A Conflict-Free Low-Jitter Guaranteed-Rate MAC Protocol for Base-Station Communications in Wireless Mesh Networks

T.H. Szymanski

Bell Canada Chair in Data Communications
Dept. ECE, McMaster University, Hamilton, ON Canada

Abstract. A scheduling algorithm and MAC protocol which provides low-jitter guaranteed-rate (GR) communications between base-stations (BS) in a Wireless Mesh Network (WMN) is proposed. The protocol can provision long-term multimedia services such as VOIP, IPTV, or Video-on-Demand. The time-axis is partitioned into scheduling frames with F time-slots each. A directional antennae scheme is used to provide each directed link with a fixed transmission rate. A protocol such as IntServ is used to provision resources along an end-to-end path of BSs for GR sessions. The Guaranteed Rates between the BSs are then specified in a doubly stochastic traffic rate matrix, which is recursively decomposed to yield a low-jitter GR frame transmission schedule. In the resulting schedule, the end-to-end delay and jitter are small and bounded, and the cell loss rate due to primary scheduling conflicts is zero. For dual-channel WMNs, the MAC protocol can achieve 100% utilization, as well as near-minimal queueing delays and near minimal delay jitter. The scheduling time complexity is $O(NFlogNF)$, where N is the number of BSs. Extensive simulation results are presented.

Keywords: scheduling, multihop, mesh, networks, low jitter, quality of service.

1 Introduction

Emerging multihop *Wireless Mesh Networks* (WMNs) represent a key opportunity to deploy wireless broadband services in a relatively inexpensive manner [1][2]. A multihop WMN consists of a collection of geographically-fixed wireless mesh routers and gateways, called Base-Stations (BSs), which provide wireless access to the global Internet network, as shown in Fig. 1 [1]. A WMN can provide broadband access to both fixed residential users and to mobile users. This paper presents a collision-free low-jitter guaranteed-rate (GR) scheduling algorithm and MAC protocol for inter-BS communication in an infra-structure-based WMN. The protocol supports the efficient delivery of multimedia services such as VOIP, IPTV, and Video-on-Demand (VOD).

WMNs can utilize the emerging IEEE 802.16 WiMAX network standard [3]. The WiMAX physical layer provides data rates between 32 and 130 Mbps, given appropriate physical link parameters, including the available channel spectrum, modulation scheme, signal constellation and distance. The standard exploits OFDM technology to enable a

C. Wang (Ed.): AccessNets, LNICST 6, pp. 118–137, 2009.

high utilization of the available spectrum between any pair of nodes. A multi-hop WMN using WiMAX technology can potentially offer both high access rates while spanning distances of tens of kilometers.

Each BS manages a physical region called a *wireless cell*, which contains multiple *Stationary Subscriber Stations* (SSs) and multiple *Mobile Subscriber Stations* (MSs), as shown in Fig. 1a. The SSs communicate with the BS for access to the global Internet. A multi-hop WMN can be described as a graph, where BSs are represented by vertices, and where radio links between BSs are represented by directed links, as shown in Fig. 1b. The communications within a WMN can be viewed at two levels; (1) the communications within a cell between the BS and the SSs and occasionally between the SSs, and (2) the communications between BSs and the gateway BS.

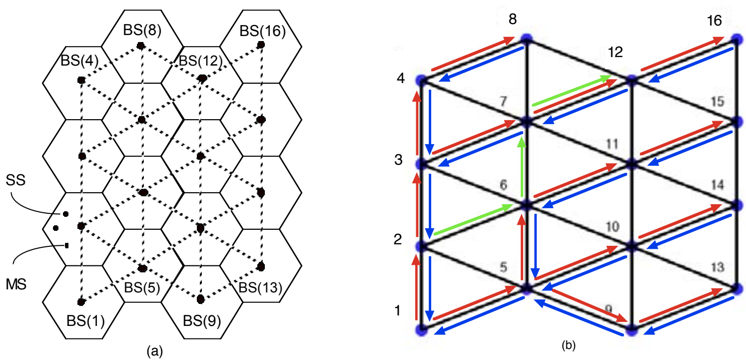

Fig. 1. (a) Hexagonal WMN. (b) Graph Model

This paper addresses the scheduling and MAC protocol between BSs. Assume an infrastructure-based multi-hop WMN, where the physical locations of the BSs are fixed. All BSs have smart antennae arrays and can implement beamforming algorithms on the IO channels, as in [2]. Beamforming antennas will allow for increased transmission rates between selected neighbors and lower interference among non-selected neighbors, which will increase the capacity of a multi-hop WMN. In Fig 1, all link performances are affected by the weather, and they can be optimized periodically by recomputing the physical link parameters, such that each directed link achieves the required data rate and SNR ratio. A similar model was proposed in [2].

The multi-hop nature of WMNs leads to several technical challenges. Capacity and scalability are critical requirements for WMNs. To increase capacity, wireless routers can exploit multiple wireless transceivers, exploiting multiple orthogonal radio channels. However, the design of routing and scheduling algorithms for such networks is challenging. According to [4], '*scheduling link transmissions in a wireless network so as to optimize one or more performance objectives has been the topic of much interest over the past several decades*'. According to a recent survey article [1], '*These advanced wireless radio technologies all require a revolutionary design in higher-level protocols, especially MAC and routing protocols*'. Currently, there are no scheduling algorithms /MAC protocols for single or multi-channel WMNs which have low computational complexity, whichachieve 100% throughput and which achieve near-minimal queueing delays, near-minimal delay jitter and near-optimal QoS.

