Proceedings of the International Conference on Computer Communications and Networks
Miami, Florida, USA, October 14–16, 2002

On the Latency Bound of Deficit Round Robin

Salil S. Kanhere and Harish Sethu
Department of ECE, Drexel University
3141 Chestnut Street, Philadelphia, PA 19104-2875.
E-mail: {salil,sethu}@ece.drexel.edu

***Abstract*— The emerging high-speed broadband packet-switched networks are expected to simultaneously support a variety of services with different Quality-of-Service (QoS) requirements over the same physical infrastructure. Fair packet scheduling algorithms used in switches and routers play a critical role in providing these QoS guarantees. Deficit Round Robin (DRR), a popular fair scheduling discipline, is very efficient with an O(1) dequeuing complexity. Using the concept of *Latency-Rate* ($\mathcal{LR}$) servers introduced by Stiliadis and Varma, we obtain an upper bound on the latency of DRR and prove that our bound is tight. Our upper bound is lower than the previously known upper bound. This illustrates that the DRR scheduler has better performance characteristics than previously believed, especially for real-time applications where the latency plays a role in the size of the playback buffer required.**

I. Introduction

A number of new applications such as video-on-demand and distance learning rely on traffic scheduling algorithms in the switches and routers to guarantee performance bounds and meet the Quality-of-Service (QoS) requirements. Given multiple packets awaiting transmission through an output link, the function of the scheduler is to determine the exact sequence in which these packets will be transmitted. Scheduling algorithms can be broadly classified into two categories – sorted-priority schedulers and frame-based schedulers. Sorted-priority schedulers such as Weighted Fair Queueing (WFQ) [1], Self-Clocked Fair Queueing (SCFQ) [2], Start-time Fair Queueing (SFQ) [3] and Worst-case Fair Weighted Fair Queueing (WF^{2}Q) [4] maintain a global variable known as the virtual time or the system potential function. This variable is used to compute the priority of each packet which is known as the timestamp. While they achieve good fairness and low latencies, they are not very efficient due to the complexity involved in computing the virtual time and the complexity of maintaining a sorted list of packets based on their timestamps. On the other hand, in frame-based schedulers such as Deficit Round Robin (DRR) [5] and Elastic Round Robin (ERR) [6], the scheduler simply visits all non-empty queues in a round-robin order. The service received by a flow in a round robin service opportunity is proportional to its fair share of the bandwidth. These schedulers do not have to maintain a global virtual time function and also do not have to perform sorting among the packets. The round robin service order of the frame-based schedulers reduces their per-packet work complexity to $O(1)$ with respect to the number of flows, rendering them attractive for implementation in routers and switches.

In recent work by Stiliadis and Varma [7, 8], the authors define a general class of schedulers, called *Latency-Rate* ($\mathcal{LR}$) servers. The authors also develop and define a notion of latency and determine an upper bound on the latency for a number of schedulers that belong to the class of $\mathcal{LR}$ servers. This notion of latency is based on the length of time it takes a new flow to begin receiving service at its guaranteed rate and therefore, is directly relevant to the size of the playback buffers required in real-time streaming applications.

DRR is one of the most popular frame-based fair scheduling disciplines that is now employed in a number of real environments involving fair scheduling, including Cisco routers and Microsoft's Windows NT. It is shown in [8] that the DRR scheduler belongs to the class of $\mathcal{LR}$ servers. Stiliadis and Varma report an upper bound on the latency of DRR [7], and its derivation is detailed in [8]. In this paper, we obtain a lower value of the upper bound on the latency of DRR and thus show that the scheduler has better performance characteristics than previously believed. We also show that our upper bound on the DRR latency is tight.

The rest of this paper is organized as follows. In Section 2, we briefly introduce the DRR scheduling discipline. In Section 3, we present our analysis of the latency bound of DRR. Section 4 presents a detailed comparison between our latency bound and the previously known bound derived in [8]. Finally, Section 5 concludes the paper.

II. Deficit Round Robin

In this section, we present a brief overview of the DRR scheduler, a detailed description of which can be found in [5].

