# Improved Delay Bound and Packet Scale Rate Guarantee for Some Expedited Forwarding Networks

Yuming Jiang
Centre for Quantifiable Quality of Service (Q2S), Norwegian University of Science and Technology

*Abstract*— **Packet Scale Rate Guarantee (PSRG) is a service guarantee defined recently for Expedited Forwarding (EF) service under the Differentiated Services framework. In the original work of PSRG [1], two classes of schedulers are proved to be PSRG servers. In addition, it is proved that end-to-end PSRG and consequently end-to-end delay bound are provided by a network of PSRG servers implementing per-flow scheduling. Moreover, a delay bound is presented for a network of PSRG servers implementing aggregate scheduling. In this paper, we show that these results can be improved for networks of PSRG servers of these schedulers. In particular, we show that the two classes of schedulers belong to a common scheduler family. In addition, we prove that the end-to-end delay bounds can be improved for networks of such schedulers. We also prove that PSRG can be derived from delay bound and call this *PSRG-from-delay-bound* property. Moreover, with this property, we derive and discuss end-to-end PSRG for both the per-flow scheduling network and the aggregate scheduling network.**

## I. Introduction

To provide service guarantees in the Internet, the Differentiated Services (DiffServ) framework [3] has attracted a lot of attention in the networking community due to its scalable and flexible design. In DiffServ, one important service type is Expedited Forwarding (EF). Its corresponding Per-Hop Behavior (PHB), EF PHB, was initially defined in RFC 2598 [7]. However, a recentl work [1] has shown that the schedulers and configuration rates on which the EF definition in RFC 2598 can be implemented are very limited and the defined EF PHB in RFC 2598 is not readily operational because RFC 2598 does not admit quantitative compliance testing. Also pointed out in [1], these limitations cannot be corrected with simple incremental fixes.

Fortunately, there is an alternative definition for EF PHB, called *Packet Scale Rate Guarantee* (PSRG), which not only captures the intuitive content of RFC 2598 but also allows quantitative compliance testing [5] [4]. The new definition has been adopted by IETF as RFC 3246 [5]. Intuitively, PSRG can be viewed as a characterization of how far a node differs from an ideal node. Unlike previous works, such as Packet GPS (PGPS) [12] and Guaranteed Rate (GR) servers [6], where the focus is on investigating how late a node can be with respect to (w.r.t.) GPS, PSRG goes one step further to capture how much a node is late or early w.r.t. GPS. Formally [1]:

*Definition 1:* A server $s$ is said to offer a flow *packet scale rate guarantee* (PSRG) with rate $R_s$ and error term $E_s$, iff the departure time $d_s^i$ of the $i$th packet of the flow satisfies the following condition: For all $i \geq 0$,

$$d_s^i \leq F_s^i + E_s \quad (1)$$

where with $F_s^0 = 0$ and $d_s^0 = 0$, $F_s^i$ is iteratively defined by

$$F_s^i = max[a_s^i, min(d_s^{i-1}, F_s^{i-1})] + \frac{l^i}{R_s}. \quad (2)$$

Here, $F_s^i$ is the target departure time for the $i$th packet of the flow; $a_s^i$ is the arrival time of the packet at the server $s$ and $l^i$ is the length of the packet.

According to the definition, PSRG has two parameters: a rate and an error term. Particularly, the latter captures the error of a server w.r.t. its corresponding fluid server. Hence, a smaller implemented error term means a better packet implementation of the idea fluid server. Since PSRG is characterized by these two parameters, it is a must to derive them for a defined PSRG: either a per-hop PSRG for EF PHB or a per-domain PSRG for EF PDB (Per-Domain Behavior). Furthermore, a smaller derived error term implies a more accurate characterization of it.

In [1], two classes of schedulers have been studied and proved to be PSRG servers. Also, it is proved that end-to-end PSRG and consequently end-to-end delay bound are provided by a per-flow scheduling network in which each node along the path of a flow provides PSRG to it. Furthermore, a delay bound is presented for an aggregate scheduling network in which each node is a PSRG server to the EF aggregate. However, for the aggregate scheduling network, it is not clear from [1] what its end-to-end PSRG is. For ease of exposition, delay in this paper is defined as the queueing delay part ignoring propagation delay.

