

A Novel Modified Secant Method for Computing the Fair Share Rate

Dong Wei and Nirwan Ansari

Advanced Networking Laboratory, Department of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark, USA

In this paper, we propose a novel modified secant method to compute the flow fair share rate within the framework of the core-stateless fair queueing [1]. The geometric explanation and numerical results demonstrate that the proposed method possesses better performance in terms of accuracy and convergence than that proposed in [1].

Keywords: core-stateless, fair queueing, fair share rate.

1. Introduction

Most scheduling schemes for quality of services (*QoS*) provisioning require all routers to manage per flow states and perform per flow operations [2] [3]. Such a stateful architecture is less scalable and less robust than the stateless architectures, such as the original IP and the DiffServ framework [4].

In [1], a framework of core-stateless fair queueing (*CSFQ*), which does not require per flow management or operation at core routers, was proposed. It is able to provide high quality of services while maintaining a high level of robustness, simplicity and scalability. In this framework, one key element is the calculation of the fair share rate, which is briefly reviewed in the next section. The accuracy and effectiveness of the scheme for computing the fair share rate affects the effectiveness of the core-stateless fair queueing scheme. A more effective method for computing the fair share rate is thus proposed in this paper.

The rest of the paper is organized as follows. In Section 2, we review the core-stateless fair queueing and the original method for computing the fair share rate. We present the proposed

method in Section 3, and compare performance between the proposed method and the original one by geometric reasoning in Section 4. Simulation results are shown in Section 5. In Section 6, we discuss the applications of the proposed method in other scenarios. Finally, concluding remarks are presented in Section 7.

2. Background

2.1. The Core-stateless Fair Queueing

In a contiguous region of a network, routers are classified as edge routers and core routers. Edge routers maintain per flow state and estimate the incoming rate of each flow; based on this estimation, each packet is labeled when it departs from the edge router. Core routers do not maintain per flow state; they just use a *FIFO* queue and a probabilistic dropping algorithm that uses the packet labels and an estimate of the aggregate traffic at the router. Since no per flow state is maintained in the core routers and packets are served based on “first come first served” policy, the implementation complexity of the core routers is $O(1)$.

Denote x_i as the arrival rate of flow i and f as the fair share rate, respectively, whereas $x_i \geq 0$ and $f \geq 0$. Then, each incoming packet of flow i is dropped with probability p_i , where $p_i = \max\left(0, 1 - \frac{f}{x_i}\right)$. Denote C as the throughput of a single link, L as its link capacity, and N as the number of flows on this link, respectively.

Thus, the throughput of this link can be written as:

$$C = C(f) = \sum_{i=1}^N \min(x_i, f). \quad (1)$$

Note that C is a continuous non-decreasing concave and piecewise-linear function of f . If the link is congested, i.e., $\sum_{i=1}^N x_i > L$, there is a unique solution of f to $\sum_{i=1}^N \min(x_i, f) = L$; if the link is not congested, i.e., $\sum_{i=1}^N x_i \leq L$, we define $f = \max_{i=1,2,\dots,N} (x_i)$. Clearly, if all x_i 's are known, f can be computed. However, this requires maintenance of per flow information. To avoid maintaining per flow state, we can use the measurement of C to compute f .

2.2. The Computation of the Fair Share Rate

If per flow state information is not maintained, all x_i 's are unknown. In this case, f can be estimated based on both the measurement and the properties of the function of C . The estimation of f can be formulated as a root finding problem of a nonlinear equation as follows:

$$\psi(f) = \sum_{i=1}^N \min(x_i, f) - L = 0. \quad (2)$$

Denote the exact fair share rate by f^* , i.e., $C(f^*) = L$. The effectiveness of the **CSFQ** scheme depends on the effectiveness of the method for computing f . Denote the k^{th} iteration of the f estimation and the measured throughput by $f^{(k)}$ and $C^{(k)}$, respectively. The original fair share rate estimation method proposed in [1], which is referred to as the **FSR_{CSFQ}** method in this paper, is written as:

$$f^{(k+1)} = f^{(k)} \cdot \frac{L}{C^{(k)}}.$$

It can be rewritten as:

$$f^{(k+1)} = f^{(k)} - \frac{f^{(k)}}{C^{(k)}}(C^{(k)} - L). \quad (3)$$

3. The Modified Secant Method

3.1. The Regula Falsi Method and the Secant Method

For the nonlinear equation Eq. (2), we know the root must be in the interval $[0, L]$. The regula falsi method [5], a closed domain method, can be expressed as follows:

$$f^{(k+1)} = \begin{cases} f^{(k)} - \frac{f^{(k)} - f^{(k-1)}}{C^{(k)} - C^{(k-1)}} \cdot (C^{(k)} - L), & \text{when } (C^{(k)} - L) \cdot (C^{(k-1)} - L) < 0 \\ f^{(k)} - \frac{f^{(k)} - f^{(k-2)}}{C^{(k)} - C^{(k-2)}} \cdot (C^{(k)} - L), & \text{otherwise} \end{cases} \quad (4)$$

whereas $f^{(0)} = 0$, $f^{(1)} = L$.

The secant method is a numerical method for solving a nonlinear equation [5], such as Eq. (2), when the derivative $\frac{\partial C^{(k)}}{\partial f}$ is unavailable.

It can be written as:

$$f^{(k+1)} = f^{(k)} - \frac{f^{(k)} - f^{(k-1)}}{C^{(k)} - C^{(k-1)}} \cdot (C^{(k)} - L) \quad (5)$$

A secant to a curve is a straight line which passes through two points on the curve. Two initial approximations, $f^{(0)}$ and $f^{(1)}$, are required. It is claimed that the secant method converges much faster than the regula falsi method [5].

3.2. The Modified Secant Method

We modify the secant method for the following two reasons: 1) f may change when the traffic pattern changes. For example, after a period of stable status, i.e., when $f^{(k)} = f^{(k-1)}$, if there is a change of the throughput, i.e., the arrival pattern changes, we need to adjust the estimation of f . In this case, since $f^{(k)} - f^{(k-1)} = 0$, Eq. (5) cannot be used; however, we know that the origin must be on the curve of Eq. (1). Therefore, in this case, we use $f^{(k-1)} = 0$, $C^{(k-1)} = 0$ to compute $f^{(k+1)}$; 2) as shown in the next section, when f decreases, by Eq. (5), $f^{(k+1)}$ may be less than 0, which cannot be used. Therefore, we let $f^{(k-1)} = 0$, in this case. The proposed

scheme can be written as a pseudo code as in Fig. 1.

Initialization Procedure

$f^{(0)} \leftarrow 0;$
 $C^{(0)} \leftarrow 0;$
 $f^{(1)} \leftarrow \delta \cdot L;$
 $C^{(1)} \leftarrow \text{Measure_Throughput};$

Update Fair Share Rate Procedure

$k \leftarrow k + 1;$
 $C^{(k)} \leftarrow \text{Measure_Throughput};$
if $(C^{(k)} - L) \cdot (C^{(k-1)} - L) < 0$
 $f^{(k-1)} \leftarrow 0;$
 $C^{(k-1)} \leftarrow 0;$
 $f^{(k+1)} \leftarrow f^{(k)} - \frac{f^{(k)} - f^{(k-1)}}{C^{(k)} - C^{(k-1)}} \cdot (C^{(k)} - L)$
if $f^{(k+1)} < 0$
 $f^{(k+1)} \leftarrow f^{(k)} \cdot \frac{L}{C^{(k)}};$

Fig. 1. The modified secant method.

When the scheduler starts to transmit packets, to avoid buffer overflow, we first attempt a fair share rate $f^{(1)} = \delta \cdot L$, which is small enough that the throughput is not greater than L . With the measurement of $C^{(1)}$, the modified secant method is initiated. The update fair share rate procedure is called upon a packet arrival. The explanation of this procedure is presented in the next section.

4. Performance Analysis

In this section, we compare the performance in terms of convergence between the method proposed in [1] and our modified secant method. We demonstrate that, with our proposed secant method, f^* can be reached whereas, with the FSR_{CSFQ} method, f^* can only be approximated,

but may not be reached. We consider two scenarios: 1) when f increases, and 2) when f decreases.