Consider an output link of transmission rate r, access to which is controlled by the DRR scheduler. Let n be the total number of flows and let ρ_i be the reserved rate for flow i. Let ρ_{min} be the lowest of these reserved rates. Note that since

This work was supported in part by the NSF CAREER Award CCR-9984161 and U.S. Air Force Contract F30602-00-2-0501.

all the flows share the same output link, a necessary constraint is that the sum of the reserved rates be no more than the transmission rate of the output link. In order that each flow receives service proportional to its guaranteed service rate, the DRR scheduler assigns a weight to each flow. The weight assigned to flow i, w_i is given by,

$$w_i = \frac{\rho_i}{\rho_{min}} \tag{1}$$

Note that $\forall i \in n, w_i \geq 1$.

A flow is said to be *active* during a certain time interval, if it always has packets awaiting service during this interval. The DRR scheduler maintains a linked list of the active flows, called the *ActiveList*. At the start of an active period of a flow, the flow is added to the tail of the *ActiveList*. A *round* is defined as one round robin iteration during which the DRR scheduler serves all the flows that are present in the *ActiveList* at the outset of the round. Each active flow is assigned a *quantum* by the DRR scheduler. The quantum allocated to a flow is defined as the service that the flow should receive during each round robin service opportunity. Let Q_i represent the quantum assigned to flow i and let Q_{min} be the quantum assigned to the flow with the lowest reserved rate. The quantum assigned to flow i, Q_i is given by $w_i Q_{min}$. Thus, the quanta assigned to the flows are in proportion of their reserved rates. In order that the work complexity of the DRR scheduler is O(1), it is necessary that Q_{min} should be greater than or equal to the size of the largest packet that may potentially arrive during the execution of the scheduler. Note that during some service opportunity, a flow may not be able to transmit a packet because doing so would cause the flow to exceed its allocated quantum. The scheduler maintains a per-flow state, the *deficit count*, which records the difference between the amount of data actually sent thus far, and the amount that should have been sent. This deficit is added to the value of the quantum in the next round as the amount of data the scheduler should try to schedule in the next round. Thus, a flow that received very little service in a certain round is given an opportunity to receive more service in the next round.

A frame is defined as the sum of the quanta allocated to all the active flows in a DRR round. Let F denote the size in bits of a DRR frame. The upper bound of the latency of DRR is derived in [8] as $(3F - 2\phi_i)/r$, where r represents the transmission rate of the output link. In the following section, we prove a tighter bound on the latency. Our bound is tighter due to the use of a tighter upper bound on the deficit count of a flow and also due to the fact that we make a distinction between the size of the largest packet that may potentially arrive at a scheduler and the size of the largest packet that actually arrives during an execution of the scheduler.

III. Latency Bound of DRR

In deriving an upper bound on the latency of DRR, we borrow the technique used in [8] based on the concept of $\mathcal{LR}$-servers first proposed in [7]. In the following, we provide some definitions and review some others that will be useful in defining the latency of guaranteed rate schedulers and our analysis of the latency bound.

Definition 1: Define W as the sum of the weights of all active flows that are being served by the DRR scheduler.

Definition 2: Define m as the size in bits of the largest packet that is actually served during the execution of a scheduling algorithm. Define M as the size in bits of the largest packet that may potentially arrive during the execution of a scheduling algorithm. Note that, $M \geq m$.

Definition 3: An *active* period of a flow is defined as the maximal interval of time during which it has at least one packet awaiting service or in service.

Definition 4: A *busy* period of a flow is defined as the maximal time interval during which the flow is active if it served exactly at its reserved rate.

The active period is different from the busy period of a flow, in the sense that it reflects the actual behavior of the scheduler since the instantaneous service offered to the flow varies according to the number of active flows. Let $Sent_i(t_1, t_2)$ represent the amount of service received by flow i during the time interval (t_1, t_2). Let the time instant α_i be the start of a busy period for flow i. Let $t > \alpha_i$ be such that flow i is continuously busy during the time interval (α_i, t). Define $S_i(\alpha_i, t)$ as the number of bits belonging to packets in flow i that arrive after time α_i and are scheduled during the time interval (α_i, t). Note that during this time interval, the scheduler may still be serving packets that arrived during a previous busy period, and hence $S_i(\alpha_i, t)$ is not necessarily the same as $Sent_i(\alpha_i, t)$. The reader is referred to [7, 8] for a detailed treatment of the differences between an active period and a busy period.