The purpose of this paper is to derive delay bound and PSRG for networks of PSRG servers that belong to the two scheduler classes. In particular, two representative networks, a per-flow scheduling network and an aggregate scheduling network, are considered. We first show that the two scheduler classes studied in [1] belong to a common scheduler family. We next prove that the end-to-end delay bound can be improved for both the per-flow and the aggregate scheduling networks. We argue that the improvement can be significant. We then prove a relationship between PSRG and delay bound, which we call *PSRG-from-delay-bound* property. This property states that PSRG can be derived from delay bound. Finally, with the property, we derive and discuss end-to-end PSRG for both the per-flow scheduling network and the aggregate scheduling network.

This work is supported in part by ERCIM Fellowship Programme.

## II. Review of Previous Results

### A. PSRG Schedulers

In [1], two classes of schedulers are studied and proved to provide PSRG. One class is the strict priority scheduler. It is proved that the flow at the highest priority level of a strict priority scheduler $s$ is given PSRG with rate $C_s$ and error term $L/C_s$ where $C_s$ is the total output rate of the scheduler and $L$ is the maximum packet length.

Another class of schedulers studied in [1] include a wide family of fair queueing schedulers that are packet-based implementations of the ideal GPS fluid scheduler. All these schedulers have the following property:

$$G_s^i - E_s^1 \leq d_s^i \leq G_s^i + E_s^2, \tag{3}$$

where $G_s^i$ represents the time the packet would departure if the scheduler were the idea GPS fluid scheduler, and $E_s^1$ and $E_s^2$, called error terms, are two constants. (3) captures how much the scheduler can be late or early w.r.t. GPS.

In [1], it is proved that if a scheduler guarantees (3), then it is PSRG server with error term $E_s$ satisfying

$$E_s = E_s^1 + E_s^2. \tag{4}$$

### B. PSRG for a Per-Flow Scheduling Network

Also in [1], it is proved that a per-flow scheduling network provides PSRG. Specifically, if along the path of a flow, each node $s(= 1, \ldots, H)$ provides PSRG to the flow with rate $R_s$ and error term $E_s$. Then, the network provides an end-to-end PSRG to the flow with rate $R(= min_s\{R_s\})$ and error term $E$ determined by

$$E \ = \ \sum_{s=1}^{H} E_s + \sum_{s=1}^{H-1} \frac{L}{R_s}. \tag{5}$$

Except for the strict priority scheduler, all other PSRG servers studied in [1] belong to the class of schedulers satisfying (3). Hence, applying (4) to (5), we further get

$$E \ = \ \sum_{s=1}^{H} E_s^1 + \sum_{s=1}^{H} E_s^2 + \sum_{s=1}^{H-1} \frac{L}{R_s}, \tag{6}$$

where if a scheduler $s$ is strict priority, we can set $E_s^2 = L/C_s$ and $E_s^1 = 0$ as will be discussed in Sec. III.

### C. Delay Bounds

PSRG implies rate guarantee [1] [10]. Specifically, if a scheduler provides a flow PSRG with rate $R_s$ and error term $E_s$, then it is Guaranteed Rate (GR) server to the flow with the same rate $R_s$ and error term $E_s$. Consequently, based on results for GR [6], it can be shown that for the per-flow scheduling network, if a flow is token bucket $(r^f, \sigma^f)$ constrained with rate $r^f$ and burstiness parameter $\sigma^f$ before entering the network and $r^f \leq R(= min_s\{R_s\})$, then for any packet of the flow, its end-to-end delay is bounded by

$$D = \frac{\sigma^f}{R} + \sum_{s=1}^{H} E_s^1 + \sum_{s=1}^{H} E_s^2 + \sum_{s=1}^{H-1} \frac{L}{R_s}. \tag{7}$$

