HAL
open science

Resource Sharing in the Most Regular Scheduling: Deterministic Performance and Guarantee

Chung Shue Chen, Wing Shing Wong

To cite this version:

Chung Shue Chen, Wing Shing Wong. Resource Sharing in the Most Regular Scheduling: Deterministic Performance and Guarantee. IEEE Military Communications Conference, IEEE, Oct 2006, Washington D. C., United States. 10.1109/MILCOM.2006.302457 . hal-00745063

HAL Id: hal-00745063

https://inria.hal.science/hal-00745063

Submitted on 24 Oct 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

RESOURCE SHARING IN THE MOST REGULAR SCHEDULING: DETERMINISTIC PERFORMANCE AND GUARANTEE

Chung Shue Chen[†*], Wing Shing Wong[*]
[*] Department of Information Engineering, The Chinese University of Hong Kong, Shatin, Hong Kong
[†] LORIA-CNRS, Rue du Jardin Botanique, 54600 Villers Les Nancy, France
Email: cschen@ieee.org, wswong@ie.cuhk.edu.hk

ABSTRACT

Providing performance guarantee is one of the most important issues in a heterogeneous network for a large number of concurrent sessions. A common question in the network control and resource scheduling is how to provide end-to-end delay bounds and offer service guarantees in a relatively distributed manner. In this paper, we will narrow down the question and address specifically on the MRBS multiple access [25]. It supports each user a flexible rate-matched time sharing with efficient resource scheduling. A generalized system modeling and analytical framework is developed based on the techniques of network calculus [13]. The result extends from single-node scheduling to multiple layered scenarios with deterministic performance guarantees. Potential applications are discussed at last.

I. INTRODUCTION

No matter what kind of scheme is used, there is always a high demand on efficient resource allocation for a communication system. Quality of service (QoS) guarantees associated with network elements such as traffic shaper, flow control and scheduling processor have been discussed in earlier works such as [1–7]. Investigations of queuing networks on their burstiness constraints, scheduling algorithm, and trade-offs between bandwidth provisioning and delay are conducted. Concepts of delay calculus and traffic characterization built on leaky bucket [2] constrained sessions, arrival curve and service curve are initialized in [3, 4]. The ideal model of generalized processor sharing [5,6] has been introduced in single node and multiple node scenarios respectively. Fair rate-based service was addressed.

Meanwhile, generalized scheduling policy for performance guarantees has raised many discussions such as [8–12]. For example, based on per node traffic shaping, end-to-end network QoS provisioning has been addressed in [8, 9]. Delay and backlog of flow control were studied. Discussions on scheduling algorithms and corresponding scalability for high speed packet switching in large scale networks can be found in [10, 11]. Generally speaking, the end-to-end delay in these mixed traffic networks is usually to be bounded over conservatively [12]. This could consequently limit the utility of obtained results to guide for an efficient resource allocation and traffic scheduling.

The work described in this paper was supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region (Project no. CUHK4366/02E). The work was carried out when Chung Shue Chen was with the Department of Information Engineering, The Chinese University of Hong Kong, Shatin, Hong Kong.

In [13], network calculus has been introduced for deterministic analysis of queuing system or flow control in communication networks. It provides an insight on how to have the system modeling and analytical investigation. The filtering theories of min-plus and max-plus algebra [14, 15] are established to measure the worst-case delay and backlog in packet flows with reservation mechanism. In addition, fluid models of queuing networks are considered in [16,17]. The result can be referred as stochastic performance guarantees. Some recent discussions on the wired and wireless QoS can also be found in [18–23].

Here, we will narrow down the question and address specifically on the resource allocation and access control based on the most regular sequence [24]. Bandwidth allocation for multi-rate operations can be provided in a relatively distributed and contention-resolved approach based on limited information exchange. In the following studies, we will have a generalization of the previous work [25] in both single node and multiple node scenarios. A systematic modeling is established for analytical performance investigation. The result has potential applications in various communication systems such as wireless mesh network [21].

II. NETWORK MODEL

Deterministic performance guarantee refers to the availability of QoS bounds in a traffic flow or queuing system [17], which may refer to a wide range of quantities such as transmission bandwidth, queue length, packet delay and transmit jitter. For a service assurance, application source may specify their traffic characteristics and make a request to the network control. In response to the requirement, the network provides a corresponding QoS guarantee based on the requested quantities by allocating or reserving an amount of the network resource accordingly. This refers to a QoS contract. As long as the source follows its traffic specification, the network should meet the agreed service generally.

Here, we focus on the studies of worst case delay bounds and service guarantees for each transmission request in mixed traffics. In a typical way, packet delay in a network can be separated as queuing delay and propagation delay. We are concentrated on the queuing delay involved in traffic scheduling and resource sharing [23]. It is assumed that the system is in a stable state under admission control such that the sum of requested service rates on a link is less than or equal to the available bandwidth. The scope here is limited to a quasi-stationary model in which requested rates are fixed within the control window. However, the result can serve as a basis and be extended for dynamic and adaptive models in a future work.