Definition 5: Define $\mathbf{T}$ as the set of all time instants at which the scheduler ends serving one flow and begins serving another. The set of all time instants at which a scheduler begins serving flow i is defined as $\mathbf{T}_i$. Note that the set $\mathbf{T}$ is the union of $\mathbf{T}_i$ for all active flows i.

Definition 6: The latency of a flow is defined as the minimum non-negative constant Θ_i that satisfies the following for all possible busy periods of the flow,

$$S_i(\alpha_i, t) \geq max\{0, \rho_i(t - \alpha_i - \Theta_i)\} \tag{2}$$

As defined in [7], a scheduler which satisfies Equation (2) for some non-negative constant value of Θ_i is said to belong to the class of *Latency Rate* ($\mathcal{LR}$) servers. The above definition captures the fact that the latency of a guaranteed-rate scheduler should not merely be the time it takes for the first packet of a flow to get scheduled, but should be a measure of the cumulative time that a flow has to wait until it begins receiving service at its guaranteed rate.

Note that even though the definition of the latency is based on flow busy periods, in practice it is easier to analyze scheduling algorithms based on the active period of a flow. Let τ_i be an instant of time when flow i becomes active. Let $t > \tau_i$ be some time instant such that the flow is continuously active during the time interval (τ_i, t). Let Θ'_i be the smallest non-negative constant such that the following equation is satisfied for all t,

$$Sent_i(\tau_i, t) \geq max\{0, \rho_i(t - \tau_i - \Theta'_i)\} \quad (3)$$

Even though (τ_i, t) may not be a continuously busy period for flow i, it is proved in [7] that the latency, as defined by (2), is bounded by Θ'_i. This allows one to determine the latency bound of a scheduler by considering only those periods during which a flow is continuously active.

Theorem 1: The DRR scheduler belongs to the class of $\mathcal{LR}$ servers, with an upper bound on the latency, Θ_i for flow i, given by:

$$\Theta_i \leq \frac{1}{r}\left((W - w_i)Q_{min} + (m-1)\left(\frac{W}{w_i} + n - 2\right)\right) \quad (4)$$

where n is the total number of active flows and r is the transmission rate of the output link.

Proof: Since the latency of an $\mathcal{LR}$ server can be estimated based on its behavior in the flow active period, we will prove the theorem by showing that,

$$\Theta'_i \leq \frac{1}{r}\left((W - w_i)Q_{min} + (m-1)\left(\frac{W}{w_i} + n - 2\right)\right)$$

Let flow i become active at time instant τ_i. In deriving an upper bound on the latency of DRR we consider a time interval (τ_i, t) during which flow i is continuously active. Then, we obtain the lower bound on the total service received by flow i during this time interval. Lastly, we express the lower bound in the form of Equation (3) to derive the latency bound.

In [9], in the context of deriving the latency bound of Elastic Round Robin [6], it is proved that if the upper bound of latency is met exactly during the active period (τ_i, t), then the following two conditions are satisfied:

1) $\tau_i \in \mathbf{T}$ and
2) $t \in \mathbf{T}_i$

It can be easily verified that these conditions are applicable in the analysis of the latency bound of all round robin schedulers including DRR. Let τ_i^k be the time instant marking the start of the k-th service opportunity of flow i. Note that τ_i^k belongs to the set $\mathbf{T}_i$. From the above, to determine a tight upper bound on the latency of the DRR scheduler we need to only consider time intervals (τ_i, τ_i^k) for all k. Fig. 1 illustrates the time interval under consideration for a given k. Note that the time instant τ_i may or may not coincide with the end of a round and the start of the subsequent round. Let k_0 be the round which is in progress at time instant τ_i or which ends exactly at time instant τ_i. Let the time instant t_h mark the end of round $(k_0 + h - 1)$ and the start of the subsequent round.

