Fair Bandwidth Allocation for Assured Forwarding (AF) Services†

Li Zhu and Nirwan Ansari
Advanced Networking Laboratory
Department of Electrical and Computer Engineering
New Jersey Institute of Technology
Newark, NJ 07012–1982, USA.
email: {lz6, nirwan.ansari}@njit.edu

Abstract—Fair bandwidth allocation is one of the most challenging research issues in the context of Assured Forwarding (AF) in the Differentiated Services (DiffServ) networks. There exist many works that tried to assure the fairness of bandwidth allocation. However, these works only focused on studying the simple case, in which multiple AF flows share a single bottleneck link, and they also lacked a solid theoretical analysis to validate themselves. In this paper, we propose a Network-assist Packet Marking (NPM) scheme to offer fair bandwidth allocation among multiple aggregates. By both theoretical analysis and experimental evaluation, we demonstrate that NPM can fairly distribute bandwidth among these aggregates in both single and multiple bottleneck link networks.

I. INTRODUCTION

The Differentiated Services (DiffServ) architecture is a promising means to provide QoS in the Internet. DiffServ provides service differentiation at the service class level instead of per flow level. In DiffServ, three service classes have been defined: Expedited Forwarding (EF), Assured Forwarding (AF), and Best Effort (EB). There are three key components in AF: Service Level Agreements (SLAs) between users and ISPs, packet markers at the network edges, and queue management in the core of the network. SLAs specify users’ traffic profiles, which can be described by Commitment Information Rates (CIRs) and Peak Information Rates (PIRs). At network edges, packet markers are used to mark packets according to their conformance to their traffic profiles. Packets that conform to the SLAs are marked as “IN”, and those do not as “OUT”. In the core of the network, “IN” packets are given higher service priority than “OUT” packets.

Two packet marking models have been proposed for AF services: per-flow based and per-aggregate based. In the per-flow based model [1]–[4], each individual flow has its own traffic profile and is marked by a dedicated packet marker at the network edges. In the per-aggregate based model [3], [5]–[8], flows in the same aggregate are characterized by one traffic profile and marked by one marker. In this paper, we adopt the aggregate-based marking approach because SLAs between customers and ISPs are usually made at the aggregate level rather than the per-flow level.

†This work has been supported in part by the National Science Foundation under Grant 0435250, and the New Jersey Commission on Science and Technology via the NJ Center for Wireless Networks and Internet Security.

In the context of AF services, a widely adopted fairness criterion of bandwidth allocation is that bandwidth should be allocated to aggregates (or flows if per-flow model is used) in proportion to their subscription rates (CIRs) [1]–[8]. It has been shown that AF bandwidth is unfairly distributed in favor of traffics with smaller subscription rates [3]. Works in [7]–[10] have shown that TCP round trip times (RTTs), subscription rates, and the number of flows in the aggregates can significantly affect the fairness.

Several schemes have been proposed to improve the fairness of AF bandwidth allocation. Authors in [1] proposed Adaptive Packet Marking (APM), which can achieve very good fairness for TCP traffics. However, APM needs to change the TCP scheme itself. Proposals in [2], [4] aimed to reduce the unfairness caused by the heterogeneity of RTTs and subscription rates. These two schemes are based on certain TCP throughput models, and they may not work well if end users adopt different protocols. Active Rate Management (ARM) was proposed in [6] to achieve fair bandwidth allocation among TCP aggregates. However, ARM has difficulty to guarantee fairness in both under-subscription and over-subscription cases. Authors in [5], [7] extended the idea of “TCP trunking” [11] to provide fair bandwidth allocation among aggregates. However, these studies have some shortcomings: first, they do not provide theoretical analysis to guarantee the fairness; second, they only perform simulations, in which each aggregate has the same target rate, and how these schemes work in a more general case remains unclear. An adaptive marking scheme was proposed in [8] to allocate bandwidth to aggregates in proportion to their target rates. Although simulations on a multiple bottleneck link network was provided in this work, it showed its effectiveness for only a specific case, and it may still have difficulty to assure fairness in a more general case.

Although these remedies can improve fairness from different aspects at different degrees, they have some common drawbacks. First, they do not provide fairness criteria for networks with multiple bottleneck links, since the fairness criterion that each aggregate receives bandwidth proportional to its target rate is only suitable to networks with single bottleneck link. Second, they did not provide solid theoretical analysis to show that the fairness can be guaranteed, especially in multiple bottleneck link networks.