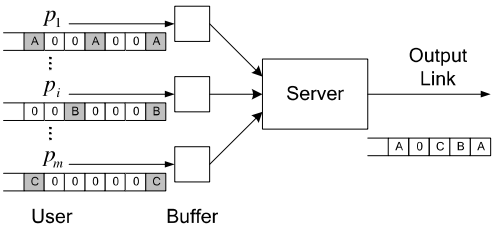

Figure 1: A network element with multiple inputs and each of them requests a rate p_i on the output link.

A. Data Packetization

Raw data from a source entering the network will pass through a data packetizer and a traffic regulator before delivered to a scheduler or other network elements. For a fixed-size packet service, the data packetizer assembles the raw data into conventional packets of a common size. Shaped data stream is sent to consequent network elements for further processing. This can be traced back to the widely adopted leaky bucket model [2], which often serves as an admission rate control and is able to specify incoming traffic characterization. Moreover, the traffic shaping can smooth bursty arrivals so as to avoid service unfairness and performance corruption to a system.

B. Network Element

In a general way, as a packet of a traffic stream passes through a network, it usually involves a sequence of servers and switching elements along the path from source to destination. For traffic scheduling and routing, they may include multiplexer, demultiplexer, traffic regulator, buffer and so on. Given a network path, the combination of these elements will figure out the resultant traffic characteristics. Fig. 1 gives an example in which there are m users requesting transmissions onto an output link. Here, the server acts as a scheduler to multiplex incoming regulated traffics. Some service disciplines for the server are necessary so as to meet specified service requirements such as transmission rate, p_i, and buffer constraints. This compound can be represented by a multiplexer with a specific output control.

C. Transmission Link

An ideal link [17] with capacity C is a network element in which the number of packets departed from is upper bounded by C packets per unit time. Here, a link is work-conserving. In other words, it transmits once arrival packets reach the buffer as long as the departure rate is less than or equal to the link capacity. Otherwise, arrivals will be buffered and the buffer for the link is assumed infinite.

III. A Framework of MRBS Scheduling in Single Node Scenarios

First, we restrict our attention to the most regular resource scheduling in single-node scenarios. The result is established on the most regular binary sequence (MRBS) [24]. A major part of the discussion here is on how to implement the MRBS scheduling in a network sense and to show that the analysis can be developed in a systematic way with deterministic performance guarantees [17]. The multiple node scenarios will be discussed afterward.

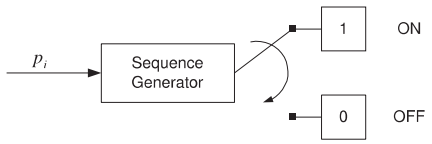

Figure 2: In an MRBS generator, '1' and '0' refer to source on and off respectively. The sequence mean is equal to p_i.

A. MRBS Traffic Regulator

Recall the definition [24, 25], an MRBS is expressible as

$$s_i(j) = \lceil (j+1)p_i \rceil - \lceil j \times p_i \rceil \tag{1}$$

for any $0 \leq p_i \leq 1$, where p_i refers to the requested proportional rate of a user i while j is the discrete time index starting from 0. The generated sequence has the following properties: (i) it has an asymptotic mean equal to p_i, (ii) the generated zero-ones are most regularly distributed in the sense [24] that this evenly spaced binary sequence has the most regular inter-arrival time among all arrival processes of the same requested rate and leads to the smallest delay mean and all its moments, as well as other related quantities, and (iii) it is periodic for any rational number p_i. For example, when $p_i = 3/7$, $s_i(j)$ is cyclic and has a core pattern of $\{1, 0, 1, 0, 1, 0, 0\}$. The period is equal to 7.

An MRBS traffic regulator is defined with respect to (1). As shown in Fig. 2, it logically acts like a switching element with on-offs regulated by the generated sequence. It can be represented by a leaky bucket as well. Tokens of fixed size are employed and each is equivalent to one packet. The token generation is governed by MRBS with corresponding rate p_i. As a result, any traffic passing through will be shaped into the MRBS pattern.

Definition 1 *For any MRBS-shaped arrivals, we denote the resultant traffic A_i of rate p_i as a discrete time process by*

$$A_i(n) \sim \textit{MRBS}(p_i) \tag{2}$$

where n is the discrete time index starting from 0 and all the packets have a fixed size.

B. Rate Matched Flow Control

As shown in [25], resource sharing among users in a common channel can be accommodated by the scheduling with MRBS. Given a requested rate R_i, the permission of transmission or channel access for user i is addressed by the '1's of MRBS with respect to the normalized rate, $p_i = R_i/R$, where R is the total bandwidth. Multi-rate services can be matched exactly. Since scheduling conflicts may occur among users if MRBS is applied to each individual independently without any synchronization, a generalized contention resolution is helpful for efficient transmissions. A conflict-free result is achievable by the proposed slot scheduling algorithm MRBS-SSA [25], which is a rate-matched flow control based on the service requirement and transmission urgency. The rules can be summarized as:

1. A user with buffered data has the highest priority so as to avoid excessive delay and buffer overflow.
2. For users with the same priority according to the first rule, the one with earlier arrival has a higher priority.

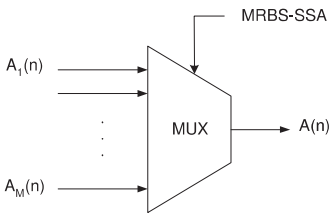

Figure 3: For an MRBS multiplexer, given inputs of MRBS, the merged output is always an MRBS as well.

