On Generalized Max-Min Rate Allocation and Distributed Convergence Algorithm for Packet Networks

Y. Thomas Hou, *Senior Member*, *IEEE*, Shivendra S. Panwar, *Senior Member*, *IEEE*, and Henry H.-Y. Tzeng

Abstract—This paper considers the fundamental problem of bandwidth allocation among flows in a packet-switched network. The classical max-min rate allocation has been widely regarded as a fair rate allocation policy. But, for a flow with a minimum rate requirement and a peak rate constraint, the classical max-min policy no longer suffices to determine rate allocation since it is not capable of supporting either the minimum rate or the peak rate constraint from a flow. In this paper, we generalize the theory of the classical max-min rate allocation with the support of both the minimum rate and peak rate constraints for each flow. Additionally, to achieve generalized max-min rate allocation in a fully distributed packet network, we present a distributed algorithm that uses a feedback-based flow control mechanism. Our design not only offers a fresh perspective on flow marking technique, but also advances the state-of-the-art flow marking technique favored by other researchers. We provide proof that such a distributed algorithm, through asynchronous iterations, will always converge to the generalized max-min rate allocation under any network configuration and any set of link distances. We use simulation results to demonstrate the fast convergence property of the distributed algorithm.

Index Terms—Max-min rate allocation, minimum rate, peak rate, centralized algorithm, distributed algorithm, convergence, flow control, packet networks.

1 Introduction

A fundamental problem in a packet-switched network is bandwidth (or rate) allocation among flows such that network bandwidth is "optimally" utilized. Such optimality usually includes the following two components: 1) *efficiency*—network bandwidth should be used as much as possible by the flows and 2) *fairness*—network bandwidth should be shared according to some fairness criterion.

The classical max-min rate allocation has been widely regarded as an optimal rate allocation policy [2]. The classical max-min approach achieves fairness by maximizing the minimum flow rate allocation in the network without exceeding a link's capacity. It follows that, under such a max-min rate allocation, each flow must pass through at least one *bottleneck link* [2]. The classical max-min rate allocation, in its current form [2], assumes a zero minimum rate requirement and no peak rate constraint. However, in practice, many media-rich real-time network applications require a certain minimum bandwidth in order to guarantee a minimum acceptable quality of service. Most of these multimedia applications are equipped with *rate-adaptive* encoders. Depending on available bandwidth in the network, these encoders can adjust their output rate to further enhance an application's quality. Additionally, in practice, there are situations where that is also an upper bound (peak rate) on an application's output rate. For example, the network's interface card (e.g., a modem) may impose a physical limit on the speed of the encoder's output rate. As another example, in a corporate network, where the network access link is shared by all users, a peak rate constraint may be imposed (through bandwidth management software at the access link) on each individual user. Consequently, a practical network application may have both MR and PR constraints. It is, therefore, essential to devise a rate allocation policy that will optimally support both of these constraints.

This paper presents a fundamental study on network bandwidth allocation based on the classical max-min approach. Our main contributions are twofold: 1) *Centralized theory:* We generalize the theory of the classical max-min rate allocation with minimum rate (MR) and peak rate (PR) support (the so-called generalized max-min (GMM) rate allocation); 2) *Distributed algorithm:* We have designed a feedback-based distributed algorithm that is proven to converge to GMM through asynchronous iterations under any network configuration and any set of link distances. In particular, our design of the distributed algorithm offers a fresh perspective on flow marking technique. By exploring the limits of flow marking technique by other researchers, we generalize such a technique and make it more flexible. Consequently, our new marking technique can be used to design a broader class of distributed algorithms.

Prior efforts on extending the classical max-min rate allocation with minimum rate support include the so-called *MR-add* policy and the *MR-prop* policy [20]. Both policies first guarantee the minimum rate of each flow. Under MR-add, remaining network bandwidth is shared among all flows using the max-min policy, i.e., equal weight for all flows;

- *Y.T. Hou is with The Bradley Department of Electrical and Computer Engineering, Virginia Tech, 340 Whittemore Hall (0111), Blacksburg, VA 24061. E-mail: thou@vt.edu.*
- *S.S. Panwar is with the Department of Electrical and Computer Engineering, Polytechnic University, Six Metrotech Center, Brooklyn, NY 11201. E-mail: panwar@catt.poly.edu.*
- *H.H.-Y. Tzeng is with Nokia, Networks Division, 313 Fairchild Drive, Mountain View, CA 94043. E-mail: Henry.Tzeng@nokia.com.*

Manuscript received 15 June 2002; revised 12 May 2003; accepted 21 Sept. 2003.
For information on obtaining reprints of this article, please send e-mail to: tpds@computer.org, and reference IEEECS Log Number 115710.

under MR-prop, remaining network bandwidth is shared among all flows using an MR-proportional max-min policy. In [8], a generic weight-based network bandwidth allocation policy, called *Weight-Proportional Max-Min* (WPMM), was proposed to generalize the MR-add and MR-prop policies. Under WPMM, each flow is associated with a generic weight, which is *decoupled* (or *independent*) from its MR. After first allocating each flow with its MR, the WPMM policy shares the remaining network bandwidth among all flows based on each flow's weight. While the MR-add, MR-prop, and WPMM rate allocations can all support MR for each flow, they are merely trivial extensions of the classical max-min approach. In contrast, the *Generalized Max-Min* (GMM) rate allocation presented in this paper generalizes the classical max-min rate allocation by exploring its underlying principle.

The second part of this paper is devoted to the problem of designing a distributed algorithm that converges to GMM rate allocation. The motivation for this effort is based on the observation that a centralized algorithm will require global information about the network, which is difficult to maintain and manage in a large scale network. Thus, it is critical to develop a distributed implementation to achieve GMM rate allocation. There has been extensive previous work on the design of distributed algorithms to achieve the classical max-min rate allocation. Early algorithms by Hayden [6], Jaffe [10], and Gafni [5] required synchronization of all nodes for each iteration. Mosely's work in [14] was the first distributed algorithm allowing asynchronous computation. Unfortunately, this algorithm could not offer satisfactory convergence performance. Later, Ramakrishnan et al. [15] proposed using a single bit to indicate congestion and achieve max-min. But, due to the binary nature of this algorithm, the source's rate exhibited oscillations. In the past few years, research in ATM ABR flow control has led to many contributions to the design of distributed algorithms to achieve the classical max-min (see, e.g., [4], [11], [12], [16], [17], [19]). In particular, the seminal work by Charny et al. [4], also known as the *Consistent Marking* (CM) algorithm, was one of the few algorithms that were proven to converge to the classical max-min.

As we shall see in the second half of this paper, Charny et al.'s Consistent Marking algorithm cannot be applied to GMM rate allocation due to its intrinsic design limitation. In particular, if we apply Consistent Marking in a network for GMM rate allocation (with MR and PR), the rate of each flow will oscillate and never converge to any rate allocation policy (see Section 3.2 for more details). In this paper, we explore the limits of the Consistent Marking technique and propose a more general flow marking technique that advances the existing Consistent Marking technique. We show that our new flow marking technique can be used to design a broader class of distributed convergence algorithms, including that for the GMM rate allocation.

The remainder of this paper is organized as follows: In Section 2, we first review key results for the classical max-min rate allocation. Then, we present the theory for the Generalized Max-Min (GMM) rate allocation with the support of MR and PR from each flow. Section 3 shows how we generalize the Consistent Marking technique. We also show how a distributed algorithm for GMM rate allocation can be designed by applying the generalized technique. In Section 4, we give a correctness proof of the convergence of the distributed algorithm. Section 5 shows simulation results for the distributed algorithm on several network configurations and demonstrates the fast convergence property of the algorithm. Section 6 concludes this paper.

2 Generalizing the Classical Max-Min Theory

In this section, we generalize the classical max-min rate allocation with MR and PR constraints. In Section 2.1, we first summarize key results of the classical max-min rate allocation. Section 2.2 presents the theory of Generalized Max-Min (GMM) rate allocation.

2.1 A Brief Review of Classical Max-Min

In our model, a network of switches are interconnected by a set of links $\mathcal{L}$. A set of flows $s \in \mathcal{S}$ traverses one or more links in $\mathcal{L}$; each flow is allocated a specific rate r_s. Denote $\mathcal{S}_\ell$ the set of flows traversing link $\ell \in \mathcal{L}$. Then, the (aggregate) allocated rate F_ℓ on link ℓ is $F_\ell = \sum_{s \in \mathcal{S}_\ell} r_s$. Let C_ℓ be the capacity of link $\ell \in \mathcal{L}$. A link ℓ is *saturated* or *fully utilized* if $F_\ell = C_\ell$. A rate vector $r = \{r_s \mid s \in \mathcal{S}\}$ is *feasible* if the following two constraints are satisfied: 1) $r_s \geq 0$ for all $s \in \mathcal{S}$ and 2) $F_\ell \leq C_\ell$ for all $\ell \in \mathcal{L}$.

A rate vector $r = \{r_s \mid s \in \mathcal{S}\}$ is *max-min* if it is feasible, and if, for each flow s, one cannot generate a new feasible rate vector by increasing the allocated rate r_s without decreasing the allocated rate of some other flow t with a rate r_t already less than or equal to r_s in the rate vector r. Formally, the classical max-min rate allocation can be defined as [2]: A rate vector $r = \{r_s \mid s \in \mathcal{S}\}$ is *max-min* if it is feasible and if, for each $s \in \mathcal{S}$ and every feasible rate vector $\hat{r} = \{\hat{r}_s \mid s \in \mathcal{S}\}$ in which $\hat{r}_s > r_s$, there exists some flow $t \in \mathcal{S}$ such that $r_s \geq r_t$ and $r_t > \hat{r}_t$.

Given a feasible rate vector $r = \{r_s \mid s \in \mathcal{S}\}$, a link $\ell \in \mathcal{L}$ is a *bottleneck link* with respect to r for a flow $s \in \mathcal{S}_\ell$ if $F_\ell = C_\ell$ and $r_s \geq r_t$ for all flows $t \in \mathcal{S}_\ell$. A feasible rate vector $r = \{r_s \mid s \in \mathcal{S}\}$ is max-min if and only if each flow $s \in \mathcal{S}$ has a bottleneck link with respect to r.

Definition 1. *Given a max-min rate vector $r = \{r_s \mid s \in \mathcal{S}\}$ and suppose that link $\ell \in \mathcal{L}$ is a max-min bottleneck link with respect to r. Denote λ_ℓ the max-min bottleneck link rate at ℓ. Then, λ_ℓ satisfies*

$$\lambda_\ell = \max_{s \in \mathcal{S}_\ell} r_s = \frac{C_\ell - \sum_{i \in \mathcal{M}_\ell} r_i}{|\mathcal{U}_\ell|},$$

where $\mathcal{U}_\ell$ denotes the set of flows that are bottlenecked at link ℓ,[1] and $\mathcal{M}_\ell$ denotes the set of flows that traverse link ℓ but are bottlenecked at some other link and $r_i < \lambda_\ell$ for $i \in \mathcal{M}_\ell$.

The following iterative steps describe a centralized algorithm to determine max-min rate allocation [2]:

1. Start the rate of each flow with zero.
2. Increase the rate of each flow currently having the smallest rate until some link becomes saturated.
3. Remove those flows that traverse saturated links and the capacity associated with these flows from the network.
4. If there is no flow left, the algorithm terminates; otherwise, go back to Step 2 for the remaining flows and remaining network capacity.

It can be shown that there exists a unique rate vector that satisfies the max-min rate allocation.

1. $|\mathcal{U}_\ell|$ denotes the number of flows in set $\mathcal{U}_\ell$.

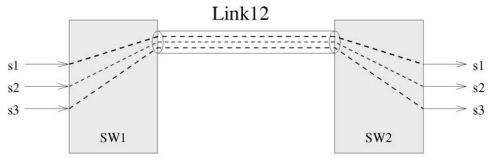

Fig. 1. A peer-to-peer network.

2.2 Generalized Max-Min Rate Allocation

Before we formally define GMM rate allocation, we use the following example to illustrate its concept. For generality, we assume that all links have one unit of capacity. In the *peer-to-peer* network configuration (Fig. 1), the output port link (Link12) of switch 1 (SW1) is the only potential bottleneck link for all flows. In the classical max-min case where there are no MR and PR constraints for each flow, the centralized max-min rate allocation algorithm allocates each flow with a rate of $1/3$. Now, let the MR requirement and PR constraint for each flow be as listed in Table 1. We describe the iterative steps of a centralized algorithm under the GMM rate allocation.

Algorithm 1: (**A Centralized Algorithm for GMM—An Informal Description**)[2]

1. Start the rate of each flow with its MR, and sort all flows in the order of increasing MR.[3]
2. Increase the rate of the flow with the smallest rate among all flows until one of the following events takes place:
   a. The rate of such flow reaches the second smallest rate among the flows.
   b. The rate of such flow reaches its PR.
   c. Some link saturates.
3. If some link saturates or the flow's rate reaches its PR in Step 2, remove the flows that either traverse the saturated link or reach their PRs, respectively, as well as the rates associated with these flows from the network capacity.[4]
4. If there is no flow left, the algorithm terminates; otherwise, go back to Step 2 for the remaining flows and network capacity.

The above centralized algorithm for GMM rate allocation enables us to complete the rate allocation problem for the peer-to-peer network configuration (Fig. 1) with the MR and PR constraints (Table 1), which we elaborate as follows.

Example 1 (A peer-to-peer network). In this example, we compute the rate allocation problem for the peer-to-peer network configuration (Fig. 1) with the MR and PR constraints (Table 1). Fig. 2 shows these iterations as a "water-filling" process.

- Initialization: As shown in Fig. 2, we start the rate of each flow with its MR (shown in the darkest shaded areas in Fig. 2).

2. A formal mathematical description of this algorithm is given in Algorithm 2.

3. In the case when there are multiple flows with the same MR, these flows will be put into a set and will be considered jointly.

4. It is worth pointing out that the PR constraint can be considered by introducing a virtual link of capacity PR at every source. That is, the PR for a flow is equivalent to the capacity of the flow's virtual access link.