For an aggregate scheduling network of arbitrary topology where each node provides PSGR to the EF aggregate that is formed with FIFO, an end-to-end delay bound has been proved [8] [1]. The assumptions are 1) each EF flow is token bucket constrained at the ingress with parameters $(r^f, \sigma^f)$; 2) each node $s$ provides to the EF aggregate PSRG with rate $R_s$ and error term $E_s$; 3) the amount of EF traffic on any link does not exceed a certain ratio $\alpha < 1$ of the configured rate $R_s$, i.e. $\sum_{f \in \mathcal{F}_s} r^f \leq \alpha R_s$ where $\mathcal{F}_s$ denotes the set of EF flows on the link; 4) for any link, let $\beta_s = \sum_{f \in \mathcal{F}_s} \sigma^f / R_s$ and $\beta$ be a bound on all $\beta_s$; 5) the route of any flow in the network traverses at most $H$ nodes. With these assumptions, if $\alpha < \frac{1}{H-1}$, the bound is

$$D = \frac{H}{1-(H-1)\alpha}(E_{s,max} + \beta), \tag{8}$$

with

$$E_{s,max} = max_s\{E_s^1 + E_s^2 + \frac{L}{R_s}\}. \tag{9}$$

### D. Remarks

Based on (7) and (8), it is clear that both $E_s^1$ and $E_s^2$ affect the obtained delay bounds. This could lead to a belief that to provide a small delay bound, all nodes should have both small $E_s^1$ and small $E_s^2$. However, in the following, we shall prove that these bounds can be improved w.r.t. (7) and (8) if each node belongs to the two scheduler classes in [1]. Based on the improved results, we argue that the belief is not necessarily true. Another point worth highlighting is that while the delay bound has been given as (8) in [1] for the aggregate scheduling network, it is not clear what its end-to-end PSRG is. On the other hand, from the end-user's point of view, a service guarantee makes more sense end-to-end than per-hop. In this sense, it is desirable to study the end-to-end PSRG guaranteed by the aggregate scheduling network. Since in DiffServ networks aggregate scheduling is adopted, this study becomes critical, even though the end-to-end PSRG has been studied for the per-flow scheduling network as shown by (6) [1] [9].

## III. Improved Delay Bounds

We now show that the delay bounds can be improved.

### A. Reference Fluid Server Model

We begin with introducing a server model. In Sec. II, we considered a family of fair queueing schedulers that approximate GPS. Based on the same idea, we extend (3) to characterize the approximation of a packet scheduler w.r.t. its reference fluid scheduler. In particular, we say a packet scheduler $s$ approximates its reference fluid scheduler with error terms $E_s^1$ and $E_s^2$, iff

$$\hat{G}_s^i - E_s^1 \leq d_s^i \leq \hat{G}_s^i + E_s^2, \quad (10)$$

where $\hat{G}_s^i$ is the time packet $i$ would departure if the scheduler were its reference fluid scheduler.

Comparing (10) with (3), it is clear that the class of fair queueing schedulers belong to the scheduler class characterized by (10). The following theorem shows a similar result for strict priority (SP) scheduler.

*Theorem 1:* For the flow at the highest priority level, a packet SP scheduler $s$ approximates its reference fluid SP scheduler with $E_s^1 = 0$ and $E_s^2 = \frac{L}{C_s}$.

*Proof:* The proof includes two parts. i) When any packet at the highest priority level, denoted as the tagged packet, reaches the head of queue, it would be served with rate $C_s$ immediately if the scheduler were the reference fluid SP. However, the tagged packet could be delayed due to a packet in service in the packet SP. Hence, the departure time of the tagged packet from the packet SP is never earlier than it would depart from the fluid SP. So, we have $E_s^1 = 0$. ii) For the tagged packet, the maximum time it may wait after reaching the head of queue is $\frac{L}{C_s}$, which is caused by the packet in service. After the packet in service is serviced, the packet SP scheduler will keep on serving packets at the highest priority level starting from the head-of-queue tagged packet till the highest priority queue is empty. Hence, all these packets depart from the packet SP scheduler with a maximum $\frac{L}{C_s}$ delay w.r.t. the fluid SP scheduler. In other words, $E_s^2 = \frac{L}{C_s}$. ■

With (10), we have the following result:

*Theorem 2:* If a scheduler satisfies (10) and its corresponding fluid scheduler guarantees a rate $R_s$ to the flow, then the scheduler provides PSRG to the flow with rate $R_s$ and error term $E_s^1 + E_s^2$.