3. If previous rules fail to resolve the priority, a user requesting a higher rate has a higher priority.
4. Users with the same rate are assigned different predefined priorities to resolve an ambiguity.

By MRBS-SSA, with positive rational p_i, the resultant transmission ordering of any user i is always deterministic and periodic. This offers a relatively distributed multiple access control in the sense that any user can individually generate the equivalent scheduling result as long as the information of requested rates p_i is given. The result can be encapsulated as an MRBS multiplexer described below.

C. MRBS Multiplexer

An ideal multiplexer (MUX) is a network element which combines inputs into an aggregate signal to be transported over a transmission link and its output is equivalent to the sum of the inputs. To be precise, assume there are M inputs and each is denoted by $A_i(n)$, there exists an integer $N \geq 0$ such that the output, denoted by $A(n)$, is expressible as

$$\sum_{n=0}^{N} A(n) = \sum_{n=0}^{N} \sum_{i=1}^{M} A_i(n) \tag{3}$$

where $i = 1, 2, \ldots, M$, and n starts from 0. In convention, this multiplexer is a non-blocking switching device with a large enough buffering for each arrival process.

An MRBS multiplexer is an ideal multiplexer with multiplexing rules according to the MRBS-SSA. The model is shown in Fig. 3. For input of MRBS arrivals, the output is always an MRBS traffic with the aggregate rate

$$p_{\tilde{M}} = \sum_{i=1}^{M} p_i \tag{4}$$

where p_i corresponds to each individual rate. We have

$$A(n) \sim \text{MRBS}(p_{\tilde{M}}) \tag{5}$$

as the merger of inputs $A_i(n) \sim \text{MRBS}(p_i)$.

Theorem 1 *For any number of inputs $A_i(n) \sim$ MRBS(p_i), where $i = 1, 2, \ldots, M$, the output of an MRBS multiplexer is also an MRBS traffic which can be characterized as $A(n) \sim$ MRBS$(p_{\tilde{M}})$, where $p_{\tilde{M}} = \sum_{i=1}^{M} p_i$.*

Proof: By MRBS-SSA, it has been shown [25] that a buffer of size analogously equal to one packet is sufficient to achieve a collision-free scheduling for any time instant provided that $A_i(n) \sim \text{MRBS}(p_i)$. Besides, there is no early transmission. So, all service slots addressed by the aggregate rate are utilized. Hence, by an MRBS multiplexer, $A(n) \sim \text{MRBS}(p_{\tilde{M}})$ as long as $A_i(n) \sim \text{MRBS}(p_i)$. □

The operation of an MRBS multiplexer can be shown by the demonstration depicted in Fig. 1 with MRBS-SSA employed for the multiplexing. Three MRBS input streams, labeled by A, B and C, with $p_i = \{1/3, 1/4, 1/6\}$ and their multiplexed result with respect to an MRBS allocation of the aggregate rate $p_{\tilde{M}} = 3/4$ are shown. A buffer of size equal to one packet at each input is sufficient to offer a lossless multiplexing and combining. In the scheduling model shown by Fig. 1, the minimum necessary buffering of size one is employed conceptually so as to avoid excessive delay. The MRBS-SSA schedules arrivals and is capable of providing a fully flexible rate-matched service for each request. Moreover, for any set of rational number p_i, a solution of finite integer N for (3) always exists. Let $p_i = m_i/l_i$ where m_i and l_i are relatively prime integers, a solution of N equals to the lowest common multiple of l_i. The merged result is periodic and deterministic.

D. Delay Performance of MRBS Multiplexer

A transmission may experience a scheduling delay in MRBS multiplexer due to the resource sharing and conflict resolution. The delay is addressed below.

Proposition 1 *For any input $A_i(n) \sim$ MRBS(p_i), the packet delay due to transmission scheduling by an MRBS multiplexer is upper bounded by $\lceil 1/p_i \rceil$ packet durations.*

Proof: Since a buffer of size '1' analogous to the size of one packet suffices to avoid overflow, buffered packets of an arrival stream will be served before the accumulation of coming ones. Therefore, the delay is at most equal to the duration between two successive '1's in an MRBS. Including packet arriving time in delay measurement, the result is always less than or equal to $\lceil 1/p_i \rceil$ units. □

Furthermore, as there is no early transmission according to MRBS-SSA, we have the following delay bound.

Proposition 2 *Let $D_{Mux}(p_i)$ be the packet delay of arrival $A_i(n) \sim$ MRBS(p_i) due to an MRBS multiplexer, we have*

$$0 \leq D_{Mux}(p_i) \leq \lceil 1/p_i \rceil. \tag{6}$$

Next, we can measure the delay jitter, which is defined as the difference of the maximum delay to the minimum one [26]. The bound on jitter is given below.

Proposition 3 *Let $J_{Mux}(p_i)$ be the delay jitter of arrival $A_i(n) \sim$ MRBS(p_i) due to an MRBS multiplexer, we have*

$$J_{Mux}(p_i) \leq \lceil 1/p_i \rceil. \tag{7}$$

As shown in [24], among all arrival sequences of a given rate, the MRBS is the optimal one which yields the smallest mean delay and all the moments in a queuing or scheduling system due to the most regular inter-arrival time. Here, a simulation study is conducted to investigate the packet delay due to different arrival processes or access patterns. The duration that a packet buffered from its arrival until departure in the scheduling system denoted by Fig. 3 is measured. This corresponds to the delay of modified transmission sequence or resultant packet streaming due to the access control or traffic scheduling in reference to the original one. We look at the mean and variance.