In this paper, we first extend the existing fairness criterion
to networks with multiple bottleneck links. Based on the extended fairness criterion, a novel adaptive packet marking scheme, Network-assist Packet Marking (NPM) is proposed. We use both theoretical and experimental studies to show that NPM can achieve fair bandwidth allocation in both single and multiple bottleneck link networks, regardless of the RTT, subscription rate, and the number flows of each aggregate.

The remainder of this paper is organized as follows. Section II presents the necessary background. Section III presents the main ideas of NPM. Section IV presents the simulation results, followed by conclusions in Section V.

II. BACKGROUND

Assume there are \( N \) TCP aggregates competing bandwidth in the network. Denote \( R_{0,j} \), \( R_{f,j} \), and \( R_j \) as the subscription rate, fair share rate, and the received rate of the \( j^{th} \) \( (j = 1, 2, \ldots, N) \) aggregate, respectively. The packets in each aggregate are marked by one marker at the network edges. There are two types of packet makers: Time Sliding Window (TSW) marker characterized by target rate \( R_{t,j} \), and Token Bucket (TB) marker characterized by target rate \( R_{t,j} \) and bucket depth \( B_j \). At the early stage of study on AF services, people used to set the target rate of the markers to the subscription rate of each aggregate, i.e., \( R_{t,j} = R_{0,j} \). However, studies have shown that this approach leads to unfairness, and many remedies have been proposed [1], [2], [5], [6], [9], [10]. The basic idea behind these new approaches is to dynamically change the target rate \( R_{t,j} \) (and \( B_j \) if TB is used) so that the fairness is improved.

Based on the fairness criterion introduced in Section I, the fair share rate of aggregate \( j \) in a single bottleneck network should be proportional to its subscription rate:

\[
R_{f,j} = \frac{R_{0,j}}{\sum_{k=1}^{N} R_{0,k}} C, \tag{1}
\]

where \( C \) is the bottleneck link bandwidth. If the received rate of each aggregate equals to its fair share, i.e., \( R_j = R_{f,j} \), we say the fairness is achieved. In order to quantitatively describe the bandwidth allocation fairness, we introduce the following two definitions:

**Definition 1:** The fairness ratio \( F_j \) of flow \( j \) is defined as the ratio between its received rate and its fair share rate:

\[
F_j = \frac{R_j}{R_{f,j}}. \tag{2}
\]

**Definition 2:** The fairness index \( F \) of the bandwidth allocation in the entire network is defined as:

\[
F = \frac{\min_j \{F_j\}}{\max_j \{F_j\}}. \tag{3}
\]

Note that \( F \leq 1 \). The closer \( F \) is to one, the fairer it is.

In this paper, we adopt RIO [12] in the core routers. In RIO, \( p_{in} \) and \( p_{out} \), the dropping probability for “IN” and “OUT” packets, are computed based on the average queue length \( q \). Interested readers are referred to [12] for more details.

III. NETWORK-ASSIST PACKET MARKING (NPM)

Before presenting NPM, we first extend the fairness criterion for single bottleneck networks to multiple bottleneck networks. The new fairness adopted in this paper is weighted max-min fairness, in which the weight is the subscription rate \( R_{0,j} \) for aggregate \( j \). It can be shown that weighted max-min fairness is reduced to (1) in the single bottleneck case. Therefore, weighted max-min fairness is a natural extension of the fairness described in (1).

In NPM, two steps are used to achieve this new fairness: first, the weighted max-min fair share rate for each aggregate is computed; second, after the fair share rate for each aggregate has been computed, each ingress edge node applies an adaptive marking algorithm to regulate each aggregate at the network edge so that its received rate equals to its fair share rate. We will show the details of these two steps in the remaining of this section.

