traffic that does not exceed the threshold. On the other hand, the cell throughput of non-real-time traffic is lower than that in the DWRR algorithm.

Table 6 compares the average cell loss ratio of the MDWRR scheduling algorithm with that of the DWRR algorithm for each class of traffic. Here, the threshold of NRT-VBR and ABR traffic are set to 12 and 20, respectively. If the quantity of traffic transmitted to a sender is constant at some value, a

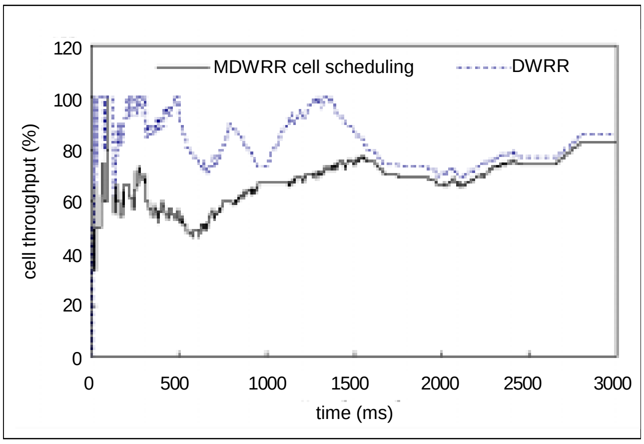

Fig. 18. The cell throughput of time in the ABR.

higher throughput of traffic means a lower loss ratio of traffic. Thus, the average cell loss ratio shows a converse relation to the average cell throughput.

V. CONCLUSIONS

The delay property is critical to real-time traffic of multimedia, because packet loss may occur because of a delay bound violation. To deal with this situation, we propose the MDWRR scheduling algorithm, which guarantees the delay property of real-time traffic, a feature that was not considered in the DWRR algorithm. We evaluated the algorithm's performance through computer simulation in an UltraSAN modeling environment. The MDWRR algorithm considers the delay property of real-time traffic by adding a cell transmission procedure based on delay priority. It also uses a threshold to non-real-time traffic to prevent the cell loss of non-real-time traffic due to the cell transmission procedure based on delay priority. Our simulation results show that the MDWRR algorithm minimizes cell delay and decreases the required size of the temporary buffer; this is an improved performance over the DWRR scheme.

In this paper, there was no proper criterion for threshold setting of non-real-time traffic. In order to select an appropriate

Table 5. A comparison of average cell throughput for each class of traffic.

Traffic class \ Cell scheduling procedure	MDWRR cell scheduling	DWRR cell scheduling
The average cell throughput of RT-VBR traffic	91.34%	83.43%
The average cell throughput of NRT-VBR traffic	56.82%	80.80%
The average cell throughput of ABR traffic	68.28%	82.62%

Table 6. A comparison of the average cell loss ratio for each class of traffic.

Traffic class \ Cell scheduling procedure	MDWRR cell scheduling	DWRR cell scheduling
The average cell loss ratio of RT-VBR traffic	0.18%	1.73%
The average cell loss ratio of NRT-VBR traffic	6.33%	3.95%
The average cell loss ratio of ABR traffic	2.71%	1.43%

value for the threshold, we performed the simulation several times, changing the threshold. Through this, we discovered the importance of threshold setting. If the threshold is set too high, the delay property is guaranteed in real-time traffic, but more loss can occur because of overflow in non-real-time traffic. If the threshold is set too low, loss can be avoided in non-real-time traffic, but the delay property cannot be guaranteed in real-time traffic. Thus, it is important to select the best value for the threshold that can guarantee the delay property of real-time traffic while avoiding the cell loss of non-real-time traffic. Because the threshold value depends on the characteristics of service traffic, the threshold value will change whenever a new virtual connection is established. Selecting the best value for the threshold will be the object of study for a future investigation.

REFERENCES

[1] Tae Il Jeong, Jae Ho Jeong, and Sung Jo Kim, "An Efficient Scheduling Mechanism Using Multiple Thresholds for Multimedia Traffic in ATM Switching Nodes," *22nd IEEE Conf. on Local Computer Networks* (*LCN'*97), Nov. 1997.

[2] "Traffic Management Specification Version 4.0," *ATM Forum*, Apr. 1996.

[3] M. Vishnu and J.W. Mark, "Reference Queue Tracking Strategies for Delay Guarantees in ATM Networks," *Tech. Rep. CWC-08, Centre for Wireless Communications*, Dept. of Electrical & Computer Engineering, Univ. of Waterloo, Sept. 1999.

[4] J.J. Bae and T. Suda, "Survey of Traffic Control Protocol in ATM Networks," *IEEE Globecomm* 300.1.1, 1990, pp. 1-6.

[5] D. Hong and T. Suda, "Congestion Control and Prevention in ATM Networks," *IEEE Network Magazine*, July 1991, pp. 10-16.

[6] G.M. Lee et al., "Flow-Based Admission Control Algorithm in the DiffServ-Aware ATM-Based MPLS Network," *ETRI J.*, vol. 24, no. 1, Feb. 2002, pp. 43-55.