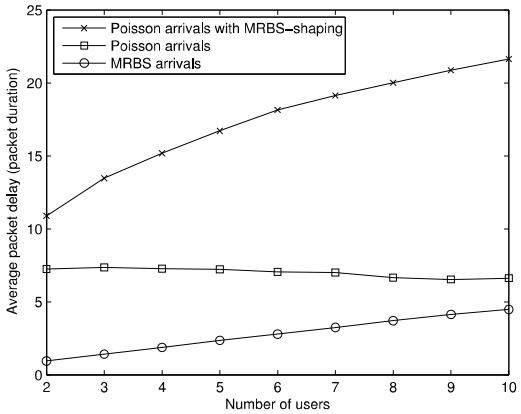

Figure 4: A comparison of the average delay.

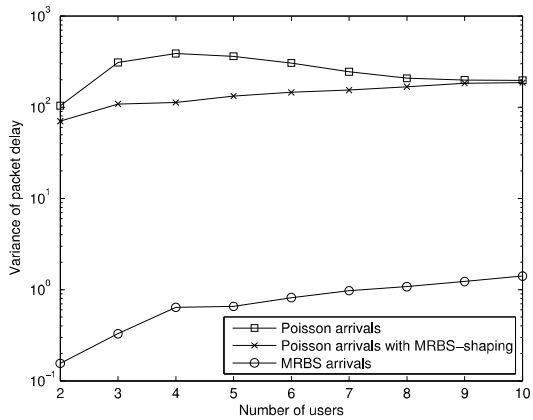

Figure 5: A comparison of the delay variance.

In simulation, the number of input streams varies from 2 to 10. In each case, p_i is drawn from a uniform distribution $(0, 1)$ and the generated p_i is then normalized to have the sum equal to 1. In this scenario, the scheduler works continuously for incoming traffics. Moreover, the transmission slot allocation has the same pattern for all arrival processes in simulation. This eliminates the impact on the comparison due to different service allocation pattern. Results for each case are obtained by 300 runs. The average is taken.

Packet delays from MRBS arrivals, Poisson arrivals, and Poisson arrivals with MRBS traffic shaping are compared. MRBS arrivals are generated by (1). Poisson arrivals of the same set of p_i are generated as well. As shown in Fig. 4, packets from MRBS arrivals have experienced a smaller average delay as compared to those from Poisson processes. The difference is particularly large when the number of users is relatively small, since in such scenarios the overall arrivals from Poisson processes tend to be much more bursty than those from the MRBS when the aggregate sum is kept as one in simulation.

Moreover, the average delay from Poisson arrivals with MRBS traffic shaping is shown in Fig. 4 as well. It indicates the average delay of packets from Poisson arrivals when an MRBS traffic regulator is inserted between the arrival process and scheduler. As a result, inputs to the scheduler will be regulated in the middle first. The plot corresponds to the end-to-end delay. Due to the additional traffic regulation and consequent buffering, it is reasonable that packets of Poisson arrivals with MRBS-shaping will suffer a larger average delay than those from the Poisson arrivals. However, the traffic shaping can result in a smaller delay variance as compared to the original one. Fig. 5 shows the result.

It should be pointed out that the impact of delay variance on stream QoS is often more significant than that of the delay mean, especially in a high-speed network since the service is usually quite sensitive to transit time variance. Since the MRBS process keeps in the most regular interarrival time, it yields a much more smooth traffic pattern, especially when compared to those bursty ones. As shown in Fig. 5, it has a much smaller delay variance as compared to that of the Poisson arrivals. On the other hand, by the results from Poisson arrivals with and without MRBS traffic shaping, it is shown that the one with MRBS-shaping suffers from a smaller delay variance as compared to that without MRBS-shaping in a trade-off of a larger delay mean.

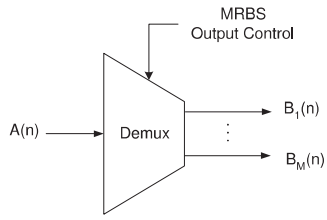

Figure 6: The model of an MRBS demultiplexer.

E. MRBS Demultiplexer

An ideal demultiplexer (DEMUX) is a network element which separates multiplexed signals from a transmission link and diverts them to proper end paths. An output control is necessary for traffic routing. To be precise, let $B_i(n)$ be the split streams of input $A(n)$, where $i = 1, 2, \ldots, M$,

$$\sum_{i=1}^{M} B_i(n) = A(n). \tag{8}$$

An MRBS demultiplexer is an ideal demultiplexer and provided that input $A(n) \sim \text{MRBS}(p_{\tilde{M}})$, the split output streams are also MRBS traffics such that

$$B_i(n) \sim \text{MRBS}(p_i) \tag{9}$$

where p_i is the proportion of output i among $A(n)$ and $\sum_{i=1}^{M} p_i = p_{\tilde{M}}$. A specific output control is necessary for the above demultiplexing objectives as shown in Fig. 6.