TABLE 1
MR Requirement, PR Constraint, and GMM Rate Allocation of Each Flow in the Peer-to-Peer Network Configuration

Flow	MR	PR	GMM Rate Allocation
$s1$	0.40	1.00	0.40
$s2$	0.10	0.25	0.25
$s3$	0.05	0.50	0.35

- First iteration: Since the rate of $s3$ (0.05) is the smallest among all flows, we increase it until it reaches the second smallest rate, which is 0.1 ($s2$).
- Second iteration: The rates of both $s2$ and $s3$ being 0.1, we increase them together until $s2$ reaches its PR constraint of 0.25. Remove $s2$ (with a rate of 0.25) from future iterations, and we now have rates of 0.40 and 0.25 for $s1$ and $s3$, respectively, with a remaining capacity of 0.10 on Link 12.
- Third iteration: Since $s3$ has a smaller rate (0.25) than $s1$ (0.4), we increase the rate of $s3$ to 0.35 and Link12 saturates. The final rate allocations for $s1$, $s2$, and $s3$ are 0.40, 0.25, and 0.35, respectively.

The above example illustrates the basic concept of GMM rate allocation: always maximize the minimum rate among all flows, while, at the same time, satisfying each flow's PR constraint and link capacity constraint. Therefore, GMM rate allocation preserves the basic principle as the classical max-min. Denote MR_s and PR_s the minimum rate requirement and the peak rate constraint for each flow $s \in \mathcal{S}$. For feasibility, we must have the following assumption.

Assumption 1. *The sum of all flows' minimum rate traversing any link is less than the link's capacity, i.e.,* $\sum_{s \in \mathcal{S}_\ell} \mathrm{MR}_s < C_\ell$ *for every* $\ell \in \mathcal{L}$.

This condition can be enforced by admission control during call setup time. It is worth pointing out that we use the strict inequality in Assumption 1. Although this may not be necessary for the centralized algorithm, it is essential to maintain *stability* in the the distributed algorithm and to guarantee the existence of σ_n, which will be defined in Section 4.

We say that a rate vector $r = \{r_s \mid s \in \mathcal{S}\}$ is *GMM-feasible* if the following two constraints are satisfied: 1) $\mathrm{MR}_s \le r_s \le \mathrm{PR}_s$ for all $s \in \mathcal{S}$ and 2) $F_\ell \le C_\ell$ for all $\ell \in \mathcal{L}$. Formally, GMM rate allocation can be defined as follows: A rate vector $r = \{r_s \mid s \in \mathcal{S}\}$ is *Generalized Max-Min* (GMM) if it is GMM-feasible and if, for every $s \in \mathcal{S}$ and every GMM-feasible rate

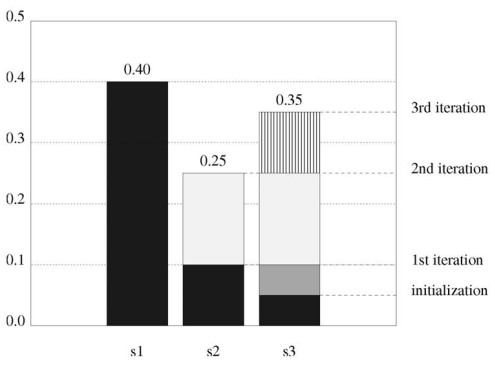

Fig. 2. Iterative steps in rate allocation as a "water-filling" process.

vector $\hat{r} = \{\hat{r}_s \mid s \in \mathcal{S}\}$ in which $\hat{r}_s > r_s$, there exists some flow $t \in \mathcal{S}$ such that $r_s \geq r_t$, and $r_t > \hat{r}_t$. Due to the GMM-feasibility constraint, we define a new notion of bottleneck link as follows.

Definition 2. *Given a GMM-feasible rate vector $r = \{r_s \mid s \in \mathcal{S}\}$, a link $\ell \in \mathcal{L}$ is a GMM-bottleneck link with respect to r for a flow $s \in \mathcal{S}_\ell$ if $F_\ell = C_\ell$ and $r_s \geq r_t$ for every flow $t \in \mathcal{S}_\ell$ for which $r_t > \mathrm{MR}_t$.*

It is crucial to maintain strict inequality $r_t > \mathrm{MR}_t$ (instead of $r_t \geq \mathrm{MR}_t$) in the above definition. In Example 1, according to Definition 2, Link12 is a GMM-bottleneck link for both $s1$ and $s3$. On the other hand, there appears to be potential ambiguity about what we should call the bottleneck link rate here. Note that flows $s1$ and $s3$ have different rate allocation (0.4 for $s1$ and 0.35 for $s3$) and there should be a unique bottleneck link rate at a bottleneck link. The following definition removes this potential ambiguity.

Definition 3 (GMM-Bottleneck Link Rate). *Given a GMM rate vector $r = \{r_s \mid s \in \mathcal{S}\}$ and suppose that link $\ell \in \mathcal{L}$ is a GMM-bottleneck link with respect to r. Denote Λ_ℓ GMM-bottleneck link rate at ℓ. Then, Λ_ℓ satisfies*

$$\Lambda_\ell \cdot \sum_{i \in \mathcal{U}_\ell} 1^+\{\mathrm{MR}_i \leq \Lambda_\ell\} + \sum_{i \in \mathcal{U}_\ell} \mathrm{MR}_i \cdot 1^+\{\mathrm{MR}_i > \Lambda_\ell\} = C_\ell - \sum_{i \in \mathcal{M}_\ell} r_i,$$

where $1^+\{\mathrm{event}\}$ is an indicator function and is defined as 1 if the event is true and 0 otherwise; $\mathcal{U}_\ell$ denotes the set of flows that are GMM-bottlenecked at link ℓ; $\mathcal{M}_\ell$ denotes the set of flows that are 1) either GMM-bottlenecked at some other link or have GMM rate allocation equal to their PRs, and 2) $r_i < \Lambda_\ell$ for $i \in \mathcal{M}_\ell$.

With Definition 3, it is easy to show that GMM-bottleneck link rate at Link12 is 0.35 in Example 1. Also, note that, in the special case when $\mathrm{MR}_s = 0$ for every $s \in \mathcal{S}$, the GMM-bottleneck link rate Λ_ℓ in Definition 3 becomes: $\Lambda_\ell \cdot |\mathcal{U}_\ell| = C_\ell - \sum_{i \in \mathcal{M}_\ell} r_i$, or $\Lambda_\ell = (C_\ell - \sum_{i \in \mathcal{M}_\ell} r_i)/|\mathcal{U}_\ell|$, which is precisely the expression for the classical max-min rate allocation at link ℓ (see Definition 1). Therefore, the classical max-min is indeed a special case under the GMM rate allocation.

The following theorem links the relationship between GMM rate allocation definition and the GMM-bottleneck link definition (Definition 2). The proof is given in the Appendix.

Theorem 1. *A GMM-feasible rate vector $r = \{r_s \mid s \in \mathcal{S}\}$ is GMM if and only if each flow $s \in \mathcal{S}$ has either a GMM-bottleneck link with respect to r or a rate allocation equal its PR.*

In the following algorithm, we present the formal mathematical description of a centralized algorithm for GMM rate allocation, which was informally described earlier in Algorithm 1. We omit its correctness proof to conserve paper space.

Algoritm 2 (**A Centralized Algorithm for GMM Rate Allocation**)

Initial conditions: $k := 0$, $r_s^{(0)} := \mathrm{MR}_s$ for every $s \in \mathcal{S}$,
$F_\ell^{(0)} := \sum_{s \in \mathcal{S}_\ell} \mathrm{MR}_s$ for every $\ell \in \mathcal{L}$;
$k := 1$, $\mathcal{S}^{(1)} := \mathcal{S}$, $\mathcal{L}^{(1)} := \mathcal{L}$.

1. Sort all the flows in $\mathcal{S}^{(k)}$ into m sets ($1 \leq m \leq |\mathcal{S}^{(k)}|$): $u_1, u_2, \cdots, u_m$, such that a) each flow in the same set has the same rate and b) rates in these sets are in increasing order, i.e., $r_{s \in u_1} < r_{t \in u_2} < \cdots < r_{y \in u_m}$.
2. Denote $n_\ell^{(k)}$ as the number of flows $s \in u_1$ traversing link ℓ, for every $\ell \in \mathcal{L}^{(k)}$. Calculate $a^{(k)}$ as follows:

$$a^{(k)} := \begin{cases} \min\Big\{ \min\limits_{\ell \text{ traversed by } s \in u_1} \frac{(C_\ell - F_\ell^{(k-1)})}{n_l^{(k)}}, \\ \quad (r_{t \in u_2} - r_{s \in u_1}), \\ \quad \min\limits_{s \in u_1}(\mathrm{PR}_s - r_s^{(k-1)}) \Big\} & \text{if } m > 1, \\ \min\Big\{ \min\limits_{\ell \text{ traversed by } s \in u_1} \frac{(C_\ell - F_\ell^{(k-1)})}{n_l^{(k)}}, \\ \quad \min\limits_{s \in u_1}(\mathrm{PR}_s - r_s^{(k-1)}) \Big\} & \text{if } m = 1. \end{cases} \quad (1)$$

3. $$r_s^{(k)} := \begin{cases} r_s^{(k-1)} + a^{(k)} & \text{if } s \in u_1; \\ r_s^{(k-1)} & \text{otherwise.} \end{cases}$$
4. $$F_\ell^{(k)} := \sum_{s \in \mathcal{S}_\ell} r_s^{(k)}, \text{ for every } \ell \in \mathcal{L}^k.$$
5. $\mathcal{L}^{(k+1)} := \{\ell \mid C_\ell - F_\ell^{(k)} > 0, \ell \in \mathcal{L}^{(k)}\}$.
6. $\mathcal{S}^{(k+1)} := \{s \mid s$ does not traverse any link $\ell \in (\mathcal{L} - \mathcal{L}^{(k+1)})$ and $r_s^{(k)} \neq \mathrm{PR}_s\}$.
7. $k := k + 1$.
8. If $\mathcal{S}^{(k)}$ is empty, then $r^{(k-1)} = \{r_s^{(k-1)} \mid s \in \mathcal{S}\}$ is the rate vector satisfying GMM rate allocation and the algorithm terminates; otherwise, go back to Step 1.

It is worth noting that, during the iterations of Algorithm 2, if $m > 1$ in Step 1, then the rate values in $u_2, \cdots, u_m$ correspond to the MRs of flows in these sets. Also, starting from the second iteration ($k = 2$), the sorting procedure in Step 1 only requires minor updates based on the sorted sets from the previous iteration. It also follows from Definition 3 and Algorithm 2 that the following property holds for GMM rate allocation: If a rate vector $r = \{r_s \mid s \in \mathcal{S}\}$ is GMM, then the rate allocation for flow $s \in \mathcal{S}$ can only be 1) a rate equal to its MR or 2) a rate equal to its PR or 3) a rate equal to its GMM-bottleneck link rate. It can be shown that there exists a unique rate vector $r = \{r_s \mid s \in \mathcal{S}\}$ satisfying GMM rate allocation.

We use the following multinode example to illustrate the concept of GMM-bottleneck link rate, which follows an ascending order after the iterations in Algorithm 2.

Example 2 (A Three-Node Network). In this network configuration (Fig. 3), the output port links of SW1 (Link12) and SW2 (Link23) are potential GMM-bottleneck links. The MR requirement and PR constraint for each flow are listed in Table 2. For brevity, we will only list the iterations of Algorithm 2 in Table 3, with a graphical display in Fig. 4. The GMM-bottleneck link rate at Link12 is 0.425, which was reached at the end of the fourth iteration, and GMM-bottleneck link rate at Link23

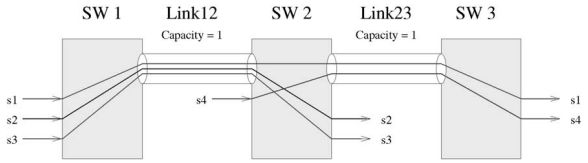

Fig. 3. A three-node network configuration.

is 0.575, which was reached at the end of the fifth iteration, and $0.425 < 0.575$. In general, by the operation of Algorithm 2, a GMM-bottleneck link rate obtained at a later iteration for some link is greater than a GMM-bottleneck link rate obtained at an earlier iteration for some other link.

We have thus completed the centralized theory for GMM rate allocation. In the rest of the paper, we address the more challenging problem of designing a distributed algorithm to achieve GMM rate allocation in a packet network.

3 A Distributed Convergence Algorithm

3.1 Background

We will employ a feedback-based flow control mechanism similar to the ATM ABR service for our distributed algorithm [1]. Such a feedback-based flow control mechanism for a flow is shown in Fig. 5. As we shall soon find out, the convergence property of our algorithm does not depend on the use of the ATM ABR standard. In fact, any flow control mechanism that provides cooperation among the source, destination, and network nodes for each flow can be used to deploy our distributed algorithm. The reason why we choose to use an ATM ABR-like flow control mechanism here is that this mechanism is well documented and understood by the networking community. It is interesting to note that max-min rate allocation has recently found applications in VPN, MPLS, and even WDM networks [18]. Therefore, we expect that the underlying theories and algorithms presented in this paper on GMM rate allocation will be relevant to current and future developments in networking technology.

As shown in Fig. 5, special control packets, called *Resource Management* (RM) cells under ATM ABR, are inserted among the regular data packets to exchange information among network components. A source sets the fields in the forward RM packets to inform the network about the source's rate information (i.e., MR, PR, etc.). While the RM packets traverse switch by switch toward to the destination, the network (switches) extracts the information from the RM packet (through its fields) and performs rate calculation. Upon arriving at the destination, an RM packet is returned back toward the source. Each switch then sets the appropriate fields in the returning RM packet to convey rate allocation information to the source. When a backward RM packet arrives at the source, the source adjusts its rate based on the feedback information in the received RM packet.