A. Computation of Fair Share Rate in NPM

Weighted max-min fairness is a straightforward extension of max-min fairness, and many algorithms used to compute max-min fair share rates can be modified to compute weighted max-min fair share rates in the context of ATM networks [13]. In this paper, we borrow the idea in [14] to compute the fair share rates. In [14], each node keeps the information of the crossing flows, and computes the fair share rate for each of them in a distributed manner. In the context of the AF service with aggregate marking model, only the fair share rate of each aggregate needs to be computed. Thus, each node only needs to keep the information of the crossing “aggregates” instead of the flows. The number of aggregates is much less than the number of flows, and keeping per aggregate information at the core nodes becomes affordable. Therefore, it is reasonable and practical to borrow the idea in [14] at the aggregate level. Next, following the similar approach to [14], we present the method in NPM to compute the fair share rates.

In order to compute fair share rates, the ingress edge nodes and the egress edge nodes need to communicate by periodically sending special control packets to each other. There are two types of feedback packets: forward packets, which are sent from ingress nodes to egress nodes, and backward packets, which are sent from egress nodes to ingress nodes. The control packet contains four fields: aggregate ID field, aggregate weight field, stamped rate field, and underloading field. Based on the ID field in each control packet, core nodes are able to identify all the crossing aggregates. The aggregate weight field is set to the subscription rate \( R_{0,j} \) for the \( j^{th} \) aggregate, and this value is used as the weight to compute the weighted max-min fair share rate for aggregate \( j \). The stamped rate field and the underloading field are used to compute the fair share rates. When the ingress edge node sends out a new forward packet, the value of the underloading field is set to zero. The purposes and usage of these two fields will be clearer later on in this section.

Each node keeps the “advertised rate” \( \mu_j \) for aggregate \( j \) that crosses it, and \( \mu_j \) represents the available bandwidth for aggregate \( j \). Upon receiving a control packet for aggregate \( j \), the node compares the “stamped rate” in the stamp rate field with the “advertised rate”. If the “advertised rate” is lower than the “stamped rate”, the stamp rate field is changed to the “advertised rate” and the underloading field is set...
to one. Otherwise, the control packet is unchanged. When the egress edge node receives the control packet, it sends it back as a backward packet to the corresponding ingress edge node. After receiving this backward packet, if the value of the underloading field is one, the ingress edge node sets the “stamped rate” in the next outgoing control packet to the “stamped rate” of the received feedback packet. Otherwise, the “stamped rate” in the next outgoing forward packet is set to infinity.

Next, we show how to compute the “advertised rate” for each aggregate [14]. Each node has a list of the crossing aggregates, and keeps track of their latest seen stamped rates, which are called “recorded rates”. It also keeps another two lists: unrestricted list $U$ and the restricted list $R$. $U$ includes aggregates whose recorded rates are higher than their “advertised rates”, and the remaining aggregates are in the set $R$. Assume that $j \in U$. Upon the reception of a new control packet from $j$, the “advertised rate” for $j$ is recomputed as:

$$\mu_j = \frac{C - C_R}{\sum_{k \in U} R_{0,k}} R_{0,j},$$

where $C_R$ is the sum of all the recorded rates of the aggregates in the restricted list $R$, and $C$ is the link capacity. Note that (4) is different from the procedure adopted in [14], since what we need to compute is the weighted max-min fair rate instead of the max-min fair rate.

B. Adaptive Marking in NPM

In this paper, we adopt the Time Sliding Window (TSW) type of marker. At the beginning, the marker’s target rate $R_{t,j}$ is initialized to the subscription rate $R_{0,j}$. Each ingress node uses a variable $R'_{f,j}$ to track the stamped rate in the backward packet. Upon the arrival of a new backward control packet, if the stamped rate in this packet is different from $R'_{f,j}$, $R'_{f,j}$ is updated to the new value, and the target rate ($R_{t,j}$) of the marker is also updated to the new $R'_{f,j}$

$$R_{t,j} = R'_{f,j}.$$ (5)

We know that $R'_{f,j}$ will converge to the fair rate $R_{f,j}$ within finite time [14]. Therefore, $R_{t,j}$ will be stabilized at the fair share rate $R_{f,j}$. We call this marking scheme the Simple Network-assist Packet Marking (S-NPM). When all $R_{t,j}$ is stabilized, all the bottleneck links in the network are exactly provisioned. In other words, the bandwidth of each bottleneck link is equal to the sum of the target rates of all aggregates crossing it. In this case, the received rate of aggregate $j$ is [8]