If the task is to simply split a source of identical packets of an amount corresponding to $p_{\tilde{M}}$ into a number of separated streams in MRBS patterns corresponding to p_i respectively without a requirement of any specific address or path mapping, one can find that such a packet demultiplexing is in fact similar to an MRBS traffic shaping with respect to p_i. A simple first-in first-out (FIFO) typed scheduling can serve the purpose. As long as $p_{\tilde{M}} = \sum_{i=1}^{M} p_i$ and the input and output links are on the corresponding speeds, there is no buffering requirement. It has no scheduling delay.

More specifically, we are interested in the packet demultiplexing of a merged source resulted from the MRBS multiplexer model as shown by Fig. 3 back to individual MRBS traffics. In this formulation, address mapping is involved. One may interpret the demultiplexer as a mirror of the multiplexer. It carries the reversed process of its multiplex-

ing counterpart. With the knowledge of an MRBS multiplexer, we can simply employ the facts of MRBS-SSA for the analysis of stream splitting in the demultiplexer and investigate from the results of an MRBS multiplexer. Note that for a source $A(n) \sim \text{MRBS}(p_{\tilde{M}})$ in the demultiplexer, given output requirement $B_i(n) \sim \text{MRBS}(p_i)$ as analogous to $A_i(n) \sim \text{MRBS}(p_i)$ in multiplexer, based on the knowledge of MRBS-SSA, the routing address of a packet in $A(n)$ is in fact known. As a result, the re-distribution of packets to each individual output i with respect to p_i is explicit in aid of MRBS-SSA. However, such an address mapping only directs packets to the correct output ends. In general, the resultant streams are not in MRBS patterns.

To achieve output $B_i(n) \sim \text{MRBS}(p_i)$, a transformation or traffic shaping of the routed result is necessary. With the information of p_i and MRBS definition, a processing on separated streams by the help of MRBS traffic regulator can obtain the result. Obviously, this may lead to a scheduling delay when comparing the input $A(n)$, which is the merged process $\text{MRBS}(p_{\tilde{M}})$, with the desired outputs $B_i(n)$. The delay experienced is equivalent to the packet-to-packet time difference between the desired MRBS output and the original pattern in merged source $A(n)$.

F. Delay Performance of MRBS Demultiplexer

The delay performance is addressed in the following.

Proposition 4 *Let $D_{Demux}(p_i)$ be the packet delay associated with p_i stream of an merged MRBS input $A(n) \sim$ MRBS($p_{\tilde{M}}$) due to the MRBS demultiplexing, we have*

$$0 \leq D_{Demux}(p_i) \leq \lceil 1/p_i \rceil. \tag{10}$$

Proof: As given by Proposition 1, for each MRBS arrival in an MRBS multiplexer, a buffered packet will be transmitted before the accumulation of the coming one. This implies that, on a time axis, a departed packet is located in the range of two successive '1's in the sequence and not later than the position of the second '1'. This is the scenario of time positions of consequently merged result in $A(n)$. With this property, in the case of an MRBS demultiplexer, by comparing the time position of a packet in $A(n)$, which is a departed packet from an MRBS multiplexer, with that in the scheduled output associated with an MRBS traffic shaping, the difference is bounded by the duration between two successive '1's as well. Hence, the packet delay associated with p_i stream is also upper bounded by $\lceil 1/p_i \rceil$. □

By (10), we have the following bound on the delay jitter.

Proposition 5 *Let $J_{Demux}(p_i)$ be the delay jitter associated with p_i stream of an merged MRBS input $A(n) \sim$ MRBS($p_{\tilde{M}}$) due to the MRBS demultiplexing, we have*

$$J_{Demux}(p_i) \leq \lceil 1/p_i \rceil. \tag{11}$$

IV. MRBS Scheduling in Multiple Node Scenarios

Extensions from single node MRBS scheduling to multiple node scenarios need to be addressed. Network operating under fixed routing strategy is considered [4]. In the previous section, isolated network elements of MRBS access and scheduling are defined and prepared for a generalized system modeling. In the following, discussions on the concatenation of network elements are given. The analysis offers an investigation on the deterministic bounds of packet delay with respect to the bandwidth allocation in a general communication system which may have multiple inputs, multiple outputs and various traffic schedulers.

A. Concatenation of Multiplexers

The concatenation of two MRBS multiplexers is shown in Fig. 7. Both the multiplexers MUX1 and MUX2 offer the scheduling service based on the MRBS-SSA. By Theorem 1, for inputs of $A_i(n) \sim \text{MRBS}(p_i)$, the merged output $B_j(n)$ from MUX1 is also an MRBS traffic. That is, $B_j(n) \sim \text{MRBS}(p_j)$. In the serial concatenation, $B_j(n)$ is driven as one of the inputs of MUX2. As long as the inputs to MUX2 are MRBS traffics, it is clear that the output $C(n)$ of the concatenation of two MRBS multiplexers is still an MRBS traffic. Each multiplexing requires only local information. By induction, we have the following conclusion.