Since the throughput C is a continuous, non-decreasing, concave, and piecewise-linear function of f , the following inequalities must hold, $\forall 0 < f^{(a)} < f^{(b)} < L$.

$$C(f^{(a)}) \leq C(f^{(b)}). \quad (6)$$

$$\frac{\partial C(f^{(b)})}{\partial f} \leq \frac{\partial C(f^{(a)})}{\partial f}. \quad (7)$$

$$\frac{C(f^{(b)})}{f^{(b)}} \leq \frac{C(f^{(a)})}{f^{(a)}}. \quad (8)$$

$$\frac{\partial C(f^{(b)})}{\partial f} \leq \frac{C(f^{(b)})}{f^{(b)}}. \quad (9)$$

Ineq. (6) demonstrates the non-decreasing property, and Ineqs. (7), (8) and (9) demonstrate the concave property. By the above inequalities and the fact that the origin is on the curve of $C(f)$, Ineq. (10) can be readily derived.

$$\frac{\partial C(f^{(b)})}{\partial f} \leq \frac{C(f^{(b)}) - C(f^{(a)})}{f^{(b)} - f^{(a)}} \leq \frac{\partial C(f^{(a)})}{\partial f} \quad (10)$$

4.1. Case 1: The Fair Share Rate Increases

Denote the estimation of f of the k^{th} iteration, by using the FSR_{CSFQ} method and by using the proposed method, as $f_{CSFQ}^{(k)}$ and $f_{Secant}^{(k)}$, respectively. Let $0 < f_{Secant}^{(k-1)} < f_{Secant}^{(k)} < f^*$, and $f_{CSFQ}^{(k)} = f_{Secant}^{(k)}$.

$$\text{By Ineq. (8), we have } \frac{C(f_{Secant}^{(k)})}{f_{Secant}^{(k)}} \leq \frac{C(f_{Secant}^{(k-1)})}{f_{Secant}^{(k-1)}}.$$

$$\text{Then, } C(f_{Secant}^{(k)}) \cdot f_{Secant}^{(k-1)} \leq C(f_{Secant}^{(k-1)}) \cdot f_{Secant}^{(k)}.$$

$$\text{Thus, } C(f_{Secant}^{(k)}) \cdot f_{Secant}^{(k)} - C(f_{Secant}^{(k)}) \cdot f_{Secant}^{(k-1)} \geq C(f_{Secant}^{(k)}) \cdot f_{Secant}^{(k)} - C(f_{Secant}^{(k-1)}) \cdot f_{Secant}^{(k)}.$$

$$\text{Therefore, } \frac{C(f_{Secant}^{(k)})}{f_{Secant}^{(k)}} \geq \frac{C(f_{Secant}^{(k)}) - C(f_{Secant}^{(k-1)})}{f_{Secant}^{(k)} - f_{Secant}^{(k-1)}}.$$

$$\therefore f_{Secant}^{(k)} \leq f^*, \text{ by Ineq. (6), } C(f_{Secant}^{(k)}) \leq L,$$

$$\begin{aligned} \therefore f_{Secant}^{(k)} - \frac{f_{Secant}^{(k)}}{C(f_{Secant}^{(k)})} \cdot (C(f_{Secant}^{(k)}) - L) \\ \leq f_{Secant}^{(k)} - \frac{f_{Secant}^{(k)} - f_{Secant}^{(k-1)}}{C(f_{Secant}^{(k)}) - C(f_{Secant}^{(k-1)})} \\ \cdot (C(f_{Secant}^{(k)}) - L). \end{aligned}$$

$$\therefore f_{CSFQ}^{(k)} = f_{Secant}^{(k)}$$

$$\begin{aligned} \therefore f_{CSFQ}^{(k)} - \frac{f_{CSFQ}^{(k)}}{C(f_{CSFQ}^{(k)})} \cdot (C(f_{CSFQ}^{(k)}) - L) \\ \leq f_{Secant}^{(k)} - \frac{f_{Secant}^{(k)} - f_{Secant}^{(k-1)}}{C(f_{Secant}^{(k)}) - C(f_{Secant}^{(k-1)})} \\ \cdot (C(f_{Secant}^{(k)}) - L). \end{aligned}$$

$$\text{By Eqs. (3) and (5), } f_{CSFQ}^{(k+1)} \leq f_{Secant}^{(k+1)}.$$

$$\text{By Ineq. (10), } \frac{C(f_{Secant}^{(k)}) - C(f_{Secant}^{(k-1)})}{f_{Secant}^{(k)} - f_{Secant}^{(k-1)}} \leq$$

$$\frac{\partial C(f^{(k)})}{\partial f} \leq \frac{C(f^*) - C(f_{Secant}^{(k)})}{f^* - f_{Secant}^{(k)}}.$$