$$R_j = \frac{(1 - p_{in}) R_{f,j}}{2} \left[1 + \sqrt{1 + \left(\frac{\sqrt{6N_j}}{(1 - p_{in})R_{f,j}T_j}\right)^2}\right],$$ (6)

where $p_{in}$ is the dropping probability of “IN” packets, $N_j$ is the number of TCP flows in aggregate $j$, and $T$ is the round trip time of aggregate $j$. It can be reasonably assumed that $p_{in}$ is very small. If $R_{f,j}T_j/N_j \gg 1$, we have

$$R_j \approx R_{f,j},$$ (7)

which means the received rate for each aggregate is approximately equal to its fair share rate. However, $R_{f,j}T_j/N_j \gg 1$ is not always true, e.g., $N_j = 20$, $T_j = 0.05$ sec, $R_{f,j} = 625$ packets/sec = 5Mbps (assume 1000 bytes per packet), $R_{f,j}T_j/N_j = 1.55 \gg 1$. Therefore, S-NPM sometimes may not guarantee fairness. In order to overcome the disadvantage of S-NPM, we propose Enhanced Network Packet Marking (E-NPM).

In E-NPM, each egress edge node monitors the received rate $R_j$ of the aggregate and feeds it back to the corresponding ingress node by attaching it to the backward packet. At the arrival of the backward packet, each ingress edge node checks the stamped rate. If it is different from the old value of $R'_{f,j}$, the target rate ($R_{t,j}$) of the marker is updated according to (5). Otherwise, $R_{t,j}$ is updated as:

$$\begin{align*}
&\text{if } R_j \geq (1 + \delta)R'_{f,j}, \quad R_{t,j} \rightarrow (1 - \beta)R_{t,j} \\
&\text{if } R_j \leq (1 - \delta)R'_{f,j}, \quad R_{t,j} \rightarrow (1 + \beta)R_{t,j} \\
&\text{otherwise, } \quad R_{t,j} \rightarrow R_{t,j}
&\text{if } R_j \geq (1 + \delta)R'_{f,j}, \quad R_{t,j} \rightarrow (1 - \beta)R_{t,j} \\
&\text{if } R_j \leq (1 - \delta)R'_{f,j}, \quad R_{t,j} \rightarrow (1 + \beta)R_{t,j} \\
&\text{otherwise, } \quad R_{t,j} \rightarrow R_{t,j}
\end{align*}$$ (8)

Here, $\delta$ and $\beta$ are two real positive numbers, which are very close to zero. Since $R'_{f,j}$ converges to $R_{f,j}$ very soon, (8) will finally become:

$$\begin{align*}
&\text{if } R_j \geq (1 + \delta)R_{f,j}, \quad R_{t,j} \rightarrow (1 - \beta)R_{t,j} \\
&\text{if } R_j \leq (1 - \delta)R_{f,j}, \quad R_{t,j} \rightarrow (1 + \beta)R_{t,j} \\
&\text{otherwise, } \quad R_{t,j} \rightarrow R_{t,j}
\end{align*}$$ (9)

Proposition 1: When the E-NPM scheme is stabilized, the fairness index of the bandwidth allocation in the entire network is

$$1 \geq F \geq \frac{1 - \delta}{1 + \delta},$$ (10)

Proof: From (9), we know that when E-NPM is stabilized, the target rate $R_{t,j}$ for the marker is also stabilized. Thus, the fair ratio of aggregate $j$ satisfies:

$$\frac{1}{1 - \delta} \leq F_j \leq \frac{1}{1 + \delta},$$ (11)

Based on the definition of fairness index

$$F = \min_j \{F_j\},$$ (12)

and combining (11) and (12), we have

$$1 \geq F \geq \frac{1 - \delta}{1 + \delta},$$ (13)

This completes the proof.

Proposition 1 tells us that the fairness index under E-NPM scheme is very close to one, i.e., E-NPM can fairly distribute bandwidth among aggregates. In the next section, we will use simulations to demonstrate E-NPM’s capability to provide fair bandwidth allocation.