Tassiulas and Ephremides [5,6,7] first considered the problems of routing and scheduling in multihop networks and other *constrained queueing systems*. The BSs typically exploit *time division duplexing* (TDD). The time-axis consists of many physical time-slots, and a BS with a single radio transceiver can either transmit or receive during one time slot. Two types of conflicts can occur [2,5,6,7]. A *primary* conflict occurs when the number of active directed links incident to one BS exceeds the number of radio transceivers available at that BS. To avoid primary conflicts, the MAC protocol must ensure that the number of active directed links per BS in every time-slot does not exceed the number of radio transceivers. A *secondary* conflict occurs when the signal power from remote nodes interferes with the signal power of the intended receiver. To mitigate secondary conflicts, smart antenna arrays and power adjusting algorithms can be used [2].

The design of interference-free schedulers for mesh networks is a hard problem [2]. References [8,9] establish that the problem of finding schedules with optimal throughput under a general multi-hop WMN interference model is NP-hard. They also establish the difficulty of scheduling and state: '*we assume the packet transmissions at the individual nodes can be finely controlled and carefully scheduled by an omniscient and omnipotent central entity, which is unrealistic*'. Many recent papers propose a joint-layer design methodology, which considers multiple layers simultaneously, ie, the physical layer, the network routing layer, and the MAC layer [6,12-17]. Parameters from each layer are thereby exposed and can be used to express an optimization problem over many layers, which can then be solved. Reference [13] decouples the routing and scheduling problems and presents polynomial time approximation algorithms for routing and scheduling, the latter relying upon graph coloring. The approach taken in this paper decouples the physical, routing and the scheduling/MAC layer optimizations. The goal of the physical layer is to provide directed links with a fixed data-rate. The goal of the routing layer is to route the end-to-end traffic flows such that no capacity constraints are violated. The goal of the scheduling algorithm and MAC layer is to allocate access to the shared wireless medium in a throughput-optimal manner, while simultaneously striving for near-perfect QoS.

In this paper, we propose a low-jitter guaranteed-rate scheduling algorithm and MAC protocol for conflict-free communications between BSs in a WMN. A resource reservation algorithm such as IntServ or DiffServ is used to reserve resources (ie buffer space and bandwidth) along an end-to-end path of BSs, for long-term multimedia flows. The bandwidth demands between all BSs in the WMN are then be specified in a doubly stochastic NxN traffic rate matrix, where N is the number of BSs. This traffic rate matrix is decomposed to yield a low-jitter frame transmission schedule, using a recursive fair stochastic matrix decomposition algorithm presented in [33,34,35]. This algorithm will yield a sequence of partial or full permutations which form a *frame transmission schedule*. For dual-channel WMNs, the frame transmission schedule can achieve 100% throughput and directly yields a conflict-free set of transmitting BSs denoted $T(j)$ and a set of receiving BSs denoted $R(j)$ for each time-slot j in the frame, where $1 \le j \le F$. For single-channel WMNs, some post-processing of the permutations is required to achieve a conflict-free schedule.

In the resulting schedule, all primary conflicts are avoided, all guaranteed-rate traffic demands are met, and the delay jitter between any cells in a traffic flow is small and bounded by *K*IIDT* time-slots for fixed constant K, where an *IIDT* denotes the ideal number of time-slots between successive cells in a provisioned flow. Given the small jitter bound, several important properties can be shown to hold [39]: (a) the expected number of cells belonging to one GR multimedia flow which are queued in any BS is near-minimal and bounded, and is typically < 2 cells per flow, (b) the expected end-to-end delay of a flow along a path of BSs with H-hops is near-minimal and bounded, and is typically < *2H*IIDT* time-slots, (c) the cell drop rate due to scheduling conflicts is zero, and (d) all network-introduced delay-jitter can be removed using a playback queue with a depth of *O(K)* cells. For dual channel WMNs with doubly stochastic traffic rate matrices, in addition to achieving near-minimal queuing delays and near-minimal jitter for provisioned multimedia traffic flows, the proposed scheduling algorithm and MAC protocol are *throughput-optimal*, ie they can achieve 100% of the allowable throughput, given an admissible routing of flows.

In our WMN model, bursty multimedia traffic such as IPTV will be transmitted over an end-to-end path with a provisioned GR. Large video-frames are partitioned into fixed-size cells, which are inserted into the WMN at the maximum provisioned rate, which will introduce an application-specific queuing delay and delay jitter for video frames at the source node (which is external to the network), as in [18,19]. This application-specific delay jitter is external to the network and can be filtered out using application-specific playback buffers at the end-users. However, all network-introduced delay jitter can be provably removed from consideration. To the best of our knowledge, this is the only known throughput-optimal scheduling algorithm and MAC protocol which yields bounded queue sizes and bounded queueing delays within general WMNs with doubly-stochastic traffic rate matrices, and which can remove all network-introduced delay jitter.

Section 2 describes constrained queueing systems, scheduling in WMNs, scheduling in Input-Queued (IQ) crossbar switches, and the WiMax technology. Section 3 describes the transformation from a scheduling problem in IQ switches to a scheduling problem in WMNs, presents the methodology to specify the traffic rate matrix between BSs in a WMN, and describes the proposed algorithm to solve the scheduling problem. Section 4 describes a typical hexagonal WMN and illustrates experimental results. Section 5 contains concluding results.