Let $Sent_i(s)$ represent the total data transmitted from flow i in round s of the DRR scheduler. Also, let $DC_i(s)$ represent the deficit count of flow i following its service in round s. It is proved in [5] that for any flow i in any round s,

$$0 \quad \leq \quad DC_i(s) \quad \leq m - 1 \quad (5)$$
$$Sent_i(s) \quad = \quad w_i Q_{min} + DC_i(s-1) - DC_i(s) \quad (6)$$

Note that the upper bound of $DC_i(s)$ is Q_i, as used in [8] in the derivation of the DRR latency, only if $Q_i = M = m$. In all other situations, the upper bound on the deficit count as specified by Equation (5) is a tighter bound.

As illustrated in Fig. 1, assume that the time instant when flow i becomes active coincides with the time instant when some flow u is about to start its service opportunity during the k_0-th round. Let G_a denote the set of flows which receive service during the time interval (τ_i, t_1), i.e., *after* flow i becomes active. Similarly, let G_b denote the set of flows which are served by the DRR scheduler during the time interval (t_0, τ_i), i.e., *before* flow i becomes active. Note that flow i is not included in either of these two sets since flow i will receive its first service opportunity only in the $(k_0 + 1)$-th round. If the time instant τ_i coincides with the time instant t_1, which marks the end of the k_0-th round and the start of the (k_0+1)-th round, then the set G_a will be empty and all the $n-1$ flows will be included in the set G_b. Note that in this case, flow i will be the last to receive service in the $(k_0 + 1)$-th round and all subsequent rounds during the time interval under consideration.

The first step towards analyzing the latency bound involves obtaining an upper bound on the size of the time interval (τ_i, τ_i^k). This time interval can be split into the following three sub-intervals:

1) (τ_i, t_1): This sub-interval includes the part of the k_0-th round during which all the flows belonging to the set G_a will be served by the DRR scheduler. Summing Equation (6) over all these flows,

$$\begin{aligned} t_1 - \tau_i \quad &= \quad \frac{1}{r}\sum_{j \in G_a}\{w_j Q_{min} + \\ &\quad + DC_j(k_0 - 1) - DC_j(k_0)\} \end{aligned} \quad (7)$$

2) (t_1, t_k): This sub-interval includes $k-1$ rounds of the DRR scheduler starting at round $(k_0 + 1)$. Consider the time interval (t_h, t_{h+1}) when round $(k_0 + h)$ is in progress. Summing Equation (6) over all n flows and since W is the sum of all the flow weights, we have,

$$\begin{aligned} t_{h+1} - t_h \quad &= \quad \frac{W}{r}(Q_{min}) \\ &\quad + \frac{1}{r}\sum_{j=1}^{n}\{DC_j(k_0 + h - 1) - DC_j(k_0 + h)\} \end{aligned}$$

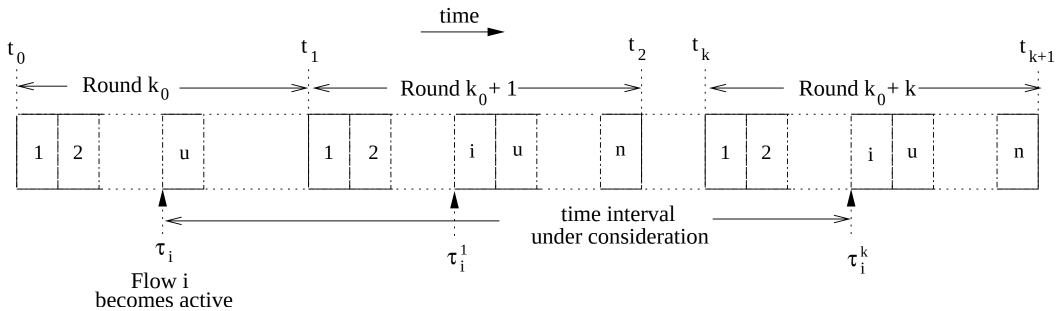

Fig. 1. An illustration of the time interval under consideration

Summing the above over $k-1$ rounds beginning with round k_0+1,

$$t_k - t_1 = \frac{W}{r}(k-1)Q_{min} + \frac{1}{r}\sum_{j=1}^{n}\{DC_j(k_0) - DC_j(k_0+k-1)\} \quad (8)$$