$$\therefore C(f_{Secant}^{(k)}) \leq L,$$

$$\begin{aligned} \therefore f_{Secant}^{(k)} - \frac{f_{Secant}^{(k)} - f_{Secant}^{(k-1)}}{C(f_{Secant}^{(k)}) - C(f_{Secant}^{(k-1)})} \\ \cdot (C(f_{Secant}^{(k)}) - L) \\ \leq f_{Secant}^{(k)} - \frac{f^* - f_{Secant}^{(k)}}{C(f^*) - C(f_{Secant}^{(k)})} \\ \cdot (C(f_{Secant}^{(k)}) - L). \end{aligned}$$

$$\therefore C(f^*) = L,$$

$$\begin{aligned} \therefore f_{Secant}^{(k)} - \frac{f_{Secant}^{(k)} - f_{Secant}^{(k-1)}}{C(f_{Secant}^{(k)}) - C(f_{Secant}^{(k-1)})} \\ \cdot (C(f_{Secant}^{(k)}) - L) \leq f^*. \end{aligned}$$

$$\text{Thus, by Eq. (5), } f_{Secant}^{(k)} < f_{Secant}^{(k+1)} < f^*.$$

Therefore,

$$f_{CSFQ}^{(k+1)} < f_{Secant}^{(k+1)} < f^* \quad (11)$$

Ineq. (11) demonstrates that the secant method can approximate the exact value of the fair share

rate f^* better than the FSR_{CSFQ} method, when the fair share rate increases.

As shown in Fig. 2, the function of C vs f can be depicted as the solid line, which passes through the origin. a , b , c , d , and e are points on the curve with Cartesian coordinates $(f_{Secant}^{(k-1)}, C(f_{Secant}^{(k-1)}))$, $(f_{Secant}^{(k)}, C(f_{Secant}^{(k)}))$, (f^*, L) , $(f_{CSFQ}^{(k+1)}, C(f_{CSFQ}^{(k+1)}))$ and $(f_{Secant}^{(k+1)}, C(f_{Secant}^{(k+1)}))$, respectively. d , b and the origin are on the same line, which is used to compute $(f_{CSFQ}^{(k+1)}, C(f_{CSFQ}^{(k+1)}))$. a , b and e are on the same line, which is used to compute $f_{Secant}^{(k+1)}$. Since $0 < f_{Secant}^{(k-1)} < f_{Secant}^{(k)} < f^*$ and $f_{CSFQ}^{(k)} = f_{Secant}^{(k)}$, the slope of the dotted line \overline{ab} , $\frac{C(f_{Secant}^{(k)}) - C(f_{Secant}^{(k-1)})}{f_{Secant}^{(k)} - f_{Secant}^{(k-1)}}$, must be greater

than the slope of the dotted line \overline{bd} , $\frac{C(f_{CSFQ}^{(k+1)})}{f_{CSFQ}^{(k+1)}}$.

The ideal slope is the slope of the dotted line \overline{bc} , which passes the point with the exact fair share rate, f^* . Therefore, with the slope used by our modified secant method, it approximates point c better than that by the FSR_{CSFQ} method.

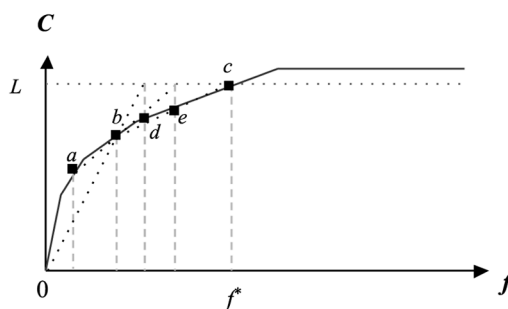


Fig. 2. The geometric explanation of the fair share rate computation when it increases.

Note that, 1) if there is only one line segment between point c and the origin, both methods are able to compute f^* ; 2) if there are more than one line segment between point c and the origin, the FSR_{CSFQ} method may only approximate point c closely, but can never reach it, mathematically. However, the proposed method can still reach point c , if, with enough iterations, line \overline{ab} passes through point c .