2 Problem Formulations

2.1 Constrained Queueing Systems

Tassiulas and Ephremides developed the concept of a *constrained queueing system*, a network of queues where the servers in a constrained set cannot provide service simultaneously [5,6,7]. This model can describe multihop wireless networks, database systems, parallel processing systems, and IQ switches. The *stability region* was defined as the set of flow traffic rate vectors which yield a stable system. They established that given an admissible traffic flow rate vector, a *dynamic scheduling algorithm* which solves a *Maximum Weight Matching (MWM)* problem on a bipartite graph for each time-slot will yield a stable system. The *MWM* will define a *transmission set* for each time-slot, ie a set of simultaneously enabled servers which do not violate the interdependency constraints. They also established that the *MWM* algorithm is a *throughput-optimal* solution, when the edge weights reflect the queue lengths in the system, such that the *MWM* scheduling maximizes throughput and minimizes the backlog of traffic in each time-slot.

The problem of determining whether a flow traffic rate vector is admissible in a multi-hop WMN is NP-hard, since it entails 2 phases: (1) the routing phase, ie determining whether the flows can be routed such that no resource is overloaded, and (2) the scheduling phase, the scheduling of cells for transmission for each multiclass queue, once the flows have been routed. The routing problem is known to be NP-hard in the general case. For a multi-hop interference model with secondary interference, scheduling in a WMN is NP-hard [8,9], while under a 1-hop interference model with primary interference, a throughput-optimal solution can be found in polynomial time using the dynamic *MWM* scheduling [5]. Tassiulas recognized that the solution of a *MWM* algorithm for each time-slot of a WMN may not be practical. He also presented a randomized iterative algorithm with linear time complexity for achieving stability and optimal throughput in the capacity region [6].

It is now well known that use of dynamic scheduling algorithm relying on computation of a *MWM* in each time-slot will achieve 100% throughput [4,5,21], but it will also result in queue sizes approaching several thousand cells [21,22,23]. In [22], a dynamically tunable control algorithm was proposed to improve the performance of existing (sub-optimal) dynamic WMN scheduling algorithms. The time allocated for any heuristic scheduling algorithm is tunable, such that more time can be used to obtain a better solution. Simulations presented in [22] indicate that typical existing algorithms lead to queue sizes of between 2,000 and 18,000 cells for loads approaching 92 percent of capacity, and that their tunable algorithm yields queue sizes of 100-800 cells for loads approaching 92 percent of capacity. However, their algorithm becomes unstable for larger loads, and the queue sizes approach infinity.

Reference [4] defined a *k*-hop interference model for WMNs, and it was shown that for a given *k* a throughput-optimal dynamic scheduler needs to solve a *MWM* problem given the *k*-hop interference constraints in each time-slot, consistent with the work of Tassiulus [5]. They formulate the scheduling problem as an optimization problem. For *k*=1, they also propose a greedy *Maximal Matching* (*MM*) scheduling algorithm which provably achieves 50% of the optimal throughput in the capacity region. The importance of the *MWM* problem to dynamic scheduling is thus well established.

Referring to the WMN in Fig. 1b, BS(1) is the gateway BS which has access to the wired Internet. A WMN can be represented as a directed Graph $G=\{V,L\}$, where $V=\{1,2,\ldots,N\}$ is the set of labels for all the BSs and where $L=\{1,2,\ldots,|L|\}$ is the set of labels for all the directed links [2]. Define the topology matrix E where $E \in R^{N \times L}$ as

$$E(n,l) = \begin{cases} 1 & \text{if BS(n) is the transmitter of link l} \\ -1 & \text{if BS(n) is the receiver of link l} \\ 0 & \text{otherwise} \end{cases}$$

Extending the formalism in [2], let $I(n)$ denote the set of incoming links to BS(*n)*, and let $O(n)$ denote the set of outgoing links from BS(*n)*. Define the 0-1 incidence matrix *I* as

$$I(l,n,t)=\begin{cases}1 & \text{if link l is active at time-slot t and is incident to/ from node n}\\ 0 & \text{otherwise}\end{cases}$$

In a single-channel WMN, primary conflicts are avoided when the following condition is met:

$$\sum_{u\in O(n),v\in I(n)} I(u,n,t)\cdot I(v,n,t)=0,\quad n=1,2,\ldots,N,\ t=1,2,\ldots F$$

In a multi-channel WMN where each BS has *R* radio transceivers and can transmit or receive on up to *R* orthogonal channels during one time-slot, primary conflicts are avoided when the following condition is met:

$$\sum_{u\in O(n),v\in I(n)} I(u,n,t)+I(v,n,t)\le R, n=1,\ldots,N,\ t=1,\ldots F$$

In our infra-structure based WMN model, secondary conflicts can be mitigated by optimizing the physical link parameters (beamforming antennae weights, modulation scheme, signal constellation, and transmission power) once the WMN topology has been fixed. When an ISP is deploying the WMN, the physical positioning of the base-stations is a variable that the network administrators can adjust to eliminate or reduce secondary conflicts. Once the positions are fixed, the antennae weights can be optimized to achieve a constant datarate over every directed link.