3) (t_k, τ_i^k): This sub-interval includes the part of the (k_0+k)-th round during which all the flows belonging to the set G_b will be served by the DRR scheduler. Summing (6) over all these flows,

$$\tau_i^k - t_k = \frac{1}{r}\sum_{j\in G_b}\{w_j Q_{min} + DC_j(k_0+k-1) - DC_j(k_0+k)\} \quad (9)$$

Combining Equations (7), (8) and (9) and since W is the sum of the weights of all the n flows, we have,

$$\begin{aligned}\tau_i^k - \tau_i &= \frac{W}{r}(k-1)Q_{min} + \left(\frac{W-w_i}{r}\right)Q_{min} \\ &+ \frac{1}{r}\sum_{j\in G_a}(DC_j(k_0-1) - DC_j(k_0+k-1)) \\ &+ \frac{1}{r}\sum_{j\in G_b}(DC_j(k_0) - DC_j(k_0+k)) \\ &+ \frac{1}{r}(DC_i(k_0) - DC_i(k_0+k-1)) \quad (10)\end{aligned}$$

Now since flow i becomes active during round k_0, its deficit count at the end of the k_0-th round, $DC_i(k_0)$ is equal to zero. Using this fact and the bounds on the deficit count from Equation (5) in Equation (10), we have,

$$\tau_i^k - \tau_i \leq \frac{W}{r}(k-1)Q_{min} + \left(\frac{W-w_i}{r}\right)Q_{min} + \frac{(n-1)(m-1)}{r} - \frac{1}{r}DC_i(k_0+k-1)$$

Solving for $(k-1)$,

$$\begin{aligned}(k-1) &\geq \frac{r}{WQ_{min}}(\tau_i^k - \tau_i) - \frac{W-w_i}{W} \\ &- \frac{1}{WQ_{min}}(n-1)(m-1) \\ &+ \frac{1}{WQ_{min}}DC_i(k_0+k-1) \quad (11)\end{aligned}$$

Note that during the time interval under consideration, (τ_i, τ_i^k), flow i receives service in $(k-1)$ rounds starting at round (k_0+1). Hence, using Equation (6) over these $(k-1)$ rounds of service for flow i, and since the deficit count of a newly active flow is 0, we get,

$$Sent_i(\tau_i, \tau_i^k) = w_i(k-1)Q_{min} - DC_i(k_0+k-1) \quad (12)$$

Using (11) to substitute for $(k-1)$ in (12), we get,

$$\begin{aligned}Sent_i(\tau_i, \tau_i^k) &\geq \frac{w_i r}{W}(\tau_i^k - \tau_i) - \frac{w_i}{W}(W-w_i)Q_{min} \\ &- \frac{wi}{W}(n-1)(m-1) + \frac{w_i}{W}DC_i(k_0+k-1) \\ &- DC_i(k_0+k-1) \quad (13)\end{aligned}$$

Now, since the reserved rates are proportional to the weights assigned to the flows as given by (1), and since the sum of the reserved rates is no more than the link rate r, we have,

$$\rho_i \leq \frac{w_i}{W}r \quad (14)$$

Using Equation (14) in Equation (13), we have,

$$\begin{aligned}Sent_i(\tau_i, \tau_i^k) &\geq \rho_i(\tau_i^k - \tau_i) - \frac{\rho_i}{r}(W-w_i)(Q_{min}) \\ &- \frac{\rho_i}{r}(n-1)(m-1) \\ &- \frac{\rho_i}{r}\left(\frac{W-w_i}{w_i}\right)DC_i(k_0+k-1) \quad (15)\end{aligned}$$