4.2. Case 2: The Fair Share Rate Decreases

In this section, we first show that, when f decreases, the FSR_{CSFQ} method is equivalent to the regula falsi method. Since it is claimed, in [5], that the secant method converges much more rapidly than the regula falsi method, our proposed method possesses better performance in terms of convergence than the FSR_{CSFQ} method. Finally, we explain the reason why, in this case, the secant method cannot be directly deployed.

Let $f^* < f_{CSFQ}^{(k)} < f_{CSFQ}^{(k-1)}$. Thus, by Eq. (6), $L < C(f_{CSFQ}^{(k)})$.

Then, by Eq. (3), $0 < f_{CSFQ}^{(k+1)} < f_{CSFQ}^{(k)}$, implying that the computed fair share rate approximates the exact rate from the right hand side, as shown in Fig. 3. a , b , and c are points on the curve with Cartesian coordinates, $(f_{CSFQ}^{(k-1)}, C(f_{CSFQ}^{(k-1)}))$, $(f_{CSFQ}^{(k)}, C(f_{CSFQ}^{(k)}))$ and (f^*, L) , respectively. The FSR_{CSFQ} method uses the ori-

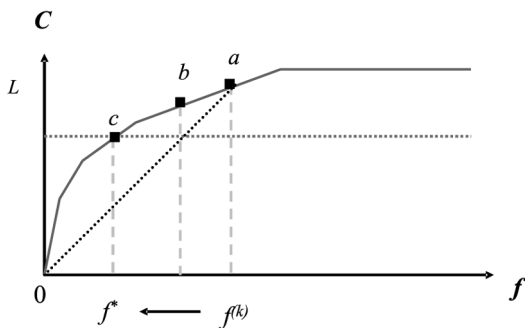


Fig. 3. The geometric explanation of the fair share rate computation, when it decreases, using the FSR_{CSFQ} method.

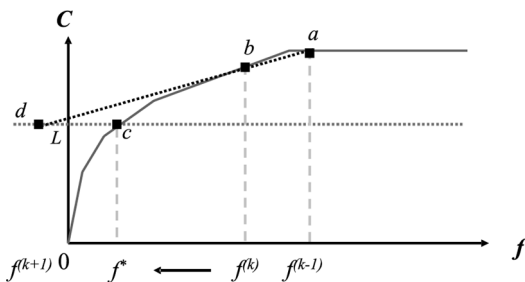


Fig. 4. The geometric explanation of the fair share rate computation, when it decreases, using the modified secant method.

gin and point b to compute $f_{CSFQ}^{(k+1)}$. It is, mathematically, equivalent to finding the root of Eq. (2), by using Eq. (4), whereas $f^{(k-2)} = 0$. Again, if there is only one line segment between point c and the origin, the FSR_{CSFQ} method is able to compute the exact fair share rate, f^* ; otherwise, the FSR_{CSFQ} method may only approximate point c closely, but can never reach it.

When the secant method is deployed, $\frac{C^{(k)} - L}{C^{(k-1)} - L}$ may be less than $\frac{f^{(k)}}{f^{(k-1)}}$, and thus by Eq. (5), $f^{(k+1)} < 0$, as shown in Fig. 4. a , b , c , and d are points on the curve with Cartesian coordinates $(f_{Secant}^{(k-1)}, C(f_{Secant}^{(k-1)}))$, $(f_{Secant}^{(k)}, C(f_{Secant}^{(k)}))$, (f^*, L) and $(f_{Secant}^{(k+1)}, C(f_{Secant}^{(k+1)}))$, respectively. The fair share rate f cannot be set to a value less than 0. Thus, in this sense, the original secant method cannot be used without modification. Therefore, when $f^{(k+1)} < 0$, we compute $f^{(k+1)}$ by Eq. (3). That is, if $f^{(k+1)} < 0$, the FSR_{CSFQ} method is used again in this iteration.

Note that, 1) if there is only one line segment between point a and the origin, in Fig. 3, both methods are able to compute the exact fair share rate, f^* ; 2) if there are more than one line segment between point a and the origin, the FSR_{CSFQ} method may approximate point c closely, but can never reach it, mathematically. However, the proposed method can still reach point c , if, with enough iterations, point b is on the line which passes through point c ; 3) theoretically, using the method proposed in [1], when the fair share rate decreases, the computed fair share rate is always greater than the exact value, *i.e.*, $f^* < f_{CSFQ}^{(k)}$, and thus the expected rate of the accepted traffic is greater than the link capacity, therefore making the expected queue size grow infinitely. Using the proposed modified secant method, the computed fair share rate of each iteration may be greater or less than the exact value before it reaches the exact value, and thus it performs better in terms of controlling the expected queue size.