TABLE 2
MR Requirement, PR Constraint, and GMM Rate Allocation for Each Flow in the Three-Node Network

Flow	MR	PR	GMM Rate Allocation
$s1$	0.20	0.50	0.425
$s2$	0.05	0.15	0.15
$s3$	0.10	0.50	0.425
$s4$	0.50	1.00	0.575

3.2 Approach

There have been extensive studies on using a feedback-based flow control mechanism (Fig. 5) for the classical max-min rate allocation (see, e.g., [4], [11], [12], [16], [17], [19]). In particular, Charny et al. [4] made a seminal contribution by introducing the so-called *Consistent Marking* technique in the distributed algorithm design. Since our distributed algorithm is based on this work, we briefly summarize Charny et al.'s work here.

In Charny et al.'s algorithm, each switch monitors its traffic by keeping track of the state information for each traversing flow. Also, each output port of a switch maintains a variable, called the *advertised rate*, to calculate the max-min rate allocation for each flow. When an RM packet arrives at the switch, the current rate (CR) value of the flow is stored in a table. If this CR value is less than or equal to the current advertised rate, then the associated flow is assumed to be bottlenecked either at this link or elsewhere, and a corresponding bit for this flow is marked at the table. The following equation is then used to update the advertised rate, φ_ℓ, at link ℓ.

$$\varphi_\ell := \frac{C_\ell - \sum_{i \in \mathcal{M}_\ell} r_\ell^i}{n_\ell - |\mathcal{M}_\ell|}, \tag{2}$$

where C_ℓ is the capacity of link ℓ, $\mathcal{M}_\ell$ is the set of flows marked at link ℓ, r_ℓ^i is the current rate (CR) value of flow i that is just recorded by link ℓ, n_ℓ is the number of flows at link ℓ, and $|\mathcal{M}_\ell|$ is the number of marked flows at link ℓ. Then, the table is examined again. For each marked flow, if its recorded CR (i.e., r_ℓ^i) is larger than this newly calculated advertised rate φ_ℓ, this flow is then unmarked. Finally, the advertised rate is calculated again. The ER field of an RM packet is then set to the minimum of all advertised rates along its traversing links. Charny et al. [4] showed that, eventually, the rate for each

TABLE 3
Iterations of Using the Centralized Algorithm for GMM Rate Allocation for the Three-Node Network

Iterations	Flow(MR, PR)				Remaining capacity	
	$s1$ (0.20, 0.50)	$s2$ (0.05, 0.15)	$s3$ (0.10, 0.50)	$s4$ (0.50, 1.00)	Link12	Link23
Initialization	0.20	0.05	0.10	0.50	0.65	0.30
1st	0.20	0.10	0.10	0.50	0.60	0.30
2nd	0.20	0.15	0.15	0.50	0.50	0.30
3rd	0.20		0.20	0.50	0.45	0.30
4th	0.425		0.425	0.50	0	0.075
5th				0.575		0

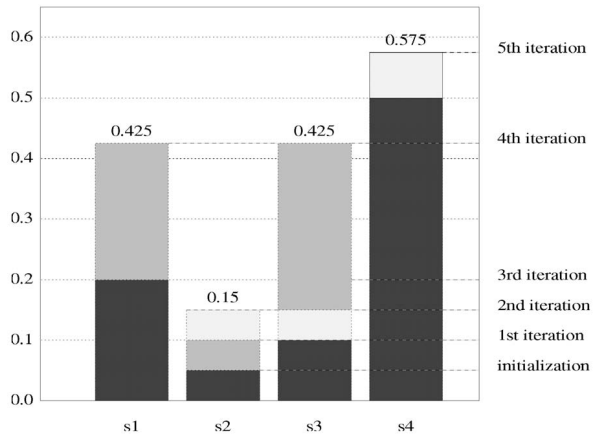

Fig. 4. Graphical display of rate allocation for each flow at each iteration in the three-node network.

flow converges to the max-min rate allocation and is marked along every link that it traverses.

We now consider how to generalize the Consistent Marking technique for GMM rate allocation [9]. To start with, it is apparent that the advertised rate calculation in (2) has to be generalized for GMM-bottleneck link rate (see Definition 3). But, the key problem remains: how to perform marking on each traversing flow at a node so that the rate allocation can converge to GMM rate allocation. We use the following simple example to illustrate that Charny et al.'s original technique will not work here.

Example 3 (A Simple Counter Example). Consider the single bottleneck network in Example 1 with MR/PR constraints in Table 1. If we mark a flow when its CR is less than or equal to the advertised rate (as in the Consistent Marking algorithm), the rate of each flow will oscillate and never converge to any rate allocation policy. To show that this is indeed the case, suppose that Charny et al.'s marking technique can converge to the optimal GMM rate allocation. Then, the advertised rate φ_ℓ for this single link ℓ (Link12) should satisfy $\varphi_\ell \geq 0.4$ (otherwise, flow $s1$ will not be marked by the definition of Charny et al.'s algorithm). We call this the initial state; the algorithm will enter the following iterations:

- Since the PR for $s3$ is 0.5, $s3$ will increase its rate to φ_ℓ, which is greater than or equal to 0.4. Consequently, the sum of rates for all flows will exceed link capacity. Then, all flows will be unmarked and φ_ℓ is set to 0 by Charny et al.'s marking algorithm.
- Now, each flow transmits at its MR and the advertised rate φ_ℓ is recalculated to $\varphi_\ell = 0.3$ based on the GMM-bottleneck link rate definition in Definition 3.
- Next, flow $s2$ will increase its rate to 0.25 (its peak rate) and will be marked when its RM packet arrives at the link. The new value for φ_ℓ becomes $\varphi_\ell = 0.35$ based on Definition 3.
- Now, flow $s3$ will increase its rate to 0.35 and will be marked. The new φ_ℓ is recalculated to $\varphi_\ell = 0.4$.
- Next, flow $s1$ will also be marked when its RM packet arrives at the link. We have just returned to the exact same initial state where we started. The loop will continue; φ_ℓ will keep oscillating and will never converge.

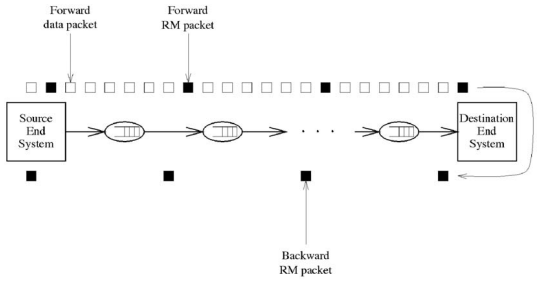

Fig. 5. A feedback-based flow control mechanism.

The difficulty here is that, under the generalized definitions of GMM-bottleneck link and GMM-bottleneck link rate, we need to consider the unique MR requirement from each flow. On the other hand, under Charny et al.'s flow marking algorithm, a flow will be considered to have converged to its expected rate allocation only if it is marked; consequently, upon convergence, all flows are expected to be marked. Clearly, such a marking technique will break here due to the new definition of GMM-bottleneck link rate. In particular, since such MR requirements are ordered in a nonlinear fashion, the rate allocation behaves like "water-filling" under the centralized algorithm, and GMM-bottleneck link rate could be smaller than the rate of some flows.

To overcome this difficulty, we must reexamine Charny et al.'s flow marking technique. We first offer a deep understanding of what minimal requirements are needed to make a flow marking algorithm converge to max-min. Under Charny et al.'s algorithm, the purpose of marking flows is to distinguish the sessions into two separate groups (i.e., those that have already converged and those that are currently undergoing an iterative convergence process). Since such a marking technique will not work for GMM rate allocation, we ask the following question: What is the most essential component in this technique that can bring a distributed algorithm to converge to max-min? Our investigation of this question led us to significantly depart from Charny et al.'s original technique.

We find that, in Charny et al.'s algorithm, it is overly restrictive to require marking of *all* sessions that traverse a bottleneck link. Although this technique makes it easier to comprehend and develop a correctness proof in the case of max-min, it severely limits the broader applicability of such a marking technique. In particular, for max-min, we find that a flow traversing its own max-min bottleneck link does not need to be marked at that link. Only flows bottlenecked *elsewhere* need to be marked. Although this finding appears counter-intuitive when we consider convergence issues, it becomes easy to understand if we notice the following: In order to calculate the max-min bottleneck link rate at a saturated link, we only need to identify flows that are bottlenecked elsewhere, rather than its own bottleneck link. Another way to look at this situation is that, at a node, all that we need is the information to distinguish between the set of marked flows (bottlenecked elsewhere) and the set of unmarked flows that are potentially bottlenecked at this particular node. In the case of GMM rate allocation, such generalization is essential to coping with the difficulties associated with GMM-bottleneck link rate definition, which in fact mandates that, at a node, only sessions that are GMM-bottlenecked elsewhere can be marked, while all other sessions must not be unmarked. Clearly, such a "minimum-effort" marking scheme is much

TABLE 4
Notation for Source Variables and RM Fields

Source Variable	**AR:** Allowed maximum rate of the source **IR:** Initial transmission rate of the source **MR:** Minimum rate requirement **PR:** Peak rate constraint
RM Packet Fields	**CR:** Current rate of the flow **MR:** Minimum rate requirement **ER:** Explicit rate

more flexible than Charny et al.'s original scheme. It turns out that this is the key to designing a distributed convergence algorithm for GMM rate allocation. It should be noted, though, that such flexibility brings in substantial complexity in keeping track of the state of each flow along a path, which in turn adds substantial difficulty in the convergence proof.

3.3 A Distributed Convergence Algorithm for GMM

Based on our revised flow marking technique, we present a distributed algorithm for GMM rate allocation. This distributed algorithm includes an algorithm for the end system (source and destination) and an algorithm for each switch along the path. We first specify the algorithm for the end system, where the source variables and RM fields are defined in Table 4.

Algorithm 3: (**End System Behavior**)
Source Behavior: The source starts to transmit at AR := IR, with IR $\geq$ MR. For every N_{RM} transmitted data packets, the source sends a forward RM(CR, MR, ER) packet with: CR := AR; MR := MR; ER := PR. Upon receipt of a backward RM(CR, MR, ER) packet from the destination, the AR at the source is adjusted to: AR := ER.
Destination Behavior: Upon receiving an RM packet, the destination returns it back toward the source.

The core component in the distributed convergence algorithm resides in the design of the switch algorithm. Basically, we need to calculate GMM-bottleneck link rate at each node (see Definition 3) so that we can properly place flows at a node into two sets: the set of flows GMM-bottlenecked elsewhere and the set of flows GMM-bottlenecked at this node. We assume a simple first-in-first-out (FIFO) scheduling discipline at each node.

In our algorithm, a switch maintains a table at each of its output ports (a FIFO queue) and keeps track of the state information for each traversing flow. More specifically, the per-flow table consists of a linked list of records, each of which is for a particular flow and contains several fields (see Table 5). In particular, r_ℓ^i is the rate of flow $i \in \mathcal{S}$ that is most recently recorded when an RM packet of flow i passes link ℓ. For each RM packet, the r_ℓ^i is independent of link ℓ along the path. But, for two different (even consecutive) RM packets for flow i along the same path, one RM packet arriving at link k and the other arriving at link ℓ, it is possible that the rate r_k^i and r_ℓ^i are different (due to rate adaptation at the source). For clarity, we use r_ℓ^i to indicate the rate from the most recent RM packet that has just traversed link ℓ. In Table 5, we also list several other parameters or variables that are maintained at a node, which will be used in our distributed algorithm.

We now describe the switch algorithm as follows: The initial conditions for each $\ell \in \mathcal{L}$ are set to: $\mathcal{S}_\ell := \emptyset$; $n_\ell := 0$; $\varphi_\ell := C_\ell$.

TABLE 5
State Information for Each Flow and Other Parameters Maintained at a Node

Per-flow State Variables	
r_ℓ^i	CR value of flow $i \in \mathcal{S}_\ell$
MR^i	MR value of flow $i \in \mathcal{S}_\ell$
b_ℓ^i	A bit used to indicate marking status of flow ℓ, $b_\ell^i := 1$ if flow i is marked, or 0 if unmarked.
Other Node Parameters	
$\mathcal{S}_\ell$	Set of flows traversing link ℓ
$\mathcal{M}_\ell$	Set of flows marked at link ℓ, i.e., $\mathcal{M}_\ell = \{i \mid i \in \mathcal{S}_\ell \text{ and } b_\ell^i = 1\}$
$\mathcal{U}_\ell$	Set of flows unmarked at link ℓ, i.e., $\mathcal{U}_\ell = \{i \mid i \in \mathcal{S}_\ell \text{ and } b_\ell^i = 0\}$, and $\mathcal{M}_\ell \cup \mathcal{U}_\ell = \mathcal{S}_\ell$
C_ℓ	Capacity of link ℓ
RC_ℓ	Remaining Capacity variable at link ℓ used for φ_ℓ calculation (Algorithm 5)
n_ℓ	Number of flows in $\mathcal{S}_\ell$, $\ell \in \mathcal{L}$, i.e., $n_\ell = \lvert\mathcal{S}_\ell\rvert$
φ_ℓ	Advertised rate at link ℓ, calculated according to Algorithm 5

Algorithm 4: (**Switch Behavior**)
Upon the receipt of a forward RM(CR, MR, ER) packet {
  if RM packet signals flow i's termination {
    $\mathcal{S}_\ell := \mathcal{S}_\ell - \{i\}$; $n_\ell := n_\ell - 1$;
    /* Update advertised rate φ_ℓ and flow marking status. */
    table_update();
    }
  if RM packet signals a new flow i's initiation {
    /* Insert a new record for this flow in the table (a linked list of records) such that the MR fields of the linked list of records are in increasing order,[5] i.e., $\mathrm{MR}[1] \leq \cdots \leq \mathrm{MR}[\mathrm{i}-1] \leq \mathrm{MR}[\mathrm{i}] \leq \mathrm{MR}[\mathrm{i}+1] \leq \cdots \leq \mathrm{MR}[|\mathcal{U}_\ell|]$. */
    $\mathcal{S}_\ell := \mathcal{S}_\ell \cup \{i\}$; $n_\ell := n_\ell + 1$;
    $b_\ell^i := 0$; $r_\ell^i := \mathrm{CR}$; $\mathrm{MR}^i := \mathrm{MR}$;
    /* Update advertised rate φ_ℓ and flow marking status. */
    table_update();
    }
  else /* RM packet belongs to an ongoing active flow i. */ {
    $r_\ell^i := \mathrm{CR}$;
    if ($r_\ell^i < \varphi_\ell$), then $b_\ell^i := 1$; /* Only mark a flow that is GMM-bottlenecked elsewhere. */
    /* Update advertised rate φ_ℓ and flow marking status. */
    table_update();
    }
  Forward RM(CR, MR, ER) toward its destination;
}
Upon the receipt of a backward RM(CR, MR, ER) packet from the destination of flow i {
  ER := $\max\{\min\{\mathrm{ER},\ \varphi_\ell\},\ \mathrm{MR}\}$;
  Forward RM(CR, MR, ER) toward its source;
}

5. Such a table creation scheme helps to eliminate sorting of MRs into increasing order for φ_ℓ calculation.

table_update()
{
rate_calculation_1: use Algorithm 5 to calculate advertised rate φ_ℓ^1;
Unmark (i.e., set $b_\ell^i = 0$) any flow $i \in \mathcal{S}_\ell$ with $r_\ell^i \geq \varphi_\ell^1$;
rate_calculation_2: use Algorithm 5 to calculate advertised rate φ_ℓ;
if $(\varphi_\ell < \varphi_\ell^1)$, then {
Unmark any flow $i \in \mathcal{S}_\ell$ with $r_\ell^i \geq \varphi_\ell$;
rate_calculation_3: use Algorithm 5 to calculate advertised rate φ_ℓ again;
}
}

Note that, in the table_update() subroutine, both φ_ℓ^1 and φ_ℓ follow the same φ_ℓ calculation in Algorithm 5. For the classical max-min policy, φ_ℓ calculated by rate_calculation_2 is always greater than or equal to φ_ℓ^1 and rate_calculation_3 is not needed [4]. But, for GMM rate allocation, φ_ℓ calculated by rate_calculation_2 can be less than φ_ℓ^1 and, therefore, a third round of unmarking and rate_calculation_3 is needed (see the proof of Proposition 1 for such a case). The following algorithm for φ_ℓ calculation is used in the table_update() subroutine in the above switch algorithm.