[7] M. Vishnu and J.W. Mark, *HOL-EDD: A Novel Service Scheduling Scheme for ATM Networks*, Dept. of Electrical and Computer Engineering, Univ. of Waterloo, Waterloo, ON, Canada, N2L 3G1, June 1997.

[8] S. Archambault, "Performance Analysis of Per-VC Queuing," *Proc. of IEEE Globecomm'*96, vol. 3, Nov. 1996, pp. 1721-1725.

[9] L. Zhang, "Virtualclock: A New Traffic Control Algorithm for Packet Switched Networks," *ACM Trans. on Computer Systems*, vol. 9, no. 2, May 1991, pp. 101-124.

[10] S. Demers, A. Keshav, and S. Shenker, "Analysis and Simulation of a Fair Queuing Algorithm," *Internet Research and Experience*, 1, 1990.

[11] S.J. Golestani, "A Self-Clocked Fair Queuing Scheme for Broadband Applications," *Proc. of IEEE INFOCOM*, June 1994, pp. 636-646.

[12] B.-H. Choi et al., "Rate Proportional SCFQ (RP-SCFQ) Algorithm for High-Speed Packet-Switched Networks," *ETRI J.*, vol. 22, no. 3, Sept. 2000, pp. 1-9.

[13] M. Katevenis et al., "Weighted Round-Robin Cell Multiplexing in a General-Purpose ATM Switch Chip," *IEEE J. of Selected Areas in Comm.*, vol. 9, no. 8, Oct. 1991, pp. 1265-1279.

[14] Chiung-Shien, Gin-Kou Ma, and Bao-Shuh P. Lin, "A Cell Scheduling Algorithm for VBR Traffic in an ATM Multiplexer," *Globecomm'*95, Singapore, Sept. 1995, pp. 632-637.

[15] Hideyuki Shimonishi, Hiroshi Suzuki, "Performance Analysis of Weighted Round Robin Cell Scheduling and its Improvement in ATM Networks," *IEICE Trans. Comm.*, vol. E81 B, no. 5, May 1998.

[16] Sung Won Lee, Dong Ho Cho, and Yong Ki Park, "Improved Dynamic Weighted Cell Scheduling Algorithm Based on Earliest Deadline First Scheme for Various Traffics of ATM Switch," *Proc. IEEE Globecomm'*96, 1996, pp. 1959-1963.

[17] M. Katevenis, S. Sidiropoluos, and C. Courcoubetis, "Weighted Round-Robin Cell Multiplexing in a General Purpose ATM Switch Chip," *IEEE J. on Selected Area in Comm.*, vol. 9, no. 8, Oct. 1991.

[18] Jae Yong Lee and Young Han Kim, "Performance Analysis of a Hybrid Priority Control Scheme for Input and Output Queuing ATM Switches," *Proc. of the Conf. on Computer Communications* (*IEEE INFOCOM*), Mar/Apr.1998, pp. 1470.

[19] K. Sohraby, "Flow and Admission Control of ON-OFF sources in High Speed Networks," *IEEE Globecomm'*94 *Tutorial* 7: *ATM Architecture and Performance*, Ch. 7, 1994.

[20] Y.M. Jang et al., "Estimation and Prediction-Based Connection Admission Control in Broadband Satellite Systems," *ETRI J.*, vol. 22, no. 4, Dec. 2000, pp. 40-50.

[21] http://www.crhc.uiuc.edu/PERFORM/UltraSAN.html.

Ji-Young Kwak received the BS and MS degrees in computer engineering from Chonnam National University, Gwangju, Korea, in 1999 and 2001. She joined the Electronics and Telecommunication Research Institute (ETRI), Korea, in 2001 and is currently working as a Research Engineer in the Mobile Collaboration Research Team of Computer Software Technology Laboratory. Her current research areas of interest include traffic control, scheduling, QoS management, and Voice over IP Protocols.

Ji-Seung Nam received the BS degree from Inha University, Incheon, Korea, in 1981, the MS degree from The University of Alabama in 1986, and the PhD degree from The University of Arizona in 1992. From 1992 to 1995, he had been with ETRI, Korea. Since 1995, he has been an Associate Professor in the Department of Computer Engineering, Chonnam National University. His current research interests are in the area of real time network protocols, Quality of Service, and multimedia communications.

Doo-Hyun Kim received his Bachelor's degree from the Seoul National University (1985) and Master degree from Korea Advanced Institute of Science and Technology (1987). He is now a Principal Research Staff of Electronics and Telecommunications Research Institute (ETRI), and also PhD Candidate of Computer Science Department of KAIST. He was an International Fellow at SRI International from 1991 to 1993. He joined a national project developing a governmental high-speed multimedia conferencing system from 1996 to 1997. And he is now a Project Leader of a governmental project called GIANT (Gigabit Information And Network Technologies) developing a multimedia collaboration middleware based on ITU-T H.323 and T.120 recommendations. Mr. Kim's research interests include performance analysis of Voice over IP, distributed multimedia system, multimedia middleware, multimedia operating system, home appliance multimedia service, Quality of Service management, Scaleable Reliable Multimedia Multicast, and real-time multimedia support for WWW.