5. Simulation Results

In the previous section, we demonstrate, geometrically, that the secant method converges faster than the FSR_{CSFQ} method. The following examples show that the proposed scheme converges to f^* faster than the FSR_{CSFQ} method.

5.1. Example 1: The Fair Share Rate Increases

Suppose 20 flows share one link with a capacity of 10. Assume, starting from time 0, the arrival rates of the first five flows are 1, 2, 3, 4, and 5, respectively, and the remaining 15 flows are 1. Assume, before t_{K-2} , the fair share rate reaches the exact rate and remains stable till t_{K-1} , i.e., by Eq. 2, $f^{(K-2)} = f^{(K-1)} = 0.5$. Starting from t_K , only the first five flows remain the same and the remaining 15 flows become idle, and therefore, the function of aggregate rate C vs the fair share rate f can be written as:

$$C = \begin{cases} 5 \cdot f, & \text{when } 0 \leq f < 1 \\ 4 \cdot f + 1, & \text{when } 1 \leq f < 2 \\ 3 \cdot f + 3, & \text{when } 2 \leq f < 3 \\ 2 \cdot f + 6, & \text{when } 3 \leq f < 4 \\ f + 10, & \text{when } 4 \leq f < 5 \\ 15, & \text{when } 5 \leq f \end{cases} \quad (12)$$

whereas the root of the equation

$$\psi(f) = \sum_{i=1}^N \min(x_i, f) - L = 0,$$

is $f^* = 2.333333$. The relative error is defined as:

$$e^{(k)} = \left| \frac{f^{(k)} - f^*}{f^*} \right| \cdot 100\% \quad (13)$$

Denote the relative error by using the FSR_{CSFQ} method and our proposed modified secant method by $e_{CSFQ}^{(k)}$ and $e_{Secant}^{(k)}$, respectively. Note that, in the computation, we only keep four digits after the decimal point. Fig. 5 and 6 show the comparison of the iterative computation of the FSR_{CSFQ} method and the modified secant method. We see that, with 2 iterations, by using the proposed method, f^* can be reached, whereas the relative error of the FSR_{CSFQ} method is still 1.47%.

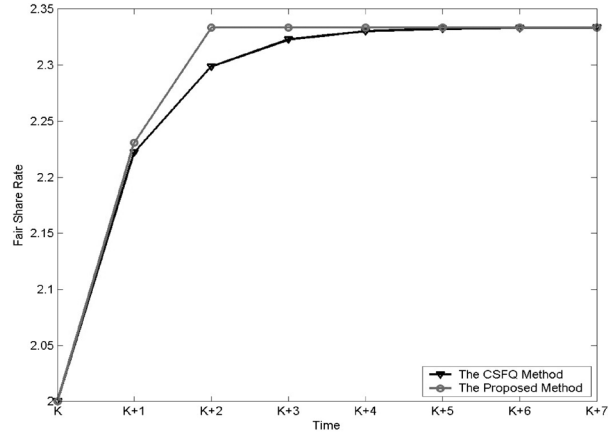


Fig. 5. Computation of the fair share rate using the FSR_{CSFQ} method and the modified secant method in Example 1.

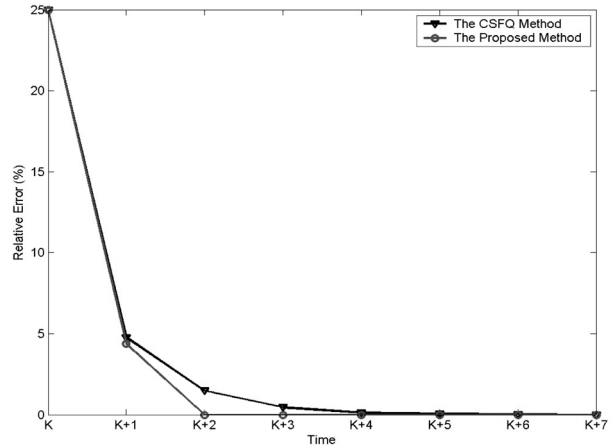


Fig. 6. The computed relative error of the fair share rate using the FSR_{CSFQ} method and the modified secant method in Example 1.