Algorithm 5: (φ_ℓ **Calculation**)
If $n_\ell = 0$, then $\varphi_\ell := C_\ell$;
Else if $n_\ell = |\mathcal{M}_\ell|$, then $\varphi_\ell := C_\ell - \sum_{i\in\mathcal{S}_\ell} r_\ell^i + \max_{i\in\mathcal{S}_\ell} r_\ell^i$;
Else /* i.e., $n_\ell \neq |\mathcal{M}_\ell|$. */ {
$RC_\ell := C_\ell - \sum_{i\in\mathcal{M}_\ell} r_\ell^i$;
if $(RC_\ell \leq \sum_{i\in\mathcal{U}_\ell} \mathrm{MR}^i)$, then $\varphi_\ell := 0$;
else /* i.e., $RC_\ell > \sum_{i\in\mathcal{U}_\ell} \mathrm{MR}^i$. */ {
/* Due to our table creation scheme (see Algorithm 4 for the case when a new flow joins the networks), the unmarked flows $s \in \mathcal{U}_\ell$ are already in increasing order of their MRs, i.e., $\mathrm{MR}[1] \leq \mathrm{MR}[2] \leq \cdots \leq \mathrm{MR}[|\mathcal{U}_\ell|]$. There is no need to perform sorting as in the centralized algorithm. */
$k := |\mathcal{U}_\ell|$; $\varphi_\ell := \frac{RC_\ell}{k}$;
while $(\varphi_\ell < \mathrm{MR}[k])$ {
$RC_\ell := RC_\ell - \mathrm{MR}[k]$;
$k := k - 1$; $\varphi_\ell := \frac{RC_\ell}{k}$;
}
}[6]
}

We now discuss the complexity of the switch algorithm. Processing complexity is dominated by the `table_update()` subroutine when processing a forward RM packet, which is $O(n_\ell)$. It is possible to reduce processing complexity by discretization on the range of the rate a flow can take, which is like a class-based rate allocation within which flows within the same class are allocated with the same rate. Another approach is to develop a heuristic algorithm [13] that removes the per-flow state information from the network's switches. These measures will help improve the scalability of the switch algorithm, but at the expense of rate granularity or convergence guarantee.

6. The combined steps in the bracket for "else" are equivalent to finding GMM-bottleneck link rate φ_ℓ for the set of unmarked flows $\mathcal{U}_\ell$ such that $\varphi_\ell \cdot \sum_{i\in\mathcal{U}_\ell} 1^+\{\mathrm{MR}^i \leq \varphi_\ell\} + \sum_{i\in\mathcal{U}_\ell} \mathrm{MR}^i \cdot 1^+\{\mathrm{MR}^i > \varphi_\ell\} = RC_\ell$. In the special case when $\mathrm{MR}^i = 0$ for every $i \in \mathcal{U}_\ell, \varphi_\ell = \frac{RC_\ell}{|\mathcal{U}_\ell|}$, i.e., the max-min rate allocation.

We observe that, by the operations of Algorithms 3 and 4, we have the following fact for the AR at the source and the CR field in the RM packet:

Fact 1. For every flow $s \in \mathcal{S}$, the AR at the source and the CR field in the RM packet are GMM-feasible, i.e., $\mathrm{MR}^s \leq \mathrm{AR}^s \leq \mathrm{PR}^s$ and $\mathrm{MR}^s \leq \mathrm{CR}^s \leq \mathrm{PR}^s$.

4 Convergence Proof of Distributed Algorithm

The convergence proof of our distributed algorithm follows a similar induction approach to that in [4]. Our main contribution here is to address how to handle the more complex and difficult problem associated with the new flow marking technique and the new definition of GMM-bottleneck link rate. The key notion used in the convergence proof is the state of *GMM-marking-consistent* for flows at a link, which is defined as follows.

Definition 4 (GMM-Marking-Consistent). *Let $\mathcal{M}_\ell$ be the set of marked flows at link $\ell \in \mathcal{L}$. We say that the marking of flows at link $\ell \in \mathcal{L}$ is in the state of GMM-marking-consistent if $r_\ell^i < \varphi_\ell$ for every flow $i \in \mathcal{M}_\ell$.*

The following proposition shows the table marking property at an output port after the switch algorithm is performed for a traversing RM packet:

Proposition 1. *After the switch algorithm is performed for an RM packet traversing a link, the marking of flows at this link is GMM-marking-consistent.*

Proof. Let $\mathcal{M}_\ell$ and $\mathcal{U}_\ell$ be the set of marked and unmarked flows at link ℓ just before rate_calculation_1 is performed, respectively; φ_ℓ^1 be the result for the advertised rate by rate_calculation_1 in function table_update(); $\mathcal{Z}_\ell \subseteq \mathcal{M}_\ell$ be the set of flows with $r_\ell^i \geq \varphi_\ell^1$, $i \in \mathcal{Z}_\ell$, and, therefore, are unmarked by the unmarking operation after rate_calculation_1 in function table_update(); φ_ℓ be the result for advertised rate by rate_calculation_2 in function table_update().

Case 1: If not all flows in $\mathcal{S}_\ell$ are marked before rate_calculation_1, i.e., $\mathcal{M}_\ell \neq \mathcal{S}_\ell$, then we have the following two scenarios.

Subcase 1-A: During rate_calculation_1, if $C_\ell - \sum_{i\in\mathcal{M}_\ell} r_\ell^i < \sum_{i\in\mathcal{U}_\ell} \mathrm{MR}^i$, then $\varphi_\ell^1 = 0$ (see Algorithm 5). Thus, every flow $i \in \mathcal{M}_\ell$ will be unmarked by the unmarking operation and φ_ℓ calculated by rate_calculation_2 satisfies

$$\varphi_\ell \cdot \sum_{i\in\mathcal{S}_\ell} 1^+\{\mathrm{MR}^i \leq \varphi_\ell\} + \sum_{i\in\mathcal{S}_\ell} \mathrm{MR}^i \cdot 1^+\{\mathrm{MR}^i > \varphi_\ell\} = C_\ell$$

and $C_\ell > \sum_{i\in\mathcal{S}_\ell} \mathrm{MR}^i$ by Assumption 1. Therefore, $\varphi_\ell \geq \varphi_\ell^1 = 0$ and GMM-marking-consistent property trivially holds.

Subcase 1-B: During rate_calculation_1 for φ_ℓ^1, if

$$C_\ell - \sum_{i\in\mathcal{M}_\ell} r_\ell^i \geq \sum_{i\in\mathcal{U}_\ell} \mathrm{MR}^i, \tag{3}$$

then φ_ℓ^1 satisfies

$$\begin{aligned} &\varphi_\ell^1 \cdot \sum_{i\in\mathcal{U}_\ell} 1^+\{\mathrm{MR}^i \leq \varphi_\ell^1\} + \sum_{i\in\mathcal{U}_\ell} \mathrm{MR}^i \cdot 1^+\{\mathrm{MR}^i > \varphi_\ell^1\} \\ &= C_\ell - \sum_{i\in\mathcal{M}_\ell} r_\ell^i. \end{aligned} \tag{4}$$

After unmarking $\mathcal{Z}_\ell \subseteq \mathcal{M}_\ell$ with $r_\ell^i \geq \varphi_\ell^1,\ i \in \mathcal{Z}_\ell$, in function table_update(), we have

$$C_\ell - \sum_{i\in(\mathcal{M}_\ell - \mathcal{Z}_\ell)} r_\ell^i = C_\ell - \sum_{i\in\mathcal{M}_\ell} r_\ell^i + \sum_{i\in\mathcal{Z}_\ell} r_\ell^i \geq \sum_{i\in\mathcal{U}_\ell} \mathrm{MR}^i + \sum_{i\in\mathcal{Z}_\ell} \mathrm{MR}^i = \sum_{i\in(\mathcal{U}_\ell\cup\mathcal{Z}_\ell)} \mathrm{MR}^i.$$

The above inequality holds by (3) and by Fact 1, $\sum_{i\in\mathcal{Z}_\ell} r_\ell^i \geq \sum_{i\in\mathcal{Z}_\ell} \mathrm{MR}^i$. In rate_calculation_2 for φ_ℓ, we have

$$\varphi_\ell \cdot \sum_{i\in(\mathcal{U}_\ell\cup\mathcal{Z}_\ell)} 1^+\{\mathrm{MR}^i \leq \varphi_\ell\} + \sum_{i\in(\mathcal{U}_\ell\cup\mathcal{Z}_\ell)} \mathrm{MR}^i \cdot 1^+\{\mathrm{MR}^i > \varphi_\ell\} = C_\ell - \sum_{i\in(\mathcal{M}_\ell-\mathcal{Z}_\ell)} r_\ell^i. \tag{5}$$

But, by (4),

$$\begin{aligned} C_\ell - \sum_{i\in(\mathcal{M}_\ell-\mathcal{Z}_\ell)} r_\ell^i &= (C_\ell - \sum_{i\in\mathcal{M}_\ell} r_\ell^i) + \sum_{i\in\mathcal{Z}_\ell} r_\ell^i \\ &= \varphi_\ell^1 \cdot \sum_{i\in\mathcal{U}_\ell} 1^+\{\mathrm{MR}^i \leq \varphi_\ell^1\} \\ &\quad + \sum_{i\in\mathcal{U}_\ell} \mathrm{MR}^i \cdot 1^+\{\mathrm{MR}^i > \varphi_\ell^1\} + \sum_{i\in\mathcal{Z}_\ell} r_\ell^i. \end{aligned} \tag{6}$$

Since $r_\ell^i \geq \varphi_\ell^1$ and $r_\ell^i \geq \mathrm{MR}^i$ for $i \in \mathcal{Z}_\ell$, to have (5) equal to (6), we must have $\varphi_\ell \geq \varphi_\ell^1$. That is, φ_ℓ calculated by rate_calculation_2 is greater than or equal to φ_ℓ^1 by rate_calculation_1. Since $r_\ell^i < \varphi_\ell^1$ for $i \in (\mathcal{M}_\ell - \mathcal{Z}_\ell)$ and $\varphi_\ell^1 \leq \varphi_\ell$, the marking of these flows continues to be GMM-marking-consistent after rate_calculation_2 is performed.

Case 2: If all flows in $\mathcal{S}_\ell$ are marked before rate_calculation_1, i.e., $\mathcal{M}_\ell = \mathcal{S}_\ell$, we have two scenarios. Let the RM packet for which the switch algorithm is performed belong to flow $s \in \mathcal{S}$.

Subcase 2-A: If flow s was not marked before the RM packet's arrival at link ℓ and is marked because of this RM packet's arrival with $r_\ell^s = \mathrm{CR} < \varphi_\ell$, where φ_ℓ was calculated by the switch algorithm for the previous traversing RM packet and satisfies

$$\varphi_\ell = C_\ell - \sum_{i\in\mathcal{S}_\ell, i\neq s} r_\ell^i.$$

After marking $b_\ell^s = 1$, we have

$$C_\ell - \sum_{i\in\mathcal{S}_\ell} r_\ell^i > 0. \tag{7}$$

During rate_calculation_1 for φ_ℓ^1:

$$\varphi_\ell^1 = C_\ell - \sum_{i\in\mathcal{S}_\ell} r_\ell^i + \max_{i\in\mathcal{S}_\ell} r_\ell^i.$$

With (7), we have

$$\varphi_\ell^1 > \max_{i\in\mathcal{S}_\ell} r_\ell^i \geq r_\ell^p$$

for every flow $p \in \mathcal{S}_\ell$. So, all flows in $\mathcal{S}_\ell$ will remain marked after the unmarking operation. Therefore, φ_ℓ, as calculated by rate_calculation_2, will be the same as φ_ℓ^1 and the marking of all flows is GMM-marking-consistent.

Subcase 2-B: If flow s was already marked before this RM packet arriving at link ℓ, the arrival of this RM packet will not change the advertised rate φ_ℓ if the CR in this RM packet is the same as r_ℓ^s in the current table at the switch. On the other hand, if the new CR is different from the recorded CR for this flow in the table, r_ℓ^s will be updated with this new CR value. During rate_calculation_1 for φ_ℓ^1, we have

$$\varphi_\ell^1 = C_\ell - \sum_{i\in\mathcal{S}_\ell} r_\ell^i + \max_{i\in\mathcal{S}_\ell} r_\ell^i.$$

Again, let $\mathcal{Z}_\ell \subseteq \mathcal{M}_\ell$ denote the set of flows with $r_\ell^i \geq \varphi_\ell^1,\ i \in \mathcal{Z}_\ell$ and, therefore, are unmarked by the unmarking operation after rate_calculation_1 in function table_update().