*Proof:* Note that in the corresponding fluid scheduler, $R_s$ is the guaranteed rate to the flow. In other words, any packet $i(\geq 1)$ of the flow receives a service rate not less than $R_s$. Hence, the departure time of packet $i$ from the fluid scheduler satisfies: with $\hat{G}_s^0 = 0$,

$$\hat{G}_s^i \leq max\{a_s^i, \hat{G}_s^{i-1}\} + \frac{l^i}{R_s}. \quad (11)$$

We now prove by induction the following result:

$$\hat{G}_s^i \leq F_s^i + E_s^1, \quad (12)$$

with which and (10), $d_s^i \leq \hat{G}_s^i + E_s^2 \leq F_s^i + E_s^1 + E_s^2$ and consequently the theorem follows.

For the base step $i = 1$, since $\hat{G}_s^1 \leq a_s^1 + \frac{l^i}{R_s}$ while $F_s^1 = a_s^1 + \frac{l^i}{R_s}$, (11) holds. For the induction step, assume (12) holds for all packets $1, \ldots, i-1$. Then, for packet $i$, there are two cases.

Case 1: $a_s^i \geq \hat{G}_s^{i-1}$. For this case, we have $\hat{G}_s^i \leq a_s^i + \frac{l^i}{R_s}$ from (11). Since it is easy to verify by definition that $F_s^i \geq a_s^i + \frac{l^i}{R_s}$, (12) holds for this case.

Case 2: $a_s^i < \hat{G}_s^{i-1}$. For this case, we have $\hat{G}_s^i \leq \hat{G}_s^{i-1} + \frac{l^i}{R_s}$ from (11). Based on the induction assumption, $\hat{G}_s^{i-1} \leq F_s^{i-1} + E_s^1$ and hence $\hat{G}_s^i \leq F_s^{i-1} + E_s^1 + \frac{l^i}{R_s}$. In addition, based on (10), $\hat{G}_s^{i-1} \leq d_s^{i-1} + E_s^1$ and hence $\hat{G}_s^i \leq d_s^{i-1} + E_s^1 + \frac{l^i}{R_s}$. Consequently, $\hat{G}_s^i \leq min\{d_s^{i-1}, F_s^{i-1}\} + E_s^1 + \frac{l^i}{R_s} \leq F_s^i + E_s^1$, and (12) also holds for the second case. ■

Similarly, we can prove that (6) and (9) are valid for the per-flow scheduling network and the aggregate scheduling network even though a scheduler adopts SP.

The following is the basis for deriving improved bounds.

*Theorem 3:* If for a flow, a scheduler satisfies (10) and its corresponding fluid scheduler guarantees a rate $R_s$ to the flow, then the scheduler is a Guaranteed Rate (GR) server [6] to the flow with rate $R_s$ and error term $E_s^2$.

*Proof:* By definition, a scheduler $s$ is said to be a Guaranteed Rate server to a flow with rate $R_s$ and error term $\gamma$, iff it guarantees that any packet $j$ of the flow is transmitted by time $GRC_s^j + \gamma$ [6], or $d_s^j \leq GRC_s^j + \gamma$, where with $GRC_s^0 = 0$, $GRC_s^j$ is iteratively defined to be

$$GRC_s^j = max\{a_s^j, GRC_s^{j-1}\} + \frac{l^j}{R_s}. \quad (13)$$

Note that in the reference fluid system, the flow receives service at a rate not less than $R_s$, and hence for any $j \geq 1$,

$$\hat{G}_s^j \leq max\{a_s^j, \hat{G}_s^{j-1}\} + \frac{l^j}{R_s}. \quad (14)$$

Comparing (13) and (14), it is easy to verify that $\hat{G}_s^j \leq GRC_s^j$. Hence from (10), $d_s^j \leq \hat{G}_s^j + E_s^2 \leq GRC_s^j + E_s^2$. ■

### *B. Improved Delay Bounds*

Having proved The. 3, we can re-apply available results and get improved end-to-end delay bounds.