Define a transmission set *TS(t)* as a subset of directed links which is free of primary and secondary conflicts during a given time-slot *t*. Any subset $T'(t)\in TS(t)$ is also a valid transmission set. A *maximal* transmission set is defined as one which is not a subset of any other transmission set [2,5]. The goal of this paper is to describe a scheduling algorithm which finds maximal transmission sets for each time-slot such that throughput is optimized, and which simultaneously achieves near-perfect QoS for every statically provisioned traffic flow in the WMN.

2.2 Scheduling in IQ Crossbar Switches

An input-queued (IQ) crossbar switch is shown in Fig 2a, and an output-queued (OQ) crossbar switch is shown in Fig 2b. An IQ switch is one example of the *constrained queueing systems* proposed by Tassiulas and Ephremides [5]. An *NxM* IQ crossbar switch has *N* input and *M* output ports, for which a traffic rate matrix can be specified. Each input port *j* $0\le j<N$ has *M* Virtual Output Queues (VOQs), one for each output port *k*, $0\le k<M$. The GR traffic requirements for an $N\times N$ crossbar switch can specified in a doubly substochastic or stochastic traffic rate matrix Λ:

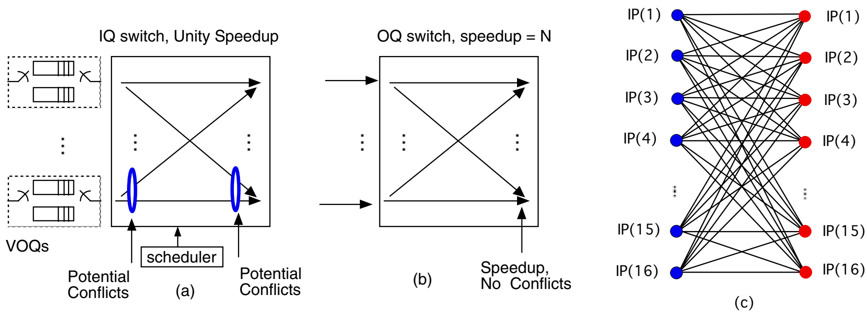

Fig. 2. (a) IQ crossbar switch. (b) OQ crossbar switch. (c) Bipartite graph model.

$$\Lambda = \begin{pmatrix} \lambda_{0,0} & \lambda_{0,1} & \cdots & \lambda_{0,N-1} \\ \lambda_{1,0} & \lambda_{1,1} & \cdots & \lambda_{1,N-1} \\ \cdots & & & \cdots \\ \lambda_{N-1,0} & \lambda_{N-1,1} & \cdots & \lambda_{N-1,N-1} \end{pmatrix}, \quad \begin{matrix} \sum_{i=0}^{N-1} \lambda_{i,j} \le 1, \\ \sum_{j=0}^{N-1} \lambda_{i,j} \le 1 \end{matrix}$$

Each element $\lambda_{j,k}$ represents the fraction of the transmission line rate reserved for guaranteed traffic between IO pair *(j,k),* equivalently *VOQ(j,k)*. The transmission of cells through the IQ switch is governed by the *frame transmission schedule*. In a schedule with *F* time-slots per frame, the minimum amount of reservable bandwidth is one time-slot reservation per frame on a recurring basis, which guarantees the fraction *1/F* of the line rate. Define a new quantized traffic rate matrix *R* where each rate is represented as an integer number of time-slot reservations per frame:

$$R = \begin{pmatrix} R_{0,0} & R_{0,1} & \cdots & R_{0,N-1} \\ R_{1,0} & R_{1,1} & \cdots & R_{1,N-1} \\ \cdots & & & \cdots \\ R_{N-1,0} & R_{N-1,1} & \cdots & R_{N-1,N-1} \end{pmatrix}, \quad \begin{matrix} \sum_{i=0}^{N-1} R_{i,j} \le F, \\ \sum_{j=0}^{N-1} R_{i,j} \le F \end{matrix}$$

One challenge when scheduling IQ crossbar switches [20,21,23-35] is resolving the conflicts which occur at the input and output ports. A conflict-free permutation which maps the input ports onto the output ports must be found for each time-slot. The problem of scheduling transmissions in IQ switches so as to optimize one or more performance objectives has been the topic of much interest over the past five decades. A related problem on time-slot-assignments was first considered by Slepian and Duguid, Benes and Clos in the 1950s [20]. See [34] for a brief history of the IQ switch scheduling problem.

It has been established that a dynamic scheduling algorithm for IQ crossbar switches which formulates and solves a **MWM** problem in each time-slot can achieve stability and 100% throughput [21], consistent with the conclusions of Tassiulas and Ephremides on generalized constrained queueing systems [5]. Simulations of many dynamic scheduling algorithms based upon **MWM** are presented in [22,23]. The number of queued cells in the steady-state will grow to several thousand cells as loads approach 100%. Not only is the complexity of *MWM* too excessive for practical use, the delay and jitter are very large such that QoS cannot be guaranteed. One approach to achieve better QoS in constrained queueing systems such as IQ switches is formulate the *MWM* problem to be solved in each time-slot such that the edge weights reflect the '*lag*' of an IQ switch, relative to an OQ switch with near ideal QoS [23]. The QoS of this scheduler is indeed better, but the computational complexity of the *MWM* is still a problem and the steady-state queue sizes, while bounded, can still reach several thousand cells at high loads [23]. As a result, there has been a considerable renewed interest into finding scheduling algorithms for IQ switches which are throughput optimal and which have low jitter and guaranteed QoS performances. Many recent algorithms rely upon stochastic matrix decompositions [24,26-30].