- If $\mathcal{Z}_\ell = \emptyset$, i.e., every flow is marked, then φ_ℓ calculated by rate_calculation_2 will be the same as φ_ℓ^1 and all flows will remain GMM-marking-consistent.
- If $\mathcal{Z}_\ell \neq \emptyset$, then the set of flows in $\mathcal{Z}_\ell$ will be unmarked since

$$r_\ell^i \geq \varphi_\ell^1,\ \ i \in \mathcal{Z}_\ell. \tag{8}$$

This is the only situation where φ_ℓ as calculated by rate_calculation_2 may be less than φ_ℓ^1. In this case (i.e., $\varphi_\ell < \varphi_\ell^1$), we perform a third around of unmarking and φ_ℓ calculation (rate_calculation_3). It is clear that the combined steps of rate_calculation_2, unmarking, and rate_calculation_3 here are equivalent to Case 1. Thus, φ_ℓ calculated by rate_calculation_3 is greater than or equal to φ_ℓ calculated by rate_calculation_2 and the marking of flows is GMM-marking-consistent. □

Proposition 1 is the foundation for the rest of the convergence proof. The following lemma gives a lower bound for φ_ℓ at link $\ell \in \mathcal{L}$.

Lemma 1. *Assume we have a set of $\mathcal{S}$ flows in the network at time $t = 0$ and the rate of each flow is controlled by the distributed algorithm at end systems and switches. Denote α_ℓ as*

$$\alpha_\ell \cdot \sum_{i\in\mathcal{S}_\ell} 1^+\{\mathrm{MR}^i \leq \alpha_\ell\} + \sum_{i\in\mathcal{S}_\ell} \mathrm{MR}^i \cdot 1^+\{\mathrm{MR}^i > \alpha_\ell\} = C_\ell$$

for every $\ell \in \mathcal{L}$. Then, there exists some time t_0 such that, for $t \geq t_0$, $\varphi_\ell \geq \alpha_\ell$ for every $\ell \in \mathcal{L}$.

Proof. Let time t_0 be the time immediately after the switch algorithm is performed for an RM packet at link ℓ and $\mathcal{M}_\ell$ and $\mathcal{U}_\ell$ denote the set of marked and unmarked flows at link ℓ, respectively. By Proposition 1, the marking of flows at link ℓ is GMM-marking-consistent. That is, every marked flow $i \in \mathcal{M}_\ell$ satisfies $r_\ell^i < \varphi_\ell$.

Case 1: If some flows in $\mathcal{S}_\ell$ are not marked, i.e., $\mathcal{M}_\ell \neq \mathcal{S}_\ell$, then

$$\varphi_\ell \cdot \sum_{i\in\mathcal{U}_\ell} 1^+\{\mathrm{MR}^i \leq \varphi_\ell\} + \sum_{i\in\mathcal{U}_\ell} \mathrm{MR}^i \cdot 1^+\{\mathrm{MR}^i > \varphi_\ell\} = C_\ell - \sum_{i\in\mathcal{M}_\ell} r_\ell^i.$$

Therefore,

$$C_\ell = \sum_{i\in\mathcal{M}_\ell} r_\ell^i + \varphi_\ell \cdot \sum_{i\in\mathcal{U}_\ell} 1^+\{\mathrm{MR}^i \le \varphi_\ell\} + \sum_{i\in\mathcal{U}_\ell} \mathrm{MR}^i \cdot 1^+\{\mathrm{MR}^i > \varphi_\ell\}.$$

On the other hand,

$$C_\ell = \alpha_\ell \cdot \sum_{i\in\mathcal{S}_\ell} 1^+\{\mathrm{MR}^i \le \alpha_\ell\} + \sum_{i\in\mathcal{S}_\ell} \mathrm{MR}^i \cdot 1^+\{\mathrm{MR}^i > \alpha_\ell\}$$

by the definition of α_ℓ. Since $r_\ell^i < \varphi_\ell$ for $i \in \mathcal{M}_\ell$, we must have $\varphi_\ell \ge \alpha_\ell$ (the equality holds only when $\mathcal{M}_\ell = \emptyset$).

Case 2: If all flows in $\mathcal{S}_\ell$ are marked, i.e., $\mathcal{M}_\ell = \mathcal{S}_\ell$, there are two possible scenarios.

Subcase 2-A: Suppose $\max_{i\in\mathcal{S}_\ell} r_\ell^i \ge \alpha_\ell$. Since $\varphi_\ell > \max_{i\in\mathcal{S}_\ell} r_\ell^i$, we have $\varphi_\ell \ge \alpha_\ell$.

Subcase 2-B: If $\max_{i\in\mathcal{S}_\ell} r_\ell^i < \alpha_\ell$, then for every flow $i \in \mathcal{S}_\ell$, $r_\ell^i < \alpha_\ell$. Let flow $p \in \mathcal{S}$ be the flow such that $r_\ell^p = \max_{i\in\mathcal{S}_\ell} r_\ell^i$. Then,

$$\varphi_\ell = C_\ell - \sum_{i\in\mathcal{S}_\ell} r_\ell^i + \max_{i\in\mathcal{S}_\ell} r_\ell^i = C_\ell - \sum_{i\in\mathcal{S}_\ell,\, i\ne p} r_\ell^i$$
$$= \left(\alpha_\ell \cdot \sum_{i\in\mathcal{S}_\ell} 1^+\{\mathrm{MR}^i \le \alpha_\ell\} + \sum_{i\in\mathcal{S}_\ell} \mathrm{MR}^i \cdot 1^+\{\mathrm{MR}^i > \alpha_\ell\}\right) - \sum_{i\in\mathcal{S}_\ell,\, i\ne p} r_\ell^i \ge \alpha_\ell.$$

The last inequality holds because

$$\alpha_\ell \cdot \sum_{i\in\mathcal{S}_\ell} 1^+\{\mathrm{MR}^i \le \alpha_\ell\} + \sum_{i\in\mathcal{S}_\ell} \mathrm{MR}^i \cdot 1^+\{\mathrm{MR}^i > \alpha_\ell\} \ge \alpha_\ell |\mathcal{S}_\ell|$$

and $\sum_{i\in\mathcal{S}_\ell,\, i\ne p} r_\ell^i \le \alpha_\ell(|\mathcal{S}_\ell| - 1)$. □

Intuitively, α_ℓ represents GMM-bottleneck link rate when the network had only a single shared link ℓ (i.e., the peer-to-peer network in Fig. 1). Lemma 1 states that, in a general multinode network, the advertised rate φ_ℓ is greater than or equal to α_ℓ at link $\ell \in \mathcal{L}$.

Denote K the total number of iterations needed to execute the centralized algorithm for GMM rate allocation (Algorithm 2). As we have shown in the correctness proof for Algorithm 2, $K \le (2|\mathcal{S}| - 1)$, where $|\mathcal{S}|$ is the total number of flows in the network. During each iteration of Algorithm 2, there are three types of events as follows (also see Algorithm 1):

1. The rate of the flow with the smallest rate reaches the rate of the flow with the second smallest rate.
2. The rate of the flow with the smallest rate reaches its PR.
3. Some link saturates.

In the centralized algorithm, in the worst-case, a type 1 event can take at most $(|\mathcal{S}| - 1)$ iterations, in which case, each flow has a unique MR and $(|\mathcal{S}| - 1)$ iterations will bring the rates of all flows to the same rate of $\max_{p\in\mathcal{S}} \mathrm{MR}^p$. Note that there is no flow being removed during type 1 events and the rate allocation for each flow is temporary. On the other hand, types 2 and 3 iterations give a permanent rate assignment to some flow and such flow is removed out of future iterations. In the following, we will focus only on types 2 and 3 iterations and index such iterations as $1, \ldots, N$, where N denotes the total number of type 2 and 3 iterations in executing Algorithm 2. We have shown in the correctness proof of Algorithm 2 that $N \le |\mathcal{S}|$.

Denote Ψ_n the set of flows being removed at the end of the nth iteration in the centralized algorithm (Algorithm 2), where $n = 1, \ldots, N$ is the newly indexed iteration when we consider *only* types 2 and 3 iterations of Algorithm 2. Flows in Ψ_n have reached their GMM rate allocation. Let $\mathcal{L}_n$, $1 \le n \le N$, be the set of links traversed by flows in Ψ_n. Note that $\Psi_1, \Psi_2, \ldots, \Psi_N$ are mutually exclusive and the sum of $\Psi_1, \Psi_2, \ldots, \Psi_N$ is $\mathcal{S}$ while $\mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_N$ may be mutually inclusive. That is, there could be links belonging to both $\mathcal{L}_n$ and $\mathcal{L}_{n+1}$. This happens when some flow in Ψ_n reaches its PR before saturating link $\ell \in \mathcal{L}_n$ at the nth iteration.

Let σ_n, $1 \le n \le N$, be defined as follows:

$$\sigma_n = \max_{s\in\Psi_n} r^s \cdot 1^+\{r^s > \mathrm{MR}^s\},$$

where r^s is GMM rate allocation for flow s by Algorithm 2. It is clear that σ_n, $1 \le n \le N$, are GMM-bottleneck link rates during the centralized algorithm. By the operation of Algorithm 2, we have the following property for σ_n, $1 \le n \le N$, $\sigma_1 < \sigma_2 < \ldots < \sigma_N$. The following lemma states the inequality between the advertised rate φ_ℓ and σ_1 on every link $\ell \in \mathcal{L}$ in the network.

Lemma 2. *Let t_0 and α_ℓ be defined as in Lemma 1.*

1. *If $\sigma_1 = \alpha_\ell \le \mathrm{PR}^s$ for $s \in \Psi_1$, i.e., flows $s \in \Psi_1$ reach the GMM-bottleneck link rate before their PRs in the centralized algorithm, then, for any $t > t_0$, $\varphi_\ell \ge \sigma_1$ for every $\ell \in \mathcal{L}_1$ and $\varphi_\ell > \sigma_1$ for every $\ell \in (\mathcal{L} - \mathcal{L}_1)$ in the distributed algorithm.*
2. *If $\sigma_1 = \mathrm{PR}^s < \alpha_\ell$ for $s \in \Psi_1$, i.e., flows $s \in \Psi_1$ reach their PRs before GMM-bottleneck link rate in the centralized algorithm, then, for any $t > t_0$, $\varphi_\ell > \sigma_1$ for every $\ell \in \mathcal{L}$ in the distributed algorithm.*

The proof of Lemma 2 is given in the Appendix. The following lemma shows that the rate allocation for flow $s \in \Psi_1$ (in the centralized algorithm) will eventually converge to GMM rate allocation in the distributed algorithm.

Lemma 3 (Base Case). *There exists a $T_1 \ge 0$ such that:*

1. *If $\sigma_1 = \alpha_\ell \le \mathrm{PR}^s$ for $s \in \Psi_1$, i.e., flows $s \in \Psi_1$ reach the GMM-bottleneck link rate before their PRs in the centralized algorithm, then, for $t \ge T_1$, the following statements hold for the distributed algorithm.*
   a. *$\varphi_\ell = \sigma_1$ for every link $\ell \in \mathcal{L}_1$.*
   b. *The ER field of every returning RM packet of flow $i \in \Psi_1$ satisfies $\mathrm{ER} = \max\{\sigma_1,\ \mathrm{MR}\}$.*
   c. *The AR at source for every flow $i \in \Psi_1$ satisfies $\mathrm{AR} = \max\{\sigma_1,\ \mathrm{MR}\}$.*
   d. *$r_\ell^i = \max\{\sigma_1,\ \mathrm{MR}\}$ for every flow $i \in \Psi_1$ and every link ℓ traversed by flow $i \in \Psi_1$; $b_\ell^i = 1$ (marked) for every flow with $r_\ell^i = \sigma_1$, $i \in \Psi_1$ and every traversing link ℓ,* except *at its GMM-bottleneck link $\ell \in \mathcal{L}_1$ where $b_\ell^i = 0$ (unmarked).*
   e. *The ER field of every returning RM packet of flow $j \in (\mathcal{S} - \Psi_1)$ satisfies $\mathrm{ER} > \sigma_1$.*
   f. *The AR at source for every flow $j \in (\mathcal{S} - \Psi_1)$ satisfies $\mathrm{AR} > \sigma_1$.*
   g. *The recorded CR of flow $j \in (\mathcal{S} - \Psi_1)$ satisfies $r_\ell^j > \sigma_1$ at every link ℓ traversed by flow j.*
2. *If $\sigma_1 = \mathrm{PR}^s < \alpha_\ell$ for $s \in \Psi_1$, i.e., flows $s \in \Psi_1$ reach their PRs before the GMM-bottleneck link rate in the*

centralized algorithm, then, for $t \geq T_1$, the following statements hold for the distributed algorithm:

a. *$\varphi_\ell > \sigma_1$ for every link $\ell \in \mathcal{L}_1$.*
b. *The ER field of every returning RM packet of flow $i \in \Psi_1$ satisfies $\mathrm{ER} = \mathrm{PR}^i$.*
c. *The AR at source for every flow $i \in \Psi_1$ satisfies $\mathrm{AR} = \mathrm{PR}^i$.*
d. *$b_\ell^i = 1$ (marked), $r_\ell^i = \mathrm{PR}^i$ for every flow $i \in \Psi_1$ and every link ℓ traversed by flow $i \in \Psi_1$.*
e. *—g. Same as statements 1e to 1g, respectively.*

Furthermore, it takes at most $2.5D$ for flows $s \in \Psi_1$ to converge to GMM rate allocation in the distributed algorithm, where D denotes the maximum round-trip time among all flows.