Proposition 6 *A concatenation of MRBS multiplexers forms an MRBS multiplexer with aggregate rate equal to the sum of input rates.*

In Fig. 7, along the concatenation of multiplexers, packets from arrivals $A_i(n)$ will be multiplexed twice. Let $D_{Mux1}(p_i)$ and $D_{Mux2}(p_j)$ be the corresponding packet delays due to the first and second multiplexers respectively. It is interesting to see what is the resultant packet delay in such a concatenation. If the two multiplexers are isolated, by (6), we have the individual packet delay bounds of

$$0 \leq D_{Mux1}(p_i) \leq \lceil 1/p_i \rceil \tag{12}$$

and

$$0 \leq D_{Mux2}(p_j) \leq \lceil 1/p_j \rceil \tag{13}$$

respectively. In such a composition, since packets in stream $A_i(n)$ will first experience a delay due to MUX1 and then experience a delay due to MUX2, the delay sum which is denoted by $D_{Mux1+Mux2}$ can be expressed as

$$0 \leq D_{Mux1+Mux2} \leq \lceil 1/p_i \rceil + \lceil 1/p_j \rceil. \tag{14}$$

Let $D_{Mux(i)}$ be the packet delay due to MRBS multiplexer i, where $i = 1, 2, \ldots, M$, respectively. Denote the packet delay due to the concatenation of these MRBS multiplexers by $D_{\sum_{i=1}^{M} Mux(i)}$. Since there is not early transmission, the delay in such a series of MRBS multiplexers is equal to the sum of delays in each MRBS multiplexer. Consequently, we can have the following generalization.

Theorem 2 *The worst-case packet delay of a concatenation of MRBS multiplexers is equal to the sum of individual worst-case packet delays. The delay bound is given by*

$$0 \leq D_{\sum_{i=1}^{M} Mux(i)} \leq \sum_{i=1}^{M} \lceil 1/p_i \rceil \tag{15}$$

where $\lceil 1/p_i \rceil$ is the worst-case delay bound in each individual MRBS multiplexer respectively.

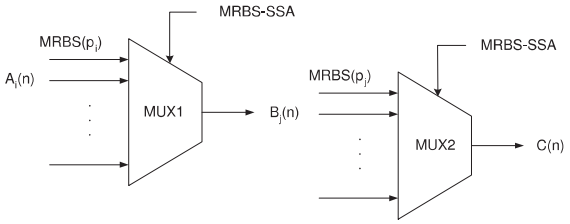

Figure 7: A concatenation of two MRBS multiplexers.

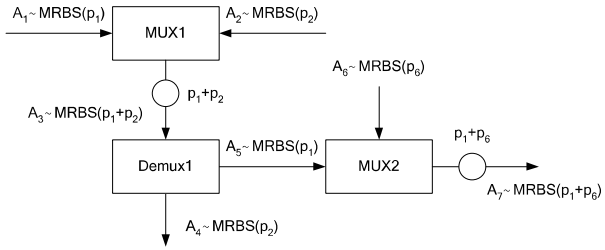

Figure 8: A network example for demonstration.

B. *Network Analysis*

Now, we will demonstrate in the following network example and illustrate how the previous ideas can be employed to have performance analysis in the MRBS scheduling. Fig. 8 shows the topology. There are two MRBS arrivals $A_1 \sim$ MRBS(p_1) and $A_2 \sim$ MRBS(p_2) to the MRBS multiplexer MUX1. Suppose that a link of capacity C corresponding to $(p_1 + p_2)$ is assigned for these two requests. Note that, in the resource allocation, we always try to keep with the minimum necessary bandwidth offering for the corresponding transmission scheduling. Based on the MRBS multiplexing, the merged output of MUX1, A_3, is still an MRBS traffic such that $A_3 \sim$ MRBS(p_1+p_2).

The MRBS demultiplexer (Demux1) refers to a splitting of the merged traffic $A_3 \sim$ MRBS$(p_1 + p_2)$. The two outputs of Demux1 are denoted by A_4 and A_5. The output A_4 from Demux1 represents a departure from the system. The output $A_5 \sim$ MRBS(p_5) is driven as one of the inputs to multiplexer MUX2. In a multiplexing of $A_5 \sim$ MRBS(p_1) and the new admission $A_6 \sim$ MRBS(p_6), we have the combined traffic such that $A_7 \sim$ MRBS$(p_1 + p_6)$.

In the above system, a rate-matched service control is maintained. Let us have a look at the delay bounds. By (6), the packet delays of A_1 and A_2 due to MUX1 are bounded by $\lceil 1/p_1 \rceil$ and $\lceil 1/p_2 \rceil$ respectively. The packet delay of A_3 stream in the demultiplexing due to Demux1 can be measured by (10). For packets of output A_5, the delay due to Demux1 is bounded by $\lceil 1/p_1 \rceil$. Look at MUX2, similarly, the scheduling delay on A_5 due to MUX2 is bounded by $\lceil 1/p_1 \rceil$. Therefore, the end-to-end delay of A_1 in the network is bounded by $3 \times \lceil 1/p_1 \rceil$. On the other hand, the delay experienced by A_6 is bounded by $\lceil 1/p_6 \rceil$.

The discussed example gives a demonstration on the idea of performance analysis in the MRBS scheduling with multiple network elements. Delay bounds are determined.