*Theorem 4:* For the per-flow scheduling network, if the flow is token bucket $(r^f, \sigma^f)$-constrained before entering the network, then, if $r^f \leq R (= min_s\{R_s\})$, the end-to-end delay bound becomes [6]

$$D = \frac{\sigma^f}{R} + \sum_{s=1}^{H} E_s^2 + \sum_{s=1}^{H-1} \frac{L}{R_s}. \quad (15)$$

*Theorem 5:* For the aggregate scheduling network, if $\alpha < \frac{1}{H-1}$, the end-to-end delay bound is [8]

$$D = \frac{H}{1-(H-1)\alpha}(E_{s,max}^2 + \beta), \quad (16)$$

where

$$E_{s,max}^2 = max_s\{E_s^2\}. \quad (17)$$

Comparing (15) with (7) and (16) with (8), it is clear that (15) and (16) are improved by removing all the $E_s^1$ items from (7) and (8) respectively. Hence, (15) is smaller and tighter than (7) and so is (16) than (8).

Note that for a scheduler satisfying (10), its $E_s^1$ term can be much larger than its $E_s^2$ term. For example, WF$^2$Q is known to have both the minimum $E_s^1(=\frac{L}{R_s})$ term and the minimum $E_s^2(=\frac{L}{C_s})$ term among various fair queueing schedulers [2]. Suppose there are $N$ flows equally sharing $C_s$. Then, $R_s=\frac{C_s}{N}$. Clearly, for this case $E_s^1=N\cdot E_s^2$ and if $N$ is large, then $E_s^1$ can be much larger than $E_s^2$.

For the network cases, if each scheduler is WF$^2$Q , then (15) is smaller than (7) by $\sum_{s=1}^{H}E_s^1$, and (16) is smaller than (8) by $\frac{H}{1-(H-1)\alpha}max_s\{E_s^1\}$ (Here, from (9) to (17), there is an additional improvement of $\frac{L}{R_s}$. Detailed discussion can be found from Lemma 4 in [8].). Hence, (15) can be much smaller than (7) and so is (16) than (8).

In addition, since (15) and (16) are independent of $E_s^1$, it is reasonable to choose schedulers with small $E_s^2$ to provide tight delay bounds. For example, WFQ, which is simpler to implement than WF$^2$Q and has the same $E_s^2$ value, may be used instead of WF$^2$Q to obtain the same delay bound. For this, however, one may argue that it is PSRG while not delay bound that EF aims to provide. Hence, based on (5) and equivalently (6), to have smaller error term for an implemented PSRG, WF$^2$Q is preferable than WFQ since the former has a much smaller $E_s^1$ value than the latter. While this argument sounds convincing, the "PSRG-from-delay-bound" property to be presented in Sec. IV shows that the PSRG provided by a network of WFQ schedulers can be as good as or even better than the PSRG derived from (6) for the corresponding WF$^2$Q network.

## IV. End-to-End Packet Scale Rate Guarantee

In this section, we first introduce the definition of per-domain PSRG, then develop a technique to study end-to-end PSRG, and finally apply it to both the per-flow scheduling network and aggregate scheduling network.

### A. Definition of Per-Domain PSRG

While Definition 1 defines PSRG for the single node case, it cannot be used to describe PSRG for the network case, for which we need the definition of per-domain PSRG.

Consider the path of a flow $f$ crossing a network domain, which is a tandem system of servers numbered $1,\ldots,H$. Along the path, the flow may aggregate with other flows. The per-domain PSRG is defined as follows [9]:

*Definition 2:* A network domain is said to provide per-domain PSRG to a flow with rate $R$ and error term $E$ iff

$$d_H^i \le \hat{F}^i + E \tag{18}$$

where $\hat{F}^i$ is iteratively defined by

$$\hat{F}^0 \;=\; 0, d_H^0 = 0$$

$$\hat{F}^i \;=\; max[a_1^i, min(d_H^{i-1}, \hat{F}^{i-1})] + \frac{l^i}{R}. \tag{19}$$

Comparing Definition 1 and Definition 2, we can view the latter as a generalization of the former. Particularly, if we view the global system of the end-to-end path of the flow as a black box, (19) is indeed the PSRG virtual finish time function for the end-to-end system with $H$ servers, where $a_1^i$ is the arrival time of packet $i$ to the black box and $d_H^i$ is the departure time of the packet leaving the black box.

### B. PSRG from Delay Bound

The following theorem presents a relationship between PSRG and delay bound, which we call *PSRG-from-delay-bound* property.