Simplifying further and noting that the latency bound reaches the upper bound when $DC_i(k_0+k-1)$ equals $(m-1)$ we get,

$$Sent_i(\tau_i, \tau_i^k) \geq max\Bigg\{0, \rho_i\Bigg(\tau_i^k - \tau_i - \frac{1}{r}\Bigg((W - w_i)Q_{min} + (m-1)\Bigg(\frac{W}{w_i} + n - 2\Bigg)\Bigg)\Bigg)\Bigg\} \quad (16)$$

As discussed earlier, flow i will experience its worst latency during an interval (τ_i, τ_i^k) for some k. Therefore, from Equation (16), the statement of the theorem is proved. □

We now proceed to show that the latency bound given by Theorem 1 is tight by illustrating a case when the bound is actually achieved. Assume that flow i becomes busy at a certain time instant τ_i, which also coincides with the start of a certain round $(k_0 + 1)$. Since the other flows in the *ActiveList* will be served first, flow i becomes backlogged instantly and τ_i is also the start of its active period. Assume that for any time instant t, $t \geq \tau_i$, a total of n flows, including flow i, are active. Let $\mathcal{F}$ represent the set of all n flows. Also, assume that the summation of the reserved rates of all the n flows equals the output link transmission rate, r. Hence $\rho_i = \frac{w_i}{W}r$. Since flow i became active at time τ_i, its deficit count at the start of round $(k_0 + 1)$ is 0. Let the deficit count of all the other flows at the start of round (k_0+1) be equal to $(m-1)$. From Equations (5) and (6), a flow j can transmit a maximum of $w_jQ_{min}+(m-1)$ bits during a round robin service opportunity. In the worst case, before flow i is served by the DRR scheduler, each of the other $(n-1)$ flows will receive this maximum service. Hence, the cumulative delay until flow i receives service, X, is given by,

$$\begin{aligned} X &= \frac{(\sum_{\substack{j \in \mathcal{F} \\ j \neq i}} w_j)(Q_{min}) + (n-1)(m-1)}{r} \\ &= \frac{(W - w_i)(Q_{min}) + (n-1)(m-1)}{r} \end{aligned} \quad (17)$$

Even though X represents the time for which flow i has to wait until its first packet is scheduled, Equation (2) does not hold true when X is substituted as Θ_i. This is because in the time interval $(\tau_i, \tau_i + X)$ flow i has not yet started receiving service at its guaranteed rate. We assume that the latency, Θ_i is given by,

$$\Theta_i = X + Y \quad (18)$$

A plot of the service received by flow i against time is illustrated in Fig. 2. Assume that the total service received by flow i during its first service opportunity is $w_iQ_{min}-(m-1)$. Note that from (5) and (6), this equals the minimum service that flow i can receive during any service opportunity. At the end of the (k_0+1)-th round, the deficit count for flow i is $(m-1)$ whereas the deficit count for all the other flows is zero. In the worst case, during the $(k_0 + 2)$-th round, each flow j from amongst the other $(n - 1)$ flows will transmit a maximum of w_jQ_{min} bits before flow i receives its second service opportunity. During this service opportunity, flow i will be able to transmit at least a minimum of w_iQ_{min} bits, and will thus start receiving service at its guaranteed rate. Referring to Fig. 2, we have,

$$Y + \frac{w_iQ_{min}}{\rho_i} - \frac{m-1}{\rho_i} = \frac{WQ_{min}}{r} - \frac{m-1}{r}$$

Now, since $\rho_i = \frac{w_i}{W}r$, simplifying further, we have,

$$Y = \frac{(m-1)}{r}\left(\frac{W - w_i}{w_i}\right) \quad (19)$$

Substituting for X and Y from Equations (17) and (19) in Equation (18), it can be easily verified that the latency bound is exactly met.