5.2. Example 2: The Fair Share Rate Decreases

Suppose 5 flows share the same link as in Example 1. Assume, starting from time 0, the arrival rate of the first four flows are 1, 2, 3, and 4, respectively, and the last flow is idle. Before t_{K-2} , the fair share rate reaches the exact rate and remains stable till t_{K-1} , i.e., $f^{(K-2)} = f^{(K-1)} = 4$. Starting from t_K , only the last flow becomes active and the arrival rate is 5, while the remaining flows remain the same as before. Then, Eq. (12) still holds, and $f^* = 2.333333$. Fig. 7 and 8 show the comparison of the iterative computation of the

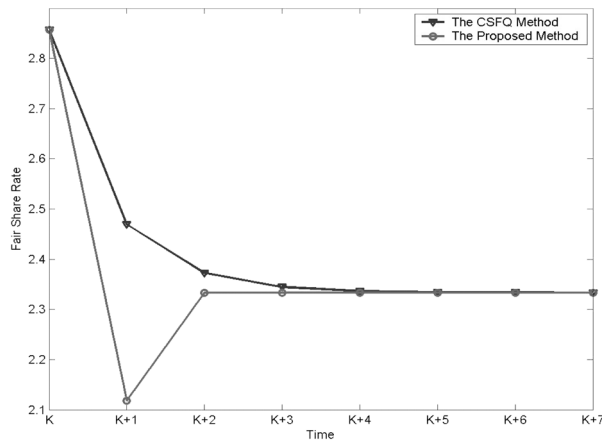


Fig. 7. Computation of the fair share rate using the FSR_{CSFQ} method and the modified secant method in Example 2.

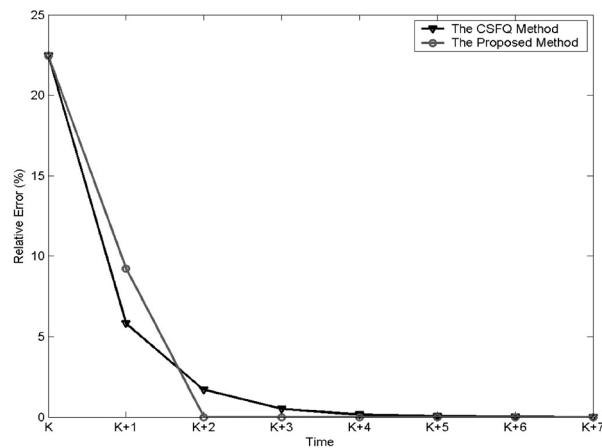


Fig. 8. The computed relative error of the fair share rate using the FSR_{CSFQ} method and the modified secant method in Example 2.

FSR_{CSFQ} method and our proposed modified secant method. We can see that, with 2 iterations, by using the proposed method, f^* can be reached, whereas the relative error of the FSR_{CSFQ} method is 1.68%. This result is very similar to that in Example 1.

With these examples, we can see that the modified secant method converges much more rapidly than the method proposed in [1].

6. Discussion

In this paper, we have proposed a novel modified secant method for computing the fair share

rate on a link shared by multiple flows. The CSFQ scheduling scheme is a min-max scheme per se [1]; the proposed method, which only improves the performance to calculate the fair share rate, does not change this property. In the examples from the previous section, all flows are treated equally, *i.e.*, each flow has the same weight. However, the proposed method can be readily extended to a weighted CSFQ scheduling scheme too, by deploying and calculating nominal fair share rate \bar{f} using Eq. (14) instead of the fair share rate f using Eq. (1),

$$C = C(\bar{f}) = \sum_{i=1}^N (x_i, w_i \cdot L\bar{f}) \quad (14)$$

where w_i is the weight of flow i and $\sum_{i=1}^N w_i = 1$.

In some applications, there are two parts in each flow - guaranteed rate (like voice and video, etc.) and best effort rate (such as data). The guaranteed rate must be satisfied and best effort rate is computed based on the fair share. The proposed method can be extended to these applications by using Eq. (15),

$$C = C(\bar{f}) = \sum_{i=1}^N (x_i, g_i + w_i \cdot L\bar{f}) \quad (15)$$

where g_i is the guaranteed rate of flow i . $\sum_{i=1}^N g_i \leq L$, and $\bar{L} = L - \sum_{i=1}^N g_i$.