V. Applications and Discussions

In the previous sections, we have developed an analytical framework of the most regular scheduling in a systematic way. The performance and guarantees of flexible rate MRBS service are investigated. We have first studied the scheduling in terms of isolated network elements and then extended the result to an exploration of joint or composite modules. The MRBS scheduling can be applied to wireless access systems [25] and a wide variety of communication networks. Some examples are discussed below.

A. *Wireless Mesh Networks*

The idea of MRBS resource scheduling can be applied to a general class of multiple layered wireless networks with the nice property that the merging or splitting of most regular sequences still follows an MRBS pattern. This feature is adoptable and particularly useful in systems with a multiple layered transmission scheduling architecture [20–22]. For example, a three-tier hierarchical wireless sensor network is considered in [20] with intermediate relays as forwarding nodes to distant access points. Compared to a flat sensor network, a small number of higher-layer network elements with better computing and communication capabilities could greatly improve the overall system performance in the throughput, reliability and life time.

An application model is depicted in Fig. 9. In the two-tier wireless mesh network, users in each cluster are connected to a nearby wireless access node [22]. It is assumed that channel interference between users in different clusters is negligible. However, multiple access control in a shared channel is necessary. The intermediate nodes are connected with neighboring ones by wireless links as well. Suppose there is a network service of video conferencing between users A, B, C, D, E and F. In a particular streaming, A, B, C, D and E will send required data to F. For example, channels C1 and C2 are deployed as indicated in Fig. 9 for the purpose. In node 1, the channel access by A and B can be resolved and accommodated by MRBS scheduling. Traffics from A and B are merged as MRBS stream automatically.

In a selection of wireless link and routing path, nodes 1 and 2 connect to node 3 in the common channel C2. Here, user D may use C2 to deliver its stream to node 3 and at the same time to preserve C1. Transmission coordination for steams of different required rates from nodes 1 and 2, and user D via C2 can be managed by the MRBS multiplexing. Periodical transmission ordering can be determined in a relatively distributed way by each individual based on an information exchange of p_i. Note that the corresponding buffering requirement at intermediate nodes and transmission delay can be determined by Propositions 1-3 similarly.

Consequently, traffics of users A and B from node 1, C from node 2, and D from node 3 are multiplexed in node 3. By the MRBS multiplexing property, the combined traffic is always an MRBS traffic. Iteratively, nodes 3 and 4 can transmit to node 5 by channel C1 via an MRBS scheduling. At last, the combined traffic from users A, B, C, D and E are forwarded by node 5 to user F. This example gives a demonstration of MRBS transmission scheduling and flow control in the two-tier wireless network. Delay bounds of streams in the integrated traffics from different users can be determined respectively. Note that, in the above example, we have chosen to multiplex traffics from nodes 1 and 2 to node 3 by C2 first and then employ C1 to the destination

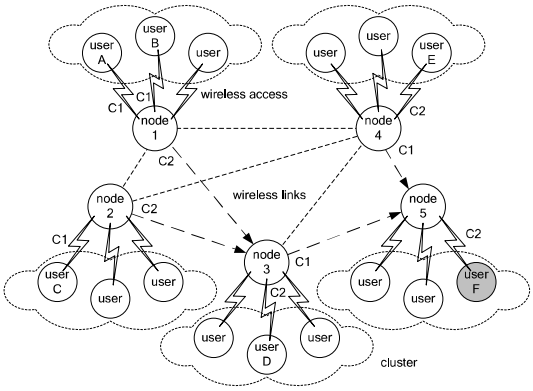

Figure 9: A model of multiple layered wireless network.

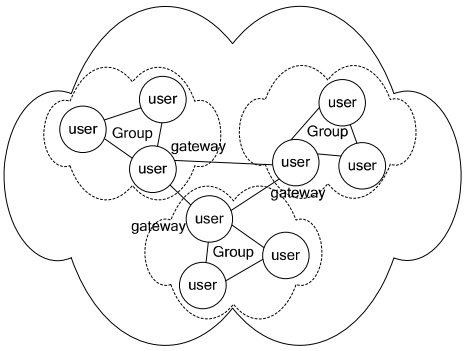

Figure 10: A two-layered hierarchical structure.

node 5. Generally, there exists other choices. For example, we may multiplex traffics from nodes 1 and 2 to node 4 first and then forward to the destination node 5. It may depend on the congestion control and routing strategy.

B. Communication Networks

The aforementioned wireless network can be summarized in a more generic two-layered hierarchical communication system as shown in Fig. 10. Traffic multiplexing and resource sharing in MRBS based scheduling is helpful by the advantage that concatenations of MRBS network elements keep their outputs in MRBS characteristics as well. The scheduling can lead to an automatic control. Moreover, network service guarantees can be determined analytically.

In a consideration of user mobility, the two-layered hierarchical structure can have the network management divided into two levels: (i) user-to-user cooperation locally within a group and (ii) gateway-to-gateway communication in a system level. Following MRBS scheduling, in systems where users in a group are usually in correlated movements for common task, frequent system-wide rearrangement is unnecessary since most re-configurations for adaptive control just need to be carried out locally and can be conducted easily by individuals in a group-wise manner.