IV. A Discussion of the New Latency Bound

In this section, we present a brief but detailed comparison of the latency bound of DRR derived in Theorem 1 with the latency bound derived in [8]. Let Θ_i^{new} represent the latency bound derived in Theorem 1. Hence,

$$\Theta_i^{new} = \frac{(W - w_i)Q_{min}}{r} + \frac{1}{r}(m-1)\left(\frac{W}{w_i} + n - 2\right) \quad (20)$$

Let Θ_i^{old} represent the latency bound of DRR as derived in [8]. We have,

$$\Theta_i^{old} = \frac{3F - 2Q_i}{r} \quad (21)$$

In the above equation, F denotes the size of a DRR frame which is equal to WQ_{min}, the summation of the quanta of all the active flows. Substituting for F in Equation (21) and simplifying we have,

$$\Theta_i^{old} = \frac{(W - w_i)Q_{min}}{r} + \frac{(2W - w_i)Q_{min}}{r} \quad (22)$$

Note that both Θ_i^{old} and Θ_i^{new} are represented as a summation of two terms of which the first term is identical. Recall that $Q_{min} \geq M$ and that $M \geq m$. Therefore it can be easily verified that Θ_i^{new} is less than Θ_i^{old}, demonstrating that the latency bound of DRR as proved in this paper is a tighter bound. Also note that the expression for Θ_i^{old} does not distinguish between m and M. In most networks including the Internet, the vast majority of the packets in the traffic are of much smaller size than the maximum possible size [10, 11]. Therefore, since $m << M$, we have $m << Q_{min}$. Thus, in situations with small numbers of flows and where $m < M$, Θ_i^{new} can be much lower than Θ_i^{old}.

In order that the reader can fully appreciate the difference, we provide a comparison of these two latency bounds of DRR within the context of a practical example. Let us assume that the DRR scheduler is controlling access to an output link with