*Theorem 6:* For a network of arbitrary topology, if the network provides a bounded delay $D^h$ to an end-to-end flow till hop $h$ with $0\le h\le H$ and $D^0=0$, and from hop $h+1$ to $H$, each node guarantees (10) to the flow, then the network provides to the flow per-domain PSRG with rate $R=min\{R_{h+1},\ldots,R_H\}$ and error term

$$E = D^h + \sum_{s=h+1}^{H}(E_s^1+E_s^2) + \sum_{s=max\{h,1\}}^{H-1}\frac{L}{R_s}, \tag{20}$$

where by convention, $\sum_{s=h+1}^{H}x=0$ for all $h\ge H$.

*Proof:* The proof includes three parts. First, from The. 2, it is known that each node $h+1$ to $H$ is PSRG server to the flow. Hence, based on the concatenation property of PSRG servers [1] [9], we can treat the concatenated system of node $h+1$ to node $H$ as a single PSRG server with rate $R=min\{R_{h+1},\ldots,R_H\}$ and error term $\sum_{s=h+1}^{H}(E_s^1+E_s^2)+\sum_{s=h+1}^{H-1}\frac{L}{R_s}$, whose PSRG virtual finish time function is defined by (19) with a replacement of $a_1^i$ with $a_{h+1}^i$. This replacement is because the arrival time of packet $i$ to the concatenated system is $a_{h+1}^i$.

Second, consider the sub-system comprised of node 1 to $h$ where the flow passes through. According to Def. 2, we define domain PSRG virtual finish time function for it as

$$\hat{F}_h^i = max[a_1^i, min(d_h^{i-1}, \hat{F}_h^{i-1})] + \frac{l^i}{R}.$$

Clearly, $\hat{F}_h^i \ge a_1^i + \frac{l^i}{R} \ge a_1^i$. In addition, the sub-system guarantees delay bound $D^h$ to the flow. Hence,

$$d_h^i \;\le\; a_1^i + D^h \le \hat{F}_h^i + D^h.$$

Then, by definition, the sub-system provides PSRG to the flow and hence can be treated as a single PSRG server.

Finally, based on the concatenation property [1] [9] and per-domain PSRG definition, the theorem is proved. ■

Before proceeding, let us re-look at The. 6, in which, $h$ can be 0 to $H$. When $h=0$, The. 6 produces the same result as (6). When $h$ changes from $1,\ldots,H$, The. 6 may generate different results. While all these results are valid, one may choose $h$ such that the generated result is the best in terms of minimum error term obtained.

While it is usual to derive delay bound from PSRG, *the PSRG-from-delay-bound makes sense* because of the following reasons. First, deriving delay bound is technically mature. A lot of techniques, such as network calculus, can be used. However, very few techniques can be used to determine PSRG, particularly end-to-end PSRG.

Second, although (6) can be used to calculate the error term for end-to-end PSRG, it is applicable only to per-flow scheduling networks. For aggregate scheduling networks of arbitrary topology, of which DiffServ is an example, to the best of our knowledge, no technique has been developed. On the other hand, it is desirable to extend EF from per-hop to per-domain, since from an end-user's point of view, end-to-end service makes more sense. In fact, this is an ongoing effort of DiffServ [11]. In addition, like EF PHB, the per-domain EF service should allow quantitative compliance testing. The. 6 provides a basis for such analysis.

Third, the error term determined from The. 6 can be as good as that determined from other techniques. For example, if $\sigma^f$ is not too large, the error term determined from The. 6 for the per-flow scheduling network may be smaller than that from (6).

### C. PSRG for the Per-Flow Scheduling Network

Note that Theorem 6 is very general, which is also applicable to the per-flow scheduling network. For such a network, letting $h = 0$ in (20), we get the following result:

*Theorem 7:* For the per-flow scheduling network, if the flow is token bucket $(r^f, \sigma^f)$-constrained at the ingress and $r^f \leq R(= min_s\{R_s\})$, then the network provides an end-to-end PSRG to the flow with rate $R = min_s\{R_s\}$ and error term $E$ determined by

$$E = \frac{\sigma^f}{R} + \sum_{s=1}^{H} E_s^2 + \sum_{s=1}^{H-1} \frac{L}{R_s}. \tag{21}$$