VI. Conclusion

In this paper, we have investigated the MRBS scheduling for flexible rate-matched resource sharing in a generalized modeling. It is motivated by the most regular sequence with its optimal queuing delay properties. The service allocation is coordinated following the MRBS-SSA in a model of strict buffering constraint. Excessive delay is avoided. Meanwhile, systematic and deterministic performance guarantees can be provided in the established analytical framework. Studies on the delay are reported accordingly. Potential applications in communication systems are introduced particularly in a wireless mesh network. The multiple access control can be carried out in a relatively distributed way with limited information exchange.

References

[1] F. P. Kelly, *Reversibility and Stochastic Networks*, Wiley, 1979.

[2] J. Turner, "New directions in communications (or Which way to the information age)," *IEEE Commun. Mag.*, vol. 24, pp. 8–15, 1986.

[3] R. L. Cruz, "A calculus for network delay, part I: network elements in isolation," *IEEE Trans. Info. Theory*, vol. 37, pp. 114–131, 1991.

[4] R. L. Cruz, "A calculus for network delay, part II: network analysis," *IEEE Trans. Info. Theory*, vol. 37, pp. 132–141, 1991.

[5] A. K. Parekh and R. G. Gallager, "A generalized processor sharing approach to flow control in integrated service networks: the single-node case," *Proc. IEEE INFOCOM*, vol. 2, May 1992.

[6] A. K. Parekh and R. G. Gallager, "A generalized processor sharing approach to flow control in integrated services networks: the multiple-node case," *Proc. IEEE INFOCOM*, vol. 2, Mar. 1993.

[7] H. Zhang, "Service disciplines for guaranteed performance service in packet-switching networks," *Proc. IEEE*, vol. 83, no. 10, Oct. 1995.

[8] L. Georgiadis, R. Guerin, V. Peris, and K. N. Sivarajan, "Efficient network QoS provisioning based on per node traffic shaping," *IEEE/ACM Trans. Networking*, vol. 4, no. 4, pp. 482–501, Aug. 1996.

[9] S. J. Golestani and S. Bhattacharyya, "A class of end-to-end congestion control algorithms for the Internet," *Proc. Int. Conf. Network Protocols*, Oct. 1998, pp. 137–150.

[10] R. L. Cruz, "SCED+: efficient management of quality of service guarantees," *Proc. IEEE INFOCOM*, vol. 2, Mar. 1998, pp. 625–634.

[11] H. Sariowan, R. L. Cruz, and G. C. Polyzos, "SCED: a generalized scheduling policy for guaranteeing quality-of-service," *IEEE/ACM Trans. Networking*, vol. 7, no. 5, pp. 669–684, Oct. 1999.

[12] B. Hajek, "A basket of system theoretic problems in communications," *Proc. Int. Symposium on Mathematical Theory of Networks and Systems*, Notre Dame, Indiana, Aug. 2002.

[13] J. Y. Le Boudec and P. Thiran, *Network Calculus: A Theory for Deterministic Queueing Systems for the Internet*, Springer, 2001.

[14] J. P. Quadrat, "Max-plus algebra and applications to system theory and optimal control," *Proc. Int. Congress of Mathematicians*, 1994.

[15] J. Y. Le Boudec and P. Thiran, "Min-plus system theory applied to communication networks," *Proc. Int. Symp. Mathematical Theory of Networks and Syst.*, Notre Dame, Indiana, Aug. 2002.

[16] D. Gamarnik, "Using fluid models to prove stability of adversarial queueing networks," *IEEE Trans. Automatic Control*, Apr. 2000.

[17] C. S. Chang, *Performance Guarantees in Communication Networks*, New York: Springer, 2000.

[18] B. Hajek, "Large bursts do no cause stability," *IEEE Trans. Automatic Control*, vol. 45, pp. 116–118, 2000.

[19] L. Zhang and T. T. Lee, "Dynamic packet scheduling for wireless channel with varying capacity," *Proc. IEEE VTC*, Apr. 2003.

[20] L. Sankaranarayanan, G. Kramer, and N. B. Mandayam, "Hierarchical sensor networks: capacity bounds and cooperative strategies using the multiple-access relay channel model," *Proc. IEEE SECON*, 2004.

[21] I. F. Akyildiz, X. Wang, and W. Wang, "Wireless mesh networks: a survey," *Computer Networks*, vol. 47, no. 4, pp. 445–487, 2005.

[22] Y. Yu and H. J. Chong, "A survey of clustering schemes for mobile ad hoc networks," *IEEE Commun. Surveys*, vol. 7, no. 1, 2005.

[23] J. Soldatos, E. Vayias, and G. Kormentzas, "On the building blocks of quality of service in heterogeneous IP networks," *IEEE Commun. Surveys*, vol. 7, no. 1, pp. 70–89, First Quarter, 2005.

[24] B. Hajek, "Extremal splitting of point process," *Math. Oper. Res.*, vol. 10, no. 4, pp. 543–556, Nov. 1985.

[25] C. S. Chen and W. S. Wong, "Bandwidth allocation for wireless multimedia systems with most regular sequences," *IEEE Trans. Wireless Commun.*, vol. 4, no. 2, pp. 635–645, Mar. 2005.

[26] A. S. Tanenbaum, *Computer Networks*, Upper Saddle River, New York: Prentice Hall PTR, 1996, pp. 384–386.