The proof of Lemma 3 is given in the Appendix. Note that, under Lemma 3, once flow $p \in \Psi_1$ has reached its optimal rate of $\max\{\sigma_1,\ \mathrm{MR}^p\}$ (in Case 1) or PR^p (in Case 2), its rate will *never* change and the marking for such flow has the following property:

1. If $r_\ell^p = \mathrm{MR}^p$ (Case 1), then flow p is not marked at its GMM-bottleneck link, but may be marked at other links it traverses.
2. If $r_\ell^p = \sigma_1$ (Case 1), then flow p is marked at all of its traversing links but not at its GMM-bottleneck link.
3. If $r_\ell^p = \mathrm{PR}^p$ (Case 2), then flow p is marked at every link it traverses. The result of Lemma 3 will now be used as the base case for induction on the index n of Ψ_n, $1 \leq n \leq N$.

Lemma 4 (Induction). *Suppose that, for some $1 \leq n \leq N-1$, there exists a $T_n \geq 0$ such that:*

1. *If $\sigma_j < \mathrm{PR}^s$ for $s \in \Psi_j$, $1 \leq j \leq n$, i.e., flows $s \in \Psi_j$ reach the GMM-bottleneck link rate before their PRs in the centralized algorithm and, for $t \geq T_n$, the following statements hold for the distributed algorithm.*

   a. *$\varphi_\ell = \sigma_j$ for every link $\ell \in \mathcal{L}_j$.*
   b. *The ER field of a returning RM packet of flow $p \in \Psi_j$ satisfies $\mathrm{ER} = \max\{\sigma_j,\ \mathrm{MR}^p\}$.*
   c. *The AR at source for every flow $p \in \Psi_j$ satisfies $\mathrm{AR} = \max\{\sigma_j,\ \mathrm{MR}^p\}$.*
   d. *$r_\ell^p = \max\{\sigma_j,\ \mathrm{MR}^p\}$ for every flow $p \in \Psi_j$ and every link ℓ traversed by flow $p \in \Psi_j$; $b_\ell^p = 1$ (marked) for every flow with $r_\ell^p = \sigma_j$, $p \in \Psi_j$ and every traversing link ℓ, except at its GMM-bottleneck link $\ell \in \mathcal{L}_j$ where $b_\ell^p = 0$ (unmarked).*
   e. *The ER field of every returning RM packet of flow $p \in (\mathcal{S} - (\Psi_1 \cup \cdots \cup \Psi_n))$ satisfies $\mathrm{ER} > \sigma_n$.*
   f. *The AR at source for every flow $p \in (\mathcal{S} - (\Psi_1 \cup \cdots \cup \Psi_n))$ satisfies $\mathrm{AR} > \sigma_n$.*
   g. *The recorded CR of flow $p \in (\mathcal{S} - (\Psi_1 \cup \cdots \cup \Psi_n))$ satisfies $r_\ell^p > \sigma_n$ at every link ℓ traversed by flow p.*
2. *If $\sigma_j = \mathrm{PR}^s$ for $s \in \Psi_j$, $1 \leq j \leq n$, i.e., flows $s \in \Psi_j$ reach their PRs before the GMM-bottleneck link rate in the centralized algorithm, and for $t \geq T_n$, the following statements hold in the distributed algorithm:*

   a. *$\varphi_\ell > \sigma_j$ for every link $\ell \in \mathcal{L}_j$.*
   b. *The ER field of every returning RM packet of flow $p \in \Psi_j$ satisfies $\mathrm{ER} = \mathrm{PR}^p$.*
   c. *The AR at source for every flow $p \in \Psi_j$ satisfies $\mathrm{AR} = \mathrm{PR}^p$.*
   d. *$b_\ell^p = 1$ (marked), $r_\ell^p = \mathrm{PR}^p$ for every flow $p \in \Psi_j$ and every link ℓ traversed by flow $p \in \Psi_j$.*
   e. *—g. Same as statements 1e to 1g, respectively.*

   Then, there exists a $T_{n+1} \geq 0$ such that, for $t \geq T_{n+1}$, all statements in 1 and 2 hold for $n+1$.

The proof of Lemma 4 is given in the Appendix. It should be clear by now that the convergence and marking/unmarking of higher level rates of flows in the distributed algorithm depend on the convergence and marking/unmarking of lower level rates of flows, which is similar to that in the centralized algorithm. The following theorem follows from Lemmas 3 and 4 and is the main result of this section.

Theorem 2 (Convergence Theorem). *For a given number of active flows in the network, the rate allocation for each flow by the distributed algorithm converges to GMM rate allocation.*

Corollary 1 (Time Bound for Convergence). *Let K be the total number of iterations needed to execute the centralized algorithm for GMM rate allocation in Algorithm 2 and denote D the maximum round-trip time among all flows. Then, an upper bound for the convergence time to GMM rate allocation by our distributed algorithm for a given number of active flows in the network is $2.5KD$.*

This corollary follows from the proofs of Lemmas 3 and 4. It is worthwhile to point out that this upper bound for the convergence time is a loose one. In practice, the actual convergence time of our distributed algorithm is expected to be much shorter because:

1. The actual RTT for most flows is smaller than D, which is the *maximum* RTT among *all* flows.
2. Since the ER setting in our switch algorithm (Algorithm 4) is performed on backward RM cells (rather than forward RM cells), the effective control loop for a flow is, therefore, between the source and the particular switch, rather than the full source-destination round trip used in Corollary 1.

The following flow marking property also follows from the proofs of Lemmas 3 and 4.

Corollary 2 (Flow Marking Property). *Upon the convergence of the distributed algorithm, a flow $s \in \mathcal{S}$ is marked into one of the following states:*

1. *If $r^s = \mathrm{MR}^s$, then flow s is not marked at its GMM-bottleneck link (but may be marked at other links it traverses).*
2. *If $r^s = \mathrm{PR}^s$, then flow s is marked at every link it traverses.*
3. *Otherwise, i.e., flow s has a rate allocation equal to some GMM-bottleneck link rate and $\mathrm{MR}^s < r^s < \mathrm{PR}^s$, then flow s is marked at every link it traverses except its GMM-bottleneck link.*

It is worth pointing out that the distributed algorithm is robust to occasional packet loss. Such robustness is mainly due to the periodic availability of RM packets sent by the source among the data packets.

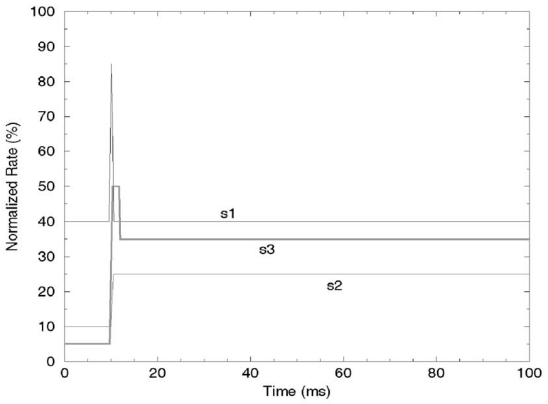

Fig. 6. The rates of all flows in the peer-to-peer network.

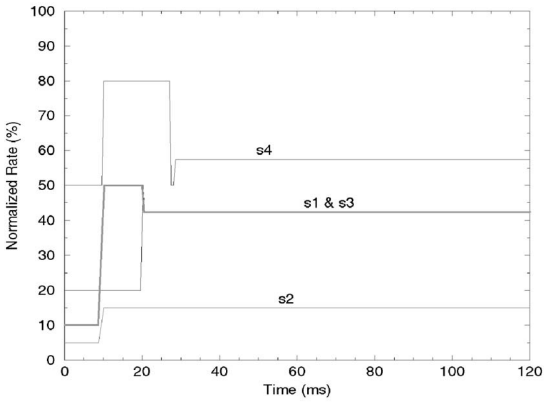

Fig. 7. The rates of all flows for the three-node network configuration.

5 Simulation Results

In this section, we implement our switch algorithm on our network simulator [7] and perform simulations on various network configurations. The network configurations that we use include the peer-to-peer and three-node network configurations shown in Figs. 1 and 3, respectively. In addition, we also use a chain network (Fig. 8) and a mesh network (Fig. 10).

All switches in the simulations are assumed to have output port buffering with internal switching capacity greater than or equal to the aggregate rates of their input ports (i.e., nonblocking switches). Each output port buffer of a switch employs the simple FIFO queuing discipline and is shared by all flows going through that port. We set the link capacity at 150 Mb/s. For stability, we set the target link utilization at 0.95. That is, we set $C_\ell = 0.95 \times 150\ \mathrm{Mb/s} = 142.5\ \mathrm{Mb/s}$ at every link $\ell \in \mathcal{L}$ for the ER calculation. By setting a target link utilization strictly less than 1, we ensure that, eventually, the potential buffer build-up during transient period will be drained. The packet transfer delay within a switch is assumed to be 4 μs (not including queuing delay at an output port). The distance from an end system (source or destination) to the switch is 1 km; the link distance between the switches is 1,000 km (corresponding to a wide area network). We assume that the propagation delay is 5 μs per km. At each source, we set its initial rate (IR) to the MR of the flow (or any small rate when MR is zero) and $N_{\mathrm{RM}} = 32$.

5.1 Peer-to-Peer Network

For this network (Fig. 1), the output port link of SW1 (Link12) is the only potential GMM-bottleneck link for all flows. Under a normalized unit link capacity, the minimum rate requirement, peak rate constraint, and GMM rate allocation of each flow are listed in Table 1.

Fig. 6 shows the AR at source for flows $s1$, $s2$, and $s3$, respectively. The rates shown in the figure are normalized with respect to the capacity C_ℓ (142.5 Mb/s) for easy comparison with those values obtained with our centralized algorithm under unit link capacity in Table 1. Each flow starts with its MR. The first RM packet for each flow returns to the source after one round trip time (RTT) or 10 ms. After some iterations, we see that the rate of each flow quickly converges to its respective GMM rate allocation as listed in Table 1.

During the course of distributed iterations, the AR of each flow in Fig. 6 maintains GMM-feasibility, i.e., $\mathrm{MR} \leq \mathrm{AR} \leq \mathrm{PR}$. Also shown in Fig. 6 is that the convergence time of our distributed algorithm is much smaller than the upper bound given in Corollary 1. In this case, the RTT is 10 ms and it takes less than 15 ms for our distributed algorithm to converge.

5.2 Three-Node Network

For this configuration (Fig. 3), the output port links of SW1 (Link12) and SW2 (Link23) are potential GMM-bottleneck links. Table 2 lists the MR requirement, PR constraint, and GMM rate allocation (obtained through the centralized algorithm) for each flow under unit link capacity. Fig. 7 shows the normalized AR of each flow under our distributed algorithm. We find that the AR of each flow converges to its GMM rate listed in Table 2. Here, the maximum RTT (D) among all flows is 20 ms ($s1$), and it takes our distributed algorithm less than 30 ms to converge to GMM rate allocation, which is much smaller than the upper bound given in Corollary 1.

5.3 Chain Network

This is one of the benchmark network configurations used to test feedback-based flow control algorithms; it is also referred to as a generic fairness configuration [3]. The specific topology that we use is shown in Fig. 8, where there are five switches interconnected in a chain with six paths traversing these switches and sharing link capacity. Table 6 lists the MR and PR constraints for each flow, as well as GMM rate allocation (obtained from the centralized algorithm) for each flow under unit link capacity.

Fig. 9 shows the normalized AR of each flow under our distributed algorithm. Again, the rate of each flow converges to its GMM rate allocation listed in Table 6. Here, the maximum RTT (D) among all flows is 30 ms ($s1$ and $s2$); it takes less than 2 RTT (60 ms) for our distributed algorithm to converge.

5.4 Mesh Network

In this last set of simulations, we implement our distributed algorithms on a more complex mesh network topology, as

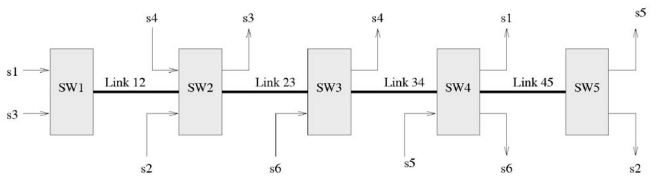

Fig. 8. A chain network configuration.

TABLE 6
MR Requirement, PR Constraint, and GMM Rate Allocation of Each Flow for the Chain Network Configuration

Flow	MR	PR	GMM Rate Allocation
$s1$	0.05	0.50	0.35
$s2$	0.10	0.25	0.25
$s3$	0.15	1.00	0.65
$s4$	0.05	0.15	0.15
$s5$	0.35	1.00	0.75
$s6$	0.40	0.60	0.40

shown in Fig. 10. In this case, we have a total of 12 flows traversing 12 paths that interact with each other along the various links. Table 7 lists flow paths, MR and PR constraints for each flow, and GMM rate allocation (obtained from the centralized algorithm) for each flow under unit link capacity.

Fig. 11 shows the normalized AR of each flow under our distributed algorithm. Again, the rate of each flow converges to its GMM rate allocation listed in Table 6. In this case, the maximum RTT (D) among all flows is 40 ms ($s11$ and $s12$) and it takes less than 60 ms ($<$ 2RTT) for our distributed algorithm to converge.

6 Conclusions

This paper investigated the fundamental problem of bandwidth allocation among flows in a packet network. We examined the classical max-min rate allocation and presented theory to generalize it with minimum rate and peak rate constraints. We designed a feedback-based distributed algorithm that achieves GMM rate allocation through asynchronous iterations. Our design offered a new perspective on flow marking technique and advanced the state-of-the-art flow marking scheme proposed by other researchers. We provided a proof of our distributed algorithm's convergence and used simulation results to demonstrate its fast convergence property.