To compare (21) with (6), let us consider an example, in which each node is again assumed to implement $\mathrm{WF^2Q}$ that has $E_s^1 = \frac{L}{R_s}$. Also assume each node allocates the same rate $R$ to the flow. Then the difference between the two error terms from (21) and (6) is

$$E^{(21)} - E^{(6)} = \frac{\sigma^f - H \cdot L}{R}. \tag{22}$$

From (22), we can conclude that the error term from (6) is smaller only if $\sigma^f > H \cdot L$. If, however, $\sigma^f \leq H \cdot L$, the error term determined from (21) is better. Note that many other well-known fair queueing schedulers such as WFQ have the same $E_s^2$ value as $\mathrm{WF^2Q}$ but have much larger $E_s^1$ value than $\frac{L}{R_s}$ [2]. In consequence, $E^{(21)} - E^{(6)} \leq 0$ or $E^{(21)} \leq E^{(6)}$ is highly possible if $H$ is also large. Hence, $E^{(21)}$ provides another option for calculating end-to-end PSRG for the per-flow scheduling network.

### D. PSRG for the Aggregate Scheduling Network

We finally present the end-to-end PSRG for the aggregate scheduling network based on Theorems 5 and 6:

*Theorem 8:* For the aggregate scheduling network, if $\alpha < \frac{1}{H-1}$, it provides to any flow $f$ end-to-end PSRG with rate $r^f$ and error term $E$ as

$$E = \frac{H}{1-(H-1)\alpha}(E_{s,max}^2 + \beta), \tag{23}$$

where $E_{s,max}^2 = max_s\{E_s^2\}$.

## V. Conclusion

This paper has made two contributions in the context of Expedited Forwarding. First, improved delay bounds have been derived for two representative networks: a per-flow scheduling network and an aggregate scheduling network of arbitrary topology, in which, each scheduler is characterized by two error terms w.r.t. its corresponding fluid scheduler. Second, the *PSRG-from-delay-bound* property has been proved. With this property, end-to-end PSRG has been derived for the two representative networks. A simple comparison between the end-to-end PSRG determined from the PSRG-from-delay-bound property and that from the original PSRG work shows that PSRG-from-delay-bound provides another option for determining end-to-end PSRG for per-flow scheduling networks. The application of PSRG-from-delay-bound property to the aggregate scheduling network has allowed to derive end-to-end PSRG for such networks.

## References


[1] J. C. R. Bennett and et al. Delay jitter bounds and packet scale rate guarantee for Expedited Forwarding. *IEEE/ACM Trans. Networking*, 10(4):529 – 540, August 2002.

[2] J. C. R. Bennett and H. Zhang. $\mathrm{WF^2Q}$: Worst-case fair weighted fair queueing. In *Proc. IEEE INFOCOM'96*, 1996.

[3] S. Blake and et al. An architecture for Differentiated Services. *IETF RFC 2475*, Dec. 1998.

[4] A. Charny and et al. Supplemental information for the new definition of the EF PHB. *IETF RFC 3247*, March 2002.

[5] B. Davie and et al. An Expedited Forwarding PHB. *IETF RFC 3246*, March 2002.

[6] P. Goyal, S. S. Lam, and H. M. Vin. Determining end-to-end delay bounds in heterogeneous networks. *Springer Multimedia Systems*, 5:157–163, 1997.

[7] V. Jacobson, K. Nichols, and K. Poduri. An Expedited Forwarding PHB. *IETF RFC 2598*, June 1999.

[8] Y. Jiang. Delay bounds for a network of Guaranteed Rate servers with FIFO aggregation. *Computer Networks*, 40(6):683–694, Dec. 2002.

[9] Y. Jiang. Per-domain packet scale rate guarantee for Expedited Forwarding. In *Proc. International Workshop on Quality of Service (IWQoS)*, LNCS 2707, pages 422–439, 2003.

[10] Y. Jiang. Relationship between guaranteed rate server and latency rate server. *Computer Networks*, 43(3):307 – 315, 2003.

[11] K. Nichols and B. Carpenter. Definition of differentiated services per domain behaviors and rules for their specification. *IETF RFC 3086*, April 2001.

[12] A. K. Parekh and R. G. Gallager. A generalized processor sharing approach to flow control in intergrated services networks: The single-node case. *IEEE/ACM Trans. Networking*, 1(3):344–357, June 1993.