IV. SIMULATIONS

In this section, we evaluate our proposals in both single and multiple bottleneck link networks. RIO [12] is adopted in the core routers to provide service differentiation between “IN” and “OUT” packets. The parameters for RIO are set as follows: $(q^{\min}_{int}, q^{ \max}_{int}, q^{\min}_{out}, q^{\max}_{out}) = (50, 150, 0.1)$ and $(q^{\min}_{in}, q^{\max}_{in}, q^{\max}_{in}) = (150, 500, 0.02)$. Both $\alpha$ and $\beta$ in E-NPM are set to 0.01.
A. Single Bottleneck Link Network

In this set of simulations, we study the performance of our proposals in a network with one bottleneck link, which is shown in Fig. 1. The link from core router $C_1$ to core router $C_2$ is the only bottleneck, and $E_1$ – $E_4$ are the edge routers. There are two TCP aggregates, each of which consists of 20 long-lived TCP flows by default. The first aggregate is from $S_1$ to $R_1$, and the second aggregate is from $S_2$ to $R_2$. The TCP packet size is 1000 bytes.

First, we compare the performance of S-NPM, E-NPM, and the original TSW. The subscription rate $R_{0,1}$ of aggregate 1 is fixed to 10Mbps, and the subscription rate $R_{0,2}$ of aggregate 2 varies between 1Mbps and 30Mbps. The fairness indices (computed based on (2) and (3)) under these schemes are shown in Fig. 2. It is clear that both S-NPM and E-NPM have better performance than the original TSW. When $R_{0,2}$ is between 1Mbps and 5Mbps, the fairness index of S-NPM is significantly smaller than one, and this demonstrates that S-NPM sometimes has difficulty to guarantee fairness. The reason is that the assumption $R_{f,j}T_{j}/N_{j} \gg 1$ and (7) do not always hold. On the other hand, the fairness index of E-NPM is always very close to one, demonstrating that E-NPM can guarantee fair bandwidth allocation all the time. In the remaining of this paper, we focus on the simulation study on the performance of E-NPM.

Next, we study the effect of RTT of TCP aggregates on the performance of E-NPM, and results are shown in Fig. 3 and Fig. 4. The propagation delay between $E_1$ and $C_1$ is fixed to 50ms, and the propagation delay between $E_2$ and $C_1$ is varied between 10ms and 150ms. We consider two cases: under-subscription and over-subscription. In the under-subscription case, the subscription rates for aggregate one and two are fixed to (5Mbps, 10Mbps). According to the fairness criterion, the fair rates are (6.667Mbps, 13.333Mbps), which are shown by two solid horizontal lines in Fig. 3. In the over-subscription case, the subscription rates for aggregate one and two are fixed to (40Mbps, 10Mbps), and the corresponding fair rates are (16Mbps, 4Mbps), which are also shown by two solid horizontal lines in Fig. 4. From Fig. 3 and Fig. 4, we can see that in both cases, the received rates for two aggregates are very close to their fair share rates, and this demonstrates that RTT has very little impact on the performance of E-NPM.

We also investigate how the number of TCP flows in the aggregate can affect the performance of E-NPM. We still consider two cases: under-subscription and over-subscription. There are 20 TCP flows in aggregate 1, and the number of TCP flows in aggregate 2 varies between 20 and 80. The subscription rates for aggregates 1 and 2 are the same as those in the simulations for Fig. 3 and Fig. 4. Fig. 5 and Fig. 6 show the results. Again, we can see that E-NPM can achieve fair rate allocation regardless of the number of TCP flows in the aggregate.

B. Multiple Bottleneck Link Network

In this simulation, we study the performance of E-NPM in the multiple bottleneck link network, which is shown in Fig. 7. $C_1$ – $C_3$ are core routers, and $E_1$ – $E_5$ are the edge routers. The link between $C_1$ and $C_2$ and the link between $C_2$ and $C_3$ are the bottlenecks. There are three TCP aggregates,
V. CONCLUSIONS

In this paper, we have introduced a new packet marking scheme called NPM (S-NPM and E-NPM), which can be used for AF services in the DiffServ networks. NPM is based on per aggregate marking, and it only introduces a slight computation overhead in the core routers. We have also extended the fairness criterion of bandwidth allocation from the single bottleneck case to the multiple bottleneck case. By both theoretical analysis and experimental studies, we have shown that both S-NPM and E-NPM significantly outperform the original TSW packet marking scheme. Moreover, E-NPM can assure fair bandwidth allocation in both single and multiple bottleneck networks, and its performance is not affected by the subscription rate, round trip time, and number of flows in each aggregate.

REFERENCES