The problem of finding a perfect zero-jitter schedule for IQ switches with bounded speedup has been shown to be NP-hard [26,27]. Those authors present a polynomial time scheduler based upon *Greedy Low Jitter Decomposition* which achieves reasonably low jitter for loads near 80%, while requiring a worst-case speedup of *O(logN)*.

A stochastic matrix decomposition algorithm which attempts to bound the jitter in IQ switches was proposed in [24]. A traffic rate matrix is quantized and decomposed into a convex set of permutation matrices and weights, which are then scheduled. With speedup $S=1+sN$ between 1… 2, the maximum '*Service Lag*' over all *IO* pairs is bounded by $O((N/4)(S/(S-1)))$ time-slots. The speedup affects the QoS provided by the switch. According to [24]: "*with a fairly large class of schedulers a maximum service lag of* $O(N^2)$ *is unavoidable for input queued switches. To our knowledge, no scheduler which overcomes this* $O(N^2)$ *has been developed so far. For many rate matrices, it is not always possible to find certain points in time for which the service lag is small over all I-O pairs simultaneously*". For a speedup approaching 2 the service lag does not exceed approximately *N/2* time-slots, whereas is can go as large as $O(N^2)$ time-slots when no speedup is allowed.

A greedy stochastic matrix decomposition algorithm was also proposed in [30]. The algorithm cannot guarantee 100% throughput or short-term fairness. The authors establish a jitter bound, but their bound grows as the IQ switch size *N* increases. The authors identify an open problem: "*to determine the minimum speedup required to provide hard guarantees, and whether such guarantees are possible at all*". The problem of finding low-jitter schedulers with low speedup requirements is difficult.

A low-jitter scheduling algorithm based on recursive fair stochastic matrix decomposition was introduced in [33,34,35]. The algorithm is throughput-optimal, ie it can achieve 100% throughput and will also bound the delay jitter on every competing traffic flow to a small number of *IIDTs*, while requiring unity speedup. Proofs establishing very low delay and jitter are presented in [34]. A custom network simulator with over 20,000 lines of code was developed to gather detailed jitter statistics, to complement the theory. An exposition of very low jitter in a linear chain of IP routers is presented in [37]. An exposition of very low jitter in a multi-hop Fat-Tree mesh network with 256 simultaneous competing traffic flows is presented in [38]. Mathematical bounds on the end-to-end queuing delay and jitter are established in [39]. In this paper, this low-jitter algorithm will be adapted to the problem of scheduling traffic in a WMN, bounds on the delay and jitter are stated and detailed simulation results are presented.

2.3 The WiMAX Technology

The WiMAX standard defines 3 operation modes; (i) point-to-multipoint (PMP), (ii) centralized mesh mode (CMM), and (iii) distributed mesh mode (DMM) [3]. A WiMAX WMN system consists of BSs (Mesh Base Stations - MBS) that are connected to the wired IP network and which manage communications between the SSs in a wireless cell. In the PMP and CMM modes, the time axis is divided into scheduling frames, each consisting of several time-slots.

WiMax allows for both centralized and distributed scheduling [3]. In the centralized scheme the transmission schedules for all SSs within a cell are made by the centralized gateway BS. The scheduling information is distributed to the other SSs within a wireless cell using the '*Mesh Centralized Scheduling*' (MCS) control messages on the control channel. Statistics on the traffic loading can be sent from the SSs back to the gateway BS using the same MCS control messages. Therefore, a BS can act as the centralized scheduler which can precompute transmission schedules of all the SSs within the cell. We assume a similar model in a multi-hop WMN: the gateway BS can act as the centralize scheduler which can precompute transmission schedules for all other inter-BS communications within the WMN.

WiMax supports several service categories: (i) The Unsolicited Grant Service (UGS) which can be used for real-time constant-rate services, (ii) the real-time polling service (ttPS) suitable for IPTV multi-casting [3], (iii) the non-real-time polling service (nrtPS) suitable for video-on-demand (VOD) [3], and (iv) the best-effort service (BE). In this paper, all long-term multimedia traffic, such as VOIP, IPTV and VOD can be handled. The frame transmission schedules can be used to provide GR service in the UGS class for all these traffic types, or the schedule can be used to reserve bandwidth for a preferred polling order, to minimize cell jitter and queue sizes within base-stations, when the rtPS and nrtPS service classes are used.

In the WiMax standard, a TDD frame may have durations of 0.5, 1 or 2 milliseconds in the PMP mode, or 2.5, 4, 5, 8, 10, 12.5 or 20 milliseconds in the mesh mode. Typically, the physical parameters controlling the data-rate for each SS can be recomputed for each frame. In all physical layer transmissions, data bits are randomized for robustness, forward error correction is employed, and the modulation schemes, the signal constellations and the power transmission levels can be optimized. The modulation scheme and constellations include binary phase-shift keying (BPSK), quadrature phase shift keying (QFSK), and 16, 64 or 256-quadrature amplitude modulation (QAM). [3]

3 Transformation of IQ Switch Scheduling to WMN Scheduling

We describe the transformation for the IQ switch scheduling problem to the WMN scheduling problem. An *NxN* IQ switch has *N* input ports and *N* output ports. Each input port has *N* VOQs, and the IQ switch has N^2 VOQs. The switch also has N^2

internal links from the N^2 VOQs to the N output ports. An IQ switch has 2 sets of constraints, denoted *c1* and *c2*. Constraint (*c1*) requires that each set of N VOQs belonging to an input port transmits at most 1 cell per time-slot. Constraint (*c2*) requires that from each set of N VOQs associated with an output port, only 1 VOQ can be active per time-slot, so that each output port receives at most 1 cell per time-slot.