Note that, in these two scenarios, C is a still continuous non-decreasing concave and piecewise-linear function of \bar{f} . Therefore, the proposed method can be employed to calculate the nominal fair share rate \bar{f} instead of f .

7. Conclusions

In this paper, we have proposed a novel modified secant method for computing the fair share rate on a link shared by multiple flows. By geometric reasoning, we demonstrate that the proposed method converges to the exact value of the fair share rate, f^* , faster than the FSR_{CSFQ}

method, and can finally reach f^* . The numerical results also show that the proposed method performs better in terms of accuracy and convergence than the original method proposed in [1].

Acknowledgement

This work has been supported in part by the New Jersey Commission on Science and Technology via the NJWINS, and the New Jersey Commission on Higher Education via the NJ I-TOWER project.

References

- [1] STOICA, I., SHENKER, S. AND HUI ZHANG, Core-stateless fair queueing: a scalable architecture to approximate fair bandwidth allocations in high-speed networks, *IEEE/ACM Transactions on Networking*, Volume: 11, Issue: 1, Feb. 2003, pp. 33–46.
- [2] PAREKH, A.K. AND GALLAGER, R.G, A generalized processor sharing approach to flow control in integrated services networks: the single-node case, *IEEE/ACM Transactions on Networking*, Volume: 1, Issue: 3, June 1993, pp. 344–357.
- [3] M. SHREEDHAR AND G. VARGHESE, Efficient fair queueing using deficit round robin, *ACM/IEEE Transactions on Networking*, Volume: 4, Issue: 3, June 1996, pp. 375–385.
- [4] S. BLAKE, D. BLACK, M. CARLSON, E. DAVIES, Z. WANG, AND W. WEISS, An Architecture for Differentiated Services, *IETF RFC 2475*, December 1998.
- [5] J. HOFFMAN, *Numerical Methods for Engineers and Scientists, Second Edition – Revised and Expanded*, Marcel Dekker Inc., 2001.

Received: January, 2005
Accepted: May, 2005

Contact address:

Dong Wei and Nirwan Ansari
Advanced Networking Laboratory,
Department of Electrical and Computer Engineering
New Jersey Institute of Technology
Newark, NJ 07102, USA
e-mail: Nirwan.Ansari@njit.edu

DONG WEI graduated from Tsinghua University with a BSEE degree. He received the MSEE and Ph.D. degrees from NJIT in 2001 and 2004, respectively. He worked in Siemens Ltd., China, as a system engineer for 4 years. He is currently a member of technical staff in Siemens Corporate Research.

NIRWAN ANSARI started schooling at the age of 10 in Hong Kong, proceeded to acquire his B.S. (summa cum laude), M.S., and Ph.D. degrees, all in electrical engineering, from NJIT, University of Michigan, and Purdue University in 1982, 1983, and 1988, respectively.

He joined the Department of Electrical and Computer Engineering at the New Jersey Institute of Technology as an Assistant Professor in 1988, and was promoted to a full professor in 1997. He authored with E.S.H. Hou *Computational Intelligence for Optimization* (Kluwer, 1997, translated into Chinese in 2000), and edited with B. Yuhas *Neural Networks in Telecommunications* (Kluwer, 1994). He is a technical editor of the *IEEE Communications Magazine*, *Computer Communications*, the *ETRI Journal*, as well as the *Journal of Computing and Information Technology*. His current research focuses on various aspects of broadband networks and multimedia communications. His research has been supported by various federal and state agencies, and private industries. He has also contributed over 80 refereed journal articles plus numerous conference papers and book chapters.

He organized (as General Chair) the First IEEE International Conference on Information Technology: Research and Education (ITRE2003), was instrumental, while serving as its Chapter Chair, in rejuvenating the North Jersey Chapter of the IEEE Communications Society which received the 1996 Chapter of the Year Award and a 2003 Chapter Achievement Award, served as Chair of the IEEE North Jersey Section and in the IEEE Region 1 Board of Governors during 2001–2002, and currently serves in various IEEE committees including TPC Co-chair/Vice-chair of several conferences. In 1998 he received the NJIT Excellence Teaching Award in Graduate Instruction and in 1999 was the recipient of the IEEE Region 1 Award. He is frequently invited to deliver keynote addresses, tutorials, and speeches. He was recently selected as an IEEE Communications Society Distinguished Lecturer.