Appendix 1

Proof of Theorem 1. To show the *"only if"* part, we suppose that the GMM-feasible rate vector $r = \{r_s \mid s \in \mathcal{S}\}$ is GMM and assume that, on the contrary, there exists some flow $s \in \mathcal{S}$ which has neither a GMM-bottleneck link with respect to r nor a rate assignment equal to its PR. Then, for every nonsaturated link ℓ ($F_\ell < C_\ell$) traversed by s, we can increase r_s by an increment until it reaches the PR of s or some link saturates, whichever is smaller. For every saturated link ℓ ($F_\ell = C_\ell$) traversed by s, if $T = \{t \mid r_t > \mathrm{MR}_t, t \text{ traversing } \ell\}$ is nonempty, there must exist a flow $p \in T$, $p \neq s$, such that $r_p > r_s$. Thus, the quantity

$$\delta_\ell = \begin{cases} \min\{(C_\ell - F_\ell),\ (\mathrm{PR}_s - r_s)\} & \text{if } F_\ell < C_\ell, \\ \min\{(r_p - r_s),\ (r_p - \mathrm{MR}_p),\ (\mathrm{PR}_s - r_s)\} & \text{if } F_\ell = C_\ell \end{cases}$$

is positive. Now, let δ be the minimum of δ_ℓ over all links ℓ traversed by s. Therefore, we can increase r_s by δ while decreasing the same amount of rate from flow r_p on the links ℓ traversed by s with $F_\ell = C_\ell$. We maintain GMM-feasibility without decreasing the rate of any flow t with $r_t \leq r_s$. This contradicts the GMM definition of the rate vector r.

For the proof of the *"if"* part of Theorem 1, we assume that each flow has either a GMM-bottleneck link with respect to the GMM-feasible rate vector r or a rate assignment equal to its PR.

- *Case 1:* To increase the rate of any flow s with $r_s < \mathrm{PR}_s$ while maintaining GMM-feasibility, we

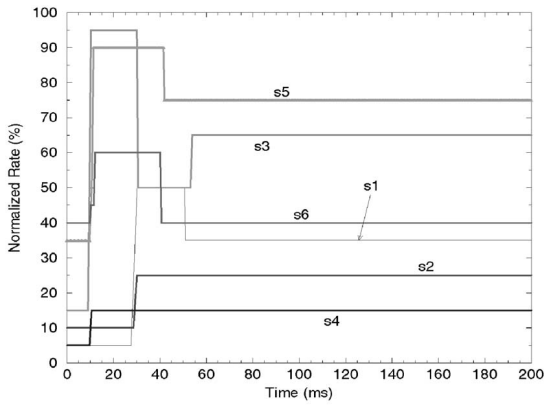

Fig. 9. The rates of all flows for the chain network configuration.

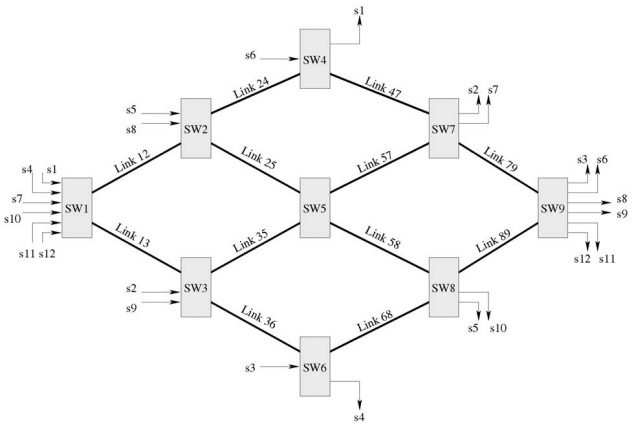

Fig. 10. A mesh network topology.

TABLE 7
Flow Route, MR Requirement, PR Constraint, and GMM Rate Allocation of Each Flow for the Mesh Network Topology

Flow	Route	MR	PR	GMM Rate Allocation
$s1$	1–2–4	0.50	0.80	0.50
$s2$	3–5–7	0.05	0.25	0.25
$s3$	6–8–9	0.10	0.90	0.30
$s4$	1–3–6	0.15	0.80	0.30
$s5$	2–5–8	0.05	0.15	0.15
$s6$	4–7–9	0.10	0.15	0.15
$s7$	1–2–4–7	0.15	0.70	0.25
$s8$	2–5–8–9	0.10	0.90	0.30
$s9$	3–5–7–9	0.05	0.80	0.35
$s10$	1–3–6–8	0.20	1.00	0.30
$s11$	1–2–5–7–9	0.10	0.70	0.25
$s12$	1–3–5–8–9	0.40	0.80	0.40

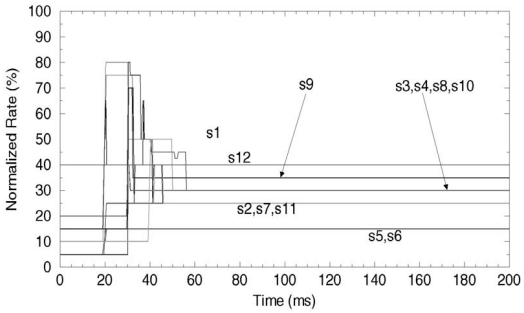

Fig. 11. The rates of all flows for the mesh network topology.

must decrease the rate of some flow p with $r_p > \mathrm{MR}_p$ and p traverses the GMM-bottleneck link ℓ of s (flow s must go through a GMM-bottleneck link since $r_s < \mathrm{PR}_s$ and we have $F_\ell = C_\ell$ by the definition of a GMM-bottleneck link). Since $r_s \geq r_p$ for all p in $T = \{t \mid r_t > \mathrm{MR}_t, t \text{ traversing } \ell\}$ by the definition of GMM-bottleneck link, the rate assignment for any flow $s \in \mathcal{S}$ with $r_s < \mathrm{PR}_s$ satisfies the definition for GMM.

- *Case 2:* For any flow s with $r_s = \mathrm{PR}_s$, we cannot further increase the rate of r_s while maintaining GMM-feasibility. That is, we cannot generate another GMM-feasible rate vector $\hat{r} = \{\hat{r}_s \mid s \in \mathcal{S}\}$ with $\hat{r}_s > r_s$. Thus, the rate assignment for any flow s with $r_s = \mathrm{PR}_s$ satisfies the requirement for GMM rate allocation.

Combining Cases 1 and 2, we have proven the "*if*" part of the theorem. The proof for Theorem 1 is now complete. □

APPENDIX 2

Proof of Lemma 2. Case 1: First, consider link $\ell \in \mathcal{L}_1$. Since $\alpha_\ell = \sigma_1$ for $\ell \in \mathcal{L}_1$ and, by Lemma 1, $\varphi_\ell \geq \alpha_\ell$ for every $\ell \in \mathcal{L}$, we have $\varphi_\ell \geq \sigma_1$ for every $\ell \in \mathcal{L}_1$. For link $\ell \in (\mathcal{L} - \mathcal{L}_1)$, since $\alpha_\ell > \sigma_1$ for $\ell \in (\mathcal{L} - \mathcal{L}_1)$, we have $\varphi_\ell > \sigma_1$ for every $\ell \in (\mathcal{L} - \mathcal{L}_1)$.

Case 2: Since $\alpha_\ell > \sigma_1$ and, by Lemma 1, $\varphi_\ell \geq \alpha_\ell$ for every $\ell \in \mathcal{L}$, we have $\varphi_\ell > \sigma_1$ for every $\ell \in \mathcal{L}$. □

APPENDIX 3

Proof of Lemma 3. 1) In this case, by Case 1 of Lemma 2, there exists a $t_1 \geq 0$ such that, for $t \geq t_1$,

$$\varphi_\ell \geq \sigma_1 \quad \text{for every } \ell \in \mathcal{L}_1, \tag{9}$$

$$\varphi_\ell > \sigma_1 \quad \text{for every } \ell \in (\mathcal{L} - \mathcal{L}_1). \tag{10}$$

We will show that there exists a time t_2 such that, for $t \geq t_2$, the following statements are true:

$$\text{The ER field of every returning RM packet of flow } i \in \Psi_1 \text{ satisfies ER } \geq \sigma_1; \tag{11}$$

$$\text{The recorded CR of flow } i \in \Psi_1 \text{ satisfies } r_\ell^i \geq \sigma_1 \text{ at every link } \ell \text{ traversed by flow } i \in \Psi_1. \tag{12}$$

To see that (11) and (12) hold, consider that the first RM packet of flow $i \in \Psi_1$ leaves the source after time t_1. When this RM packet returns to the source at some time $t_1^{RTT} \geq t_1$, the ER field is set to[7]

$$\mathrm{ER} := \max\left\{\min\left\{\mathrm{PR}^i, \min_{\ell \text{ traversed by } i} \varphi_\ell\right\}, \ \mathrm{MR}^i\right\}. \tag{13}$$

Since 1) $\mathrm{PR}^i \geq \sigma_1$ for $i \in \Psi_1$ and 2) $\varphi_\ell \geq \sigma_1$ for every $\ell \in \mathcal{L}_1$, we have that, for $t \geq t_1^{RTT}$,

$$\mathrm{ER} \geq \max\{\sigma_1, \ \mathrm{MR}^i\} \geq \sigma_1 \quad \text{for } i \in \Psi_1.$$

Note that any feedback RM packet arriving at the source after time t_1^{RTT} corresponds to a forward RM packet which left the source after time t_1. Applying the above arguments to *any* such returning RM packet of flow $i \in \Psi_1$ and note that (9) holds for $t \geq t_1$, we have that (11) is true for $t \geq t_1^{RTT}$.

At time t_1^{RTT}, the AR at the source is set to ER and $\mathrm{AR}(t_1^{RTT}) \geq \sigma_1$. Since (11) holds for $t \geq t_1^{RTT}$, we have that the AR at source for flow $i \in \Psi_1$ satisfies $\mathrm{AR}(t) \geq \sigma_1$ for $t \geq t_1^{RTT}$.

Let $t_1^{1.5RTT}$ denote the time when an RM packet arrives at its destination after it leaves the source after time t_1^{RTT}. The recorded rate of flow $i \in \Psi_1$ at any link on its way is set after $t_1^{1.5RTT}$. We have shown that every RM packet of flow $i \in \Psi_1$ leaving the source has its CR rate set to AR, which is greater than or equal to σ_1 for any time $t \geq t_1^{RTT}$. Hence, the recorded rate r_ℓ^i satisfies (12) for $t \geq t_1^{1.5RTT}$. Let $t_2 = t_1^{1.5RTT}$ and we have proved (11) and (12).

To prove statement 1a of Lemma 3, consider any link $\ell \in \mathcal{L}_1$. Note that, in this case, only flows from Ψ_1 traverse links of $\mathcal{L}_1$. Let $\mathcal{M}_\ell$ and $\mathcal{U}_\ell$ be the set of marked and unmarked flows, respectively. Then,

- *Case 1:* Suppose that not all flows are marked, then

$$\varphi_\ell \cdot \sum_{i \in \mathcal{U}_\ell} 1^+\{\mathrm{MR}^i \leq \varphi_\ell\} + \sum_{i \in \mathcal{U}_\ell} \mathrm{MR}^i \cdot 1^+\{\mathrm{MR}^i > \varphi_\ell\} = C_\ell - \sum_{i \in \mathcal{M}_\ell} r_\ell^i$$

or

$$\sum_{i \in \mathcal{M}_\ell} r_\ell^i + \varphi_\ell \cdot \sum_{i \in \mathcal{U}_\ell} 1^+\{\mathrm{MR}^i \leq \varphi_\ell\} + \sum_{i \in \mathcal{U}_\ell} \mathrm{MR}^i \cdot 1^+\{\mathrm{MR}^i > \varphi_\ell\} = C_\ell.$$

But,

$$\sigma_1 \cdot \sum_{i \in \mathcal{S}_\ell} 1^+\{\mathrm{MR}^i \leq \sigma_1\} + \sum_{i \in \mathcal{S}_\ell} \mathrm{MR}^i \cdot 1^+\{\mathrm{MR}^i > \sigma_1\} = C_\ell$$

for $\ell \in \mathcal{L}_1$, and $r_\ell^i \geq \sigma_1$ for $i \in \mathcal{M}_\ell$ by (12). Therefore, $\varphi_\ell \leq \sigma_1$ for $t \geq t_1^{1.5RTT}$.

7. Note that the RTT term is used as a generic term for round trip time to discuss what kind of outcome will happen after an RM packet completes one round trip time. It is not used as a precise measure for time.

By (9), $\varphi_\ell \geq \sigma_1$ for every link $\ell \in \mathcal{L}_1$ for $t \geq t_1$, we have $\varphi_\ell = \sigma_1$ for $t \geq t_1^{1.5RTT}$.[8]

- *Case 2:* We now assume that all flows are marked. We will show that this will lead to contradiction and, thus, only Case 1 is possible. Let flow $p \in \mathcal{S}_\ell$ be the flow such that $r_\ell^p = \max_{i \in \mathcal{S}_\ell} r_\ell^i$. Then,

$$\begin{aligned}\varphi_\ell &= C_\ell - \sum_{i \in \mathcal{S}_\ell} r_\ell^i + \max_{i \in \mathcal{S}_\ell} r_\ell^i \\ &= \sigma_1 \cdot \sum_{i \in \mathcal{S}_\ell} 1^+\{\mathrm{MR}^i \leq \sigma_1\} + \sum_{i \in \mathcal{S}_\ell} \mathrm{MR}^i \cdot 1^+\{\mathrm{MR}^i > \sigma_1\} \\ &\quad - \sum_{i \in \mathcal{S}_\ell} r_\ell^i + \max_{i \in \mathcal{S}_\ell} r_\ell^i \\ &\leq \max_{i \in \mathcal{S}_\ell} r_\ell^i. \end{aligned} \tag{14}$$

The last inequality follows from (12) for $t \geq t_1^{1.5RTT}$ and Fact 1, $r_\ell^i \geq \mathrm{MR}^i$, $i \in \mathcal{S}_\ell$. On the other hand, since all flows are marked, $\varphi_\ell > \max_{i \in \mathcal{S}_\ell} r_\ell^i$. This contradicts (14). Therefore, flows in Ψ_1 cannot be all marked on link $\ell \in \mathcal{L}_1$.

Combining Cases 1 and 2 above, statement 1a of Lemma 3 hold for $t \geq t_1^{1.5RTT}$.