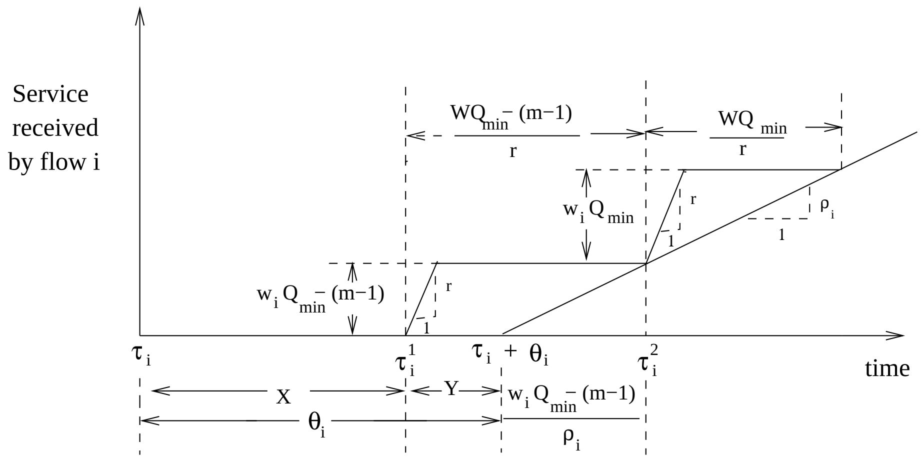

Fig. 2. Illustration of the latency bound

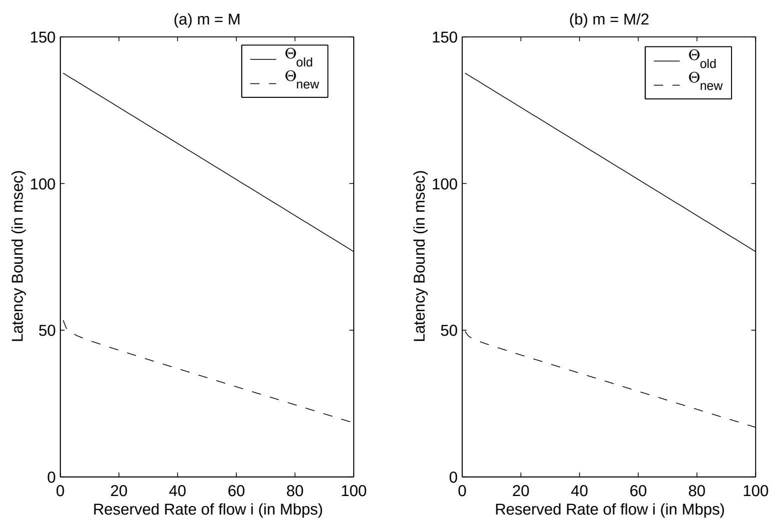

Fig. 3. Comparison of the latency bound

rate, $r = 150$ Mbps. Assume that M, the size of the largest packet that may potentially arrive during the execution of the DRR algorithm is equal to 576 bytes. Also, let Q_{min} be equal to M. Let ρ_{min} be equal to 0.1 Mbps and let the number of flows, n be equal to 100. Also assume that the output link is completely utilized, i.e. $\sum_{i=1}^{n} \rho_i = r$. Note that this implies that the sum of all the weights is $150/0.1 = 1500$. We compare the two latency bounds, Θ_i^{old} and Θ_i^{new} for flow i as a function of its reserved rate, ρ_i, for two values of m: (a) $m = M$, (b) $m = M/2$. Fig. 3 illustrates a plot of the latency bounds of flow i for both values of m. Note that expressions for Θ_i^{new} and Θ_i^{old} depend on the sum of the weights of all the flows but not on the distribution of the weights among all the flows other than weight i. Therefore, the weights of flows other than flow i are not discussed in the context of this illustration. Fig. 3 reveals that Θ_i^{new} is a tighter bound.

V. Conclusion

In this paper, we have derived an upper bound on the latency of Deficit Round Robin (DRR), a popular fair scheduling discipline that has found use in a number of commercial products including Cisco routers and Microsoft's Windows NT. Our bound is lower than the previously believed upper bound. We also show that our upper bound is a tight one. The latency experienced by a flow captures the length of time it takes a new flow to begin receiving service at the guaranteed rate, and therefore, it is directly relevant to the size of playback buffers needed for real-time applications. This paper shows that the DRR scheduler has better performance characteristics, and thus, is more suitable for both best-effort and guaranteed services environments than previously believed.

REFERENCES

[1] A. Demers, S. Keshav, and S. Shenker, "Design and analysis of a fair queuing algorithm," in *Proceedings of ACM SIGCOMM*, Austin, September 1989, pp. 1–12.

[2] S. J. Golestani, "A self-clocked fair queueing scheme for broadband applications," in *Proceedings of IEEE INFOCOM*, Toronto, Canada, June 1994, pp. 636–646.

[3] P. Goyal, H. M. Vin, and H.Cheng, "Start-time fair queueing; a scheduling algorithm for integrated services packet switched networks," *IEEE Transactions on Networking*, vol. 5, pp. 690–704, October 1997.

[4] J. C. R. Bennett and H. Zhang, "WF^2Q: Worst-case fair weighted fair queueing," in *Proceedings of IEEE INFOCOM*, San Francisco, CA, March 1996, pp. 120–128.

[5] M. Shreedhar and G. Varghese, "Efficient fair queuing using deficit round-robin," *IEEE Transactions on Networking*, vol. 4, no. 3, pp. 375–385, June 1996.

[6] S. Kanhere, H. Sethu, and A. Parekh, "Fair and efficient packet scheduling using elastic round robin," *IEEE Transactions on Parallel and Distributed Systems*, vol. 13, no. 3, pp. 324–336, March 2002.

[7] D. Stiliadis and A. Verma, "Latency-rate servers: A general model for analysis of traffic scheduling algorithms," *IEEE Transactions on Networking*, vol. 6, no. 3, pp. 611–624, October 1996.

[8] D. Stiliadis, *Traffic Scheduling in Packet-Switched Networks: Analysis, Design, and Implementation*, Ph.D. thesis, University of California, Santa Cruz, CA, 1996.

[9] S. Kanhere and H. Sethu, "Low-latency guaranteed-rate scheduling using elastic round robin," *Computer Communications*, vol. 25, no. 14, pp. 1315–1322, September 2002.

[10] K. Thompson, G. Miller, and R. Wilder, "Wide-area internet traffic patterns and characteristics," *IEEE Network*, pp. 10–23, November 1997.

[11] I. Widjaja and A. I. Elwalid, "Performance issues in vc-merge capable switches for multiprotocol label switching," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 6, pp. 1178–1189, June 1999.