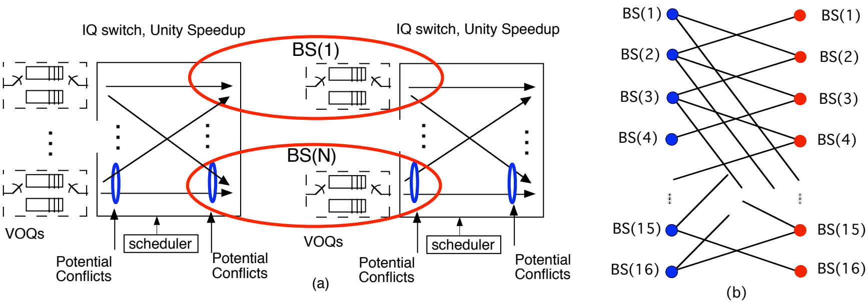

Fig. 3. (a) Transformation, IQ Switch to WMN. (b) Bipartite graph model.

The transformation from an IQ switch with N ports to a WMN with N Base-Stations is illustrated in Fig. 3. Fig. 3a illustrates a single IQ switch where the N external output links labeled *1..N* are fed back into the N input ports *1.. N* of the same switch (two IQ switches are drawn for simplicity). Each Base-Station j in the WMN equals the union of the output port j of the IQ switch, plus the N VOQs at input port j of the IQ switch, as shown in Fig 3a. The N^2 internal links of the IQ switch become the (up to) N^2 wireless links in the WMN, and each BS can be modeled as an Output-Queued (OQ) switch. The constraint (*c1*) of the IQ switch requires that each BS transmits at most 1 cell per time-slot. The constraint (*c2*) of the IQ switch requires that each BS receives at most 1 cell per time-slot. Effectively, the input (output) constraints on the IQ switch become the output (input) constraints on the WMN, respectively.

The traffic requirements to be met by an IQ switch can be specified in a doubly stochastic traffic rate matrix as established in section 2.2, and this matrix can be represented by a weighted bipartite graph. By the above transformation, the traffic requirements to be met by a WMN must also be specified by a doubly stochastic traffic rate matrix and weighted bipartite graph. The hexagonal WMN of Fig. 1 with N base-Stations can be represented as a bipartite graph, as shown in Fig 3b. Each BS has $\leq$ 6 incoming links and $\leq$ 6 outgoing links.

A frame-based low-jitter GR scheduling algorithm for IQ switches based on recursive fair stochastic matrix decomposition was proposed in [33,34,35]. A doubly substochastic or stochastic traffic rate matrix is first quantized to have integer values, and then is recursively decomposed in a relatively fair manner. Let *P(M,F)* denote the problem of scheduling an admissible quantized traffic rate matrix M into a transmission frame of length F time-slots. The problem *P(M,F)* is recursively decomposed into 2 problems *P(M1,F/2)* and *P(M2,F/2)*, such that matrices *M1+M2=M*, where *M1* and *M2* are admissible traffic rate matrices, and for all j and k where $0 \le j < N$ and $0 \le k < N$, $M1(j,k) \le M2(j,k) + c$ and $M2(j,k) \le M1(j,k) + c$ for constant c=1 or 2, depending upon assumptions. One step in an example decomposition for an 4x4 switch operating at 99.2% load with unity speedup is shown below:

$$\begin{bmatrix} 106 & 222 & 326 & 345 \\ 177 & 216 & 303 & 326 \\ 459 & 232 & 183 & 147 \\ 282 & 352 & 211 & 178 \end{bmatrix} = \begin{bmatrix} 53 & 111 & 163 & 172 \\ 88 & 108 & 152 & 163 \\ 230 & 116 & 91 & 74 \\ 141 & 176 & 105 & 89 \end{bmatrix} + \begin{bmatrix} 53 & 111 & 163 & 173 \\ 89 & 108 & 151 & 163 \\ 229 & 116 & 92 & 73 \\ 141 & 176 & 106 & 89 \end{bmatrix}$$

The low-jitter GR scheduling algorithm proposed in [33,34,35] for IQ crossbar switches bounds the service lead and service lag of each VOQ to $K \cdot IIDT$ time-slots for constant K, where *IIDT* represents the '*Ideal Inter-Departure Time*' for cells belonging to a VOQ. The bound applies to all VOQs in an IQ switch simultaneously. Furthermore, it can be shown that the bound applies to all cells in an end-to-end GR flow, provided that cells are selected for service within each VOQ along the end-to-end path according to a GPS scheduling algorithm. This algorithm will be applied to scheduling in WMNs. We now define several terms for the scheduling problem in WMN. These definitions are adapted from [33,34].

Definition: A “*Frame transmission schedule*” of length F is a sequence of partial or full permutation matrices (or vectors) which define the transmission sets and reception sets for a WMN, for F time-slots within a frame. Each transmission set identifies up to N conflict-free matchings of Base-Stations for service in the time-slot. The frame transmission schedule for a WMN with N BSs can be represented as an *FxN* matrix, where in each time-slot t for $1 \le t \le F$ up to N BSs are identified for transmission and reception. Given directed links with a capacity of L, the frame length F is determined by the desired minimum quota of reservable bandwidth, which equals L/F. To set the minimum quota of reservable bandwidth to $\le$ 1 % of L, set $F \ge 100$, *ie* $F = 128$.