Note that flow $i \in \Psi_1$ traverses at least one link $\ell \in \mathcal{L}_1$. By (13) and statement 1a of this lemma, any RM packet that arrives at its destination after $t_1^{1.5RTT}$ returns to the source with the ER field set to $\max\{\sigma_1,\ \mathrm{MR}\}$, $i \in \Psi_1$. Denote the time of the return of this feedback RM packet to the source by t_1^{2RTT}. This shows that statement 1b of the lemma is true for $t \geq t_1^{2RTT}$. It then follows that, for $t \geq t_1^{2RTT}$, the AR at the source is set to $\max\{\sigma_1,\ \mathrm{MR}\}$, $i \in \Psi_1$, which is statement 1c of the lemma.

Let $t_1^{2.5RTT}$ be the time of an RM packet arriving at its destination after leaving the source after t_1^{2RTT}. Then, with the operation of the algorithm, every flow $i \in \Psi_1$ with $r_\ell^i = \sigma_1$ will be marked with $b_\ell^i = 1$ at every link it traverses, except at its GMM-bottleneck link $\ell \in \mathcal{L}_1$, and will remain marked ever after as long as the set of flows remain unchanged for $t \geq t_1^{2.5RTT}$. Thus, statement 1d of Lemma 3 also holds.

So far, we have proven that statements 1a to 1d of Lemma 3 hold for $t \geq t_1^{2.5RTT}$.

To see that statement 1e of Lemma 3 is true, consider that the first RM packet of flow $j \in (\mathcal{S} - \Psi_1)$ leaves the source after time t_1. When this RM packet returns to the source at some time $t_1^{RTT} \geq t_1$, the ER field is set to

$$\mathrm{ER} := \max\left\{\min\left\{\mathrm{PR}^j,\ \min_{\ell \text{ traversed by } j} \varphi_\ell\right\},\ \mathrm{MR}^j\right\}.$$

Since $\mathrm{PR}^j > \sigma_1$ for $j \in (\mathcal{S} - \Psi_1)$ and $\varphi_\ell > \sigma_1$ for every $\ell \in (\mathcal{L} - \mathcal{L}_1)$, we have that, for $t \geq t_1^{RTT}$,

$$\mathrm{ER} > \sigma_1 \quad \text{for } j \in (\mathcal{S} - \Psi_1).$$

Now, using similar arguments as above for the proofs of (11) and (12) and taking (10) into account, it can be shown that statements 1e to 1g hold for $t \geq t_1^{1.5RTT}$.

Denote $T_1 = t_1^{2.5RTT}$. Then, all statements in case 1 of Lemma 3 are proved.

2) The proof of this case is simpler than Case 1 and also follows similar steps. We omit presenting it here to conserve space.

Finally, let D denote the maximum round-trip time among all flows. Then, we have just shown that it takes at most $2.5D$ for flows $s \in \Psi_1$ to converge to GMM rate allocation in the distributed algorithm. □

APPENDIX 4

Proof of Lemma 4. By the induction hypothesis, for $t \geq T_n$, 1) every flow $p \in \Psi_j$, $1 \leq j \leq n$ has reached its GMM rate allocation ($\max\{\sigma_j,\ \mathrm{MR}^p\}$) in case 1 or PR^p in case 2, and these rates do not change as long as the set of flows in the network remain unchanged; and 2) every flow $p \in \Psi_j$, $1 \leq j \leq n$ is in the following marking state:

- If $r_\ell^p = \mathrm{MR}^p$, then flow p is not marked at its GMM-bottleneck link and may be marked at other links it traverses.
- If $r_\ell^p = \mathrm{PR}^p$, then flow p is marked at every link it traverses.
- If $r_\ell^p = \sigma_j$, then flow p is marked at all of its traversing links except its GMM-bottleneck link.

Since, by the induction hypothesis, every flow in $(\Psi_1 \cup \cdots \cup \Psi_n)$ has stabilized at its GMM rate allocation with one of the aforementioned marking states on the traversing links along its path for $t \geq T_n$, we can therefore consider a reduced network with links $\hat{\mathcal{L}} = \mathcal{L}_{n+1} \cup \mathcal{L}_{n+2} \cup \cdots \cup \mathcal{L}_N$,[9] flows $\hat{\mathcal{S}} = \mathcal{S} - (\Psi_1 \cup \cdots \cup \Psi_n) = \Psi_{n+1} \cup \cdots \cup \Psi_N$, and link capacities $\hat{C}_\ell = C_\ell - \sum_{\text{flow } p \in (\Psi_1 \cup \cdots \cup \Psi_n) \text{ traversing link } \ell} r_\ell^p$, $\ell \in \hat{\mathcal{L}}$. Denote $\hat{n}_\ell$ the number of flows traversing link ℓ in the reduced network $(\hat{\mathcal{L}}, \hat{\mathcal{S}}, \hat{C})$. For the reduced network $(\hat{\mathcal{L}}, \hat{\mathcal{S}}, \hat{C})$, let

$$\hat{\alpha}_\ell \cdot \sum_{i \in \hat{\mathcal{S}}_\ell} 1^+\{\mathrm{MR}^i \leq \hat{\alpha}_\ell\} + \sum_{i \in \hat{\mathcal{S}}_\ell} \mathrm{MR}^i \cdot \{\mathrm{MR}^i > \hat{\alpha}_\ell\} = \hat{C}_\ell$$

and reapply the arguments used in the proof of Lemma 1, we have $\varphi_\ell \geq \hat{\alpha}_\ell$ for every $\ell \in \hat{\mathcal{L}}$. Using similar arguments as for the proof of Lemma 2, it is straightforward to show that statements similar to Lemma 2 hold for the reduced network. That is,

1. If $\sigma_{n+1} = \hat{\alpha}_\ell \leq \mathrm{PR}^s$ for $s \in \Psi_{n+1}$, i.e., the GMM-bottleneck link rate is reached before some flow $s \in \Psi_{n+1}$ reaches its PR in the centralized algorithm, then, for the distributed algorithm, we have

$$\begin{aligned}&\varphi_\ell \geq \sigma_{n+1} \quad \text{for every } \ell \in \mathcal{L}_{n+1}, \\ &\varphi_\ell > \sigma_{n+1} \quad \text{for every } \ell \in (\hat{\mathcal{L}} - \mathcal{L}_{n+1}).\end{aligned}$$

2. If $\sigma_{n+1} = \mathrm{PR}^s < \hat{\alpha}_\ell$ for $s \in \Psi_{n+1}$, i.e., some flow $s \in \Psi_{n+1}$ reaches its PR before the GMM-bottleneck link rate is reached in the centralized

8. This shows that $\mathcal{M}_\ell$ must be an empty set on $\ell \in \mathcal{L}_1$. That is, none of the flows $i \in \Psi_1$ is marked on link $\ell \in \mathcal{L}_1$.

9. Note that $\hat{\mathcal{L}} = \mathcal{L}_{n+1} \cup \mathcal{L}_{n+2} \cup \cdots \cup \mathcal{L}_N$ may not be the same as $\mathcal{L} - (\mathcal{L}_1 \cup \mathcal{L}_2 \cup \cdots \cup \mathcal{L}_n)$ since links in $\mathcal{L}_n$ may be part of $\mathcal{L}_{n+1}$.

algorithm, then, for the distributed algorithm, we have

$$\varphi_\ell > \sigma_{n+1} \quad \text{for every } \ell \in \hat{\mathcal{L}}.$$

Now, using the same token as in the proof of Lemma 3 for the reduced network, we can show that all the statements of Lemma 4 hold for $n+1$. □

ACKNOWLEDGMENTS

The authors wish to thank the anonymous reviewers for their constructive comments and suggestions, which have significantly improved the presentation of this paper.

REFERENCES

[1] The ATM Forum Technical Committee, Traffic Management Specification, Version 4.0, ATM Forum Contribution, AF-TM 96-0056.00, Apr. 1996.
[2] D. Bertsekas and R. Gallager, *Data Networks,* chapter 6. Prentice Hall, 1992.
[3] F. Bonomi and K.W. Fendick, "The Rate-Based Flow Control Framework for the Available Bit Rate ATM Service," *IEEE Network Magazine,* vol. 9, no. 2, pp. 25-39, Mar./Apr. 1995.
[4] A. Charny, D. Clark, and R. Jain, "Congestion Control with Explicit Rate Indication," *Proc. IEEE Int'l Conf. Comm.,* pp. 1954-1963, June 1995.
[5] E.M. Gafni, "The Integration of Routing and Flow Control for Voice and Data in a Computer Communication Network," PhD thesis, Dept. of Electrical Eng. and Computer Science, MIT, Cambridge, Mass., Aug. 1982.
[6] H.P. Hayden, "Voice Flow Control in Integrated Packet Networks," MS thesis, Dept. of Electrical Eng. and Computer Science, MIT, Cambridge, Mass., June 1981.
[7] NIST Network Simulator, http://w3.antd.nist.gov/Hsntg/prd_atm-sim.html, NIST, 2003.
[8] Y.T. Hou, H. Tzeng, S.S. Panwar, and V.P. Kumar, "ATM ABR Traffic Control with a Generic Weight-Based Bandwidth Sharing Policy: Theory and a Simple Implementation," *IEICE Trans. Comm.,* vol. E81-B, no. 5, pp. 958-972, May 1998.
[9] Y.T. Hou, S.S. Panwar, and H. Tzeng, "Available Bit Rate Flow Control for Service Allocation in a Packet Network," US Patent #6,515,965, Feb. 2003.
[10] J.M. Jaffe, "Bottleneck Flow Control," *IEEE Trans. Comm.,* vol. 29, no. 7, pp. 954-962, July 1981.
[11] L. Kalampoukas, A. Varma, and K.K. Ramakrishnan, "An Efficient Rate Allocation Algorithm for ATM Networks Providing Max-Min Fairness," *Proc. Sixth IFIP Int'l Conf. High Performance Networking,* pp. 143-154, Sept. 1995.
[12] S. Kalyanaraman, R. Jain, S. Fahmy, R. Goyal, and B. Vandalore, "The ERICA Switch Algorithm for ABR Traffic Management in ATM Networks," *IEEE/ACM Trans. Networking,* vol. 8, no. 1, pp. 87-98, Feb. 2000.
[13] Y. Kim, W.K. Tsai, M. Iyer, and J. Ros, "Minimum Rate Guarantee without Per-Flow Information," *Proc. IEEE Int'l Conf. Network Protocols,* pp. 155-162, 1999.
[14] J. Mosely, "Asynchronous Distributed Flow Control Algorithms," PhD thesis, Dept. of Electrical Eng. and Computer Science, MIT, Cambridge, Mass., June 1984.
[15] K.K. Ramakrishnan, R. Jain, and D.-M. Chiu, "Congestion Avoidance in Computer Networks with a Connectionless Network Layer—Part IV: A Selective Binary Feedback Scheme for General Topologies Methodology," DEC-TR-510, Digital Equipment Corp., 1987.
[16] L. Roberts, "Enhanced PRCA (Proportional Rate Control Algorithm)," ATM Forum Contribution, AF-TM 94-0735R1, Aug. 1994.
[17] K.-Y. Siu and H.-Y. Tzeng, "Intelligent Congestion Control for ABR Service in ATM Networks," *ACM SIGCOMM Computer Comm. Rev.,* vol. 24, no. 5, pp. 81-106, Oct. 1994.
[18] W.K. Tsai and J. Ros, "Maxmin Lambda Allocation for Dense Wavelength-Division-Multiplexing Networks," *J. Optical Networking,* vol. 1, nos. 8/9, pp. 323-337, Aug. 2002.
[19] N. Yin and M.G. Hluchyj, "On Closed-Loop Rate Control for ATM Cell Relay Networks," *Proc. IEEE INFOCOM,* pp. 99-108, June 1994.
[20] N. Yin, "Max-Min Fairness vs. MCR Guarantee on Bandwidth Allocation for ABR," *Proc. IEEE ATM Workshop,* Aug. 1996.

Y. Thomas Hou received the BE degree from the City College of New York in 1991, the MS degree from Columbia University in 1993, and the PhD degree from Polytechnic University, Brooklyn, New York, in 1998, all in electrical engineering. From 1997 to 2002, Dr. Hou was a research scientist and project leader at Fujitsu Laboratories of America, IP Networking Research Department, Sunnyvale, California. He is currently an assistant professor at Virginia Tech, The Bradley Department of Electrical and Computer Engineering, Blacksburg, Virginia. Dr. Hou's current research focuses on wireless sensor networks and multimedia delivery over wireless networks. He is a corecipient of the 2002 IEEE International Conference on Network Protocols (ICNP) Best Paper Award and the 2001 *IEEE Transactions on Circuits and Systems for Video Technology* Best Paper Award. He is a senior member of the IEEE and the IEEE Computer Society, and a member of the ACM.

Shivendra S. Panwar received the BTech degree in electrical engineering from the Indian Institute of Technology, Kanpur, in 1981, and the MS and PhD degrees in electrical and computer engineering from the University of Massachusetts, Amherst, in 1983 and 1986, respectively. He joined the Polytechnic University and is now a professor in the Electrical and Computer Engineering Department. He is currently the director of the New York State Center for Advanced Technology in Telecommunications (CATT). His research interests include performance analysis and design of networks. His current work includes protocol analysis, traffic and call admission control, switch performance, and multimedia transport over wireless networks. He is a coeditor of two books, *Network Management and Control, Vol. II,* and *Multimedia Communications and Video Coding*, both published by Plenum. He is a senior member of the IEEE.

Henry H.-Y. Tzeng received the MS and PhD degrees from the University of California (UC), Irvine, in 1993 and 1995, respectively. He is currently a senior software manager with the Nokia Network, Mountain View, California, where he is leading the development of intelligent edge devices for mobile packet core network. From 1999 to 2001, he was with Amber Networks and built the industry-first carrier-class fault-tolerant IP routers. From 1995 to 1999, he was a member of the technical staff, High-Speed Network Research, Bell Labs, Lucent Technologies, Holmdel, New Jersey. Since 1995, he has been working on high-performance and fault-resilient routing technologies and their applications to wireless and wired networks, leading to several patents and publications in the areas of fast-search algorithms, link-state routing, and interdomain routing protocols. He also has publications on reliable multicast protocols and ATM traffic management. Dr. Tzeng was a corecipient of the 1997 IEEE Browder J. Thompson Memorial Prize Award and also the UC Regent's Dissertation Fellowship in 1995.