Definition: A GR 'flow' in a WMN specifies the guaranteed traffic-rate between one origin BS and one destination BS. This traffic flow must be routed along an end-to-end path of BSs, in which buffer space is reserved in the queues of each BS and in which bandwidth is reserved in each inter-BS radio link in the path.

Definition: A “*Flow transmission schedule*” of length F is a sequence of vectors which define which flow to be serviced in each Base-Station, in the F time-slots within a frame, given a 'frame transmission schedule' which identifies the BSs to be serviced. The flow transmission schedule for a WMN with N BSs can be represented as an *FxN* matrix, where in each time-slot t for $1 \le t \le F$ a set of up to N flows is identified for service.

A flow transmission schedule can be computed from the frame transmission schedule; When a Base-Station receives service, select the flow to be serviced using the GPS algorithm. Based upon a flow transmission schedule, the following properties can be stated. Unfortunately, space constrains prevent the presentation of all results here and the proofs are established in [39].

Definition: At a given BS, the "*Inter-Departure Time*" (*IDT*) of a scheduled cell in a GR flow is defined as follows: Let the scheduled service time of cell *c* in flow *f* at the given BS be denoted *s(f,c)*. The *IDT* of cell *c* is defined as *s(f,c)-s(f,c-1)* for $c \geq 2$.

Definition: At a given BS, the "*Ideal Inter-Departure Time*" (*IIDT*) of scheduled cells in a GR flow *f* with quantized guaranteed traffic rate of $\phi(f)$ time-slot reservations per frame, given a frame of length *F*, a link datarate *L* in bytes/sec and fixed-sized cells of *C* bits, is given by: $IIDT(f) = F/\phi(f)$ time-slots, each of duration *(C/L)* sec.

Definition: At a given BS, the '*Received Service*" of a GR flow *f* with guaranteed rate $\phi(f)$ time-slot reservations per frame, at time-slot *t* within a frame schedule of length *F*, denoted $S_f(0,t)$, equals the number of matches in the frame transmission schedule in time slots *1...t,* where $t \leq F$, in which flow *f* is serviced.

Definition: At a given BS, the "*Service Lag*" of scheduled cell c of GR flow *f*, within a frame transmission schedule of length *F*, denoted $LAG_f(t)$, equals the scheduled service time of the cell minus the ideal service time of the cell, ie $LAGf(t) = s(f,cf) - c \cdot IIDT$. Intuitively, a positive Service Lag represents how many time-slots behind service the flow has fallen, relative to an ideal service schedule. A negative Service Lag is called a Service Lead, and represents how many time-slots ahead of service the flow has fallen.

The following four theorems are established in [39]. They assume each traffic flow is admitted into a network subject to shaping, and has a maximum service Lead/lag of *K IIDT*. Each BS is scheduled according to the proposed algorithm, with a maximum service lead/lag of *K IIDT* times-slots. The traffic rate matrix is doubly stochastic.

Theorem 1: Given a queue which receives a GR service of *C* cells per frame with a maximum Service Lead/Lag of $K \cdot IIDT$ time-slots, given an arriving traffic flow with a maximum Service Lead/Lag of $K \cdot IIDT$ time-slots, and given an initial queue state with $\leq O(K)$ cells, then the number of cells in the queue is upper bounded by $O(K)$ cells.

Theorem 2: When all queues in all intermediate nodes have reached steady-state, the maximum end-to-end queuing delay of a GR flow traversing *H* Base-Stations is $O(KH \cdot IIDT)$ time-slots.

Theorem 3: In the steady-state, a traffic flow *f* which leaves a Base-Station along an end-to-end path in a WMN will exhibit a maximum service lead/lag of $O(K \cdot IIDT)$ time-slots, ie the service lead/lag of a flow is not cumulative when traversing multiple Base-Stations. Equivalently, the delay jitter is not cumulative.

Theorem 4: A traffic flow which traverses *H* Base-Stations along an end-to-end path can be delivered to the end-user with *zero network-introduced delay jitter*, when a playback buffer of size $O(K)$ cells is employed.

3.1 Routing in a WMN

In order to specify the doubly stochastic traffic rate matrix to be decomposed, a set of GR flows to be provisioned in the WMN must be specified, and these flows must be routed through the WMN such that no constraints are violated. This section summarizes a typical routing problem formulation and describes how the traffic rate matrix is computed from the routing information.

Let F be the set of all traffic flows, denoted as (source,destination) pairs. Each flow f has a stationary unidirectional traffic rate r_f from the source to the destination nodes. Let P_f be set of all directed paths from the source to the destination nodes, available to carry the traffic required by flow f. Let p denote an individual path within the set P_f. Let x_p be the traffic rate in bits/second assigned along a path p. Let $\bar{\mathbf{x}}$ be the rate vector of all path flows $\left\{x_p \,\middle|\, f \in F,\ p \in P_f\right\}$. For an *admissible routing*, the rate vector must satisfy the two constraints: (a) for every flow f, the sum of the traffic rates over all paths in P_f must equal the specified traffic rate for the flow f, and (b) the traffic rate along any path must equal or exceed 0 (ie negative traffic rates are not allowed).

In our WMN model, assume every Base-Station i has a constraint on the sum of the incoming and outgoing traffic it can carry, which is denoted $C(i)$:

$$\sum_{(i,j)\in p,\, p\in P_f,\, f\in F} x_p + \sum_{(j,i)\in p,\, p\in P_f,\, f\in F} x_p \le C(i)$$

The first summation is the sum of all traffic leaving node i (for fixed i), over all edges (i,j), over all paths and flows. The second summation is the sum of all traffic entering node i, over all edges (j,i) for fixed i, over all paths and flows.

Given a set of paths available for each flow to be provisioned, a constrained optimization problem which minimizes the *unrouted traffic* in the WMN can be stated as follows:

$$\begin{aligned}
&\text{minimize} && \sum_{f\in F} r_f - \sum_{f\in F}\sum_{p\in P_f} x_p \\
&\text{subject to} && \sum_{p\in P_f} x_p = r_f \quad \text{for all } f \in F \\
& && x_p \ge 0 \qquad \text{for all } p \in P_f, \quad f \in F \\
& && \lambda_i \le C_i \qquad \text{for all } i \in V
\end{aligned}$$

where $\lambda_i = \sum_{(i,j)\in p,\, p\in P_f,\, f\in F} x_p + \sum_{(j,i)\in p,\, p\in P_f,\, f\in F} x_p$.

This problem can be solved in polynomial time using a linear programming approach. As observed by Tassiulas [5], determining an optimal routing has considerably more difficult. In the above optimization problem, the set of paths for each flow is fixed. However in an optimal formulation there are exponentially many paths and combinations of paths to be considered [5]. Fortunately, there are many polynomial time heuristic algorithms for determining good routings in general networks and WMNs. In this paper, we have decoupled the routing and scheduling problems, and we assume that an acceptable routing has been achieved. We focus on the scheduling problem.

Once an admissible routing for the GR traffic has been found, the traffic rate matrix for the WMN can be computed as follows:

$$
\begin{aligned}
&for\ f \in F\\
&\quad for\ p \in P_f\\
&\quad\quad for\ (u,v) \in p\\
&\quad\quad\quad M(u,v) = M(u,v) + r_f
\end{aligned}
$$

Once the traffic rate matrix is specified, it can be decomposed as described in [33,34], and the resulting frame transmission schedule yields the scheduling information for the WMN.

3.2 The Dual-Channel WMN

Each BS contains 2 radio transceivers which can simultaneously send and receive over 2 orthogonal channels. Therefore, the frame transmission schedule computed by the algorithm can be used directly to specify the conflict-free transmission sets of BSs in each time-slot. Link utilizations as high as 100 % for GR traffic can be achieved.

3.3 The Single-Channel WMN

Each BS contains 1 radio transceiver which can only receive or transmit during any one physical time-slot. Therefore, the frame transmission schedule computed by the algorithm cannot be used directly to specify the transmission sets in each time-slots.

To avoid primary conflicts, each permutation in the frame transmission schedule must be partitioned into multiple *(J)* sets of transmitting BSs denoted *T(1..J)* and *J* sets of receiving BSs denoted *R(1..J)*. Each BS can appear at most once as a source in one set *T* and at most once as a receiver in another set *R*. The following algorithm will partition a permutation into 3 conflict-free sets. Therefore, to find a frame-transmission schedule of length *F*=2048 time-slots, we may construct a traffic rate matrix for a smaller frame *F'*=512, and then expand each of the 512 permutations, to yield 1536 permutations, which will be partially utilized. For single-channel WMNs, each full permutation specifies at least 2 time-slots with *N* active BSs each to be realized. The partitioning algorithm specifies 3 time-slots with *2N* active BSs. Therefore, the maximum loading of GR traffic each single-channel BS is therefore 67%. The remaining 33% of the bandwidth can be used for BE traffic, which can be scheduled using the usual polling schemes.

```
Partition_Permutation(P)
T(1..3) = NULL;   R(1..3) = NULL;
for j=1:N {
     for k = 1:3 {
               u = j, v = P(j)
              if ( v ∩ T(k) = NULL ) and
                 u ∩ R(k) = NULL )) {
                        T(k) = T(k) ∪ u ;
```

$$R(k) = R(k) \cup v\,;$$
$$break; \quad \}\,\}\,\}$$

4 A WMN Example

4.1 A Communication Tree in a 16 Node WMN

In Fig. 2, 16 wireless cells are arranged in a conventional hexagonal mesh. This example is selected since the large number of directed edges will be difficult to schedule without conflicts. BS(1) at the lower left is the gateway BS with access to the wired IP network. Assume each directed link has bandwidth 128 Mbps, ie the physical link parameters are precomputed to ensure that each directed link can support a bandwidth of 128 Mbps. Since the BS locations are fixed and the physical environment (weather) is relatively static, these physical link parameters can be precomputed once and will only need to be updated when the physical environment (ie weather) changes.

Consider a downward communication tree with a root at BS(1) which delivers traffic to each BS (2..16). This tree is statically routed in the WMN, as shown by the blue arrows in Fig. 1b. Typically, trees are routed according to various optimization criteria, ie the tree may be a chosen to be minimum weight spanning tree rooted at BS(1). In this paper, the focus is the low-jitter GR scheduling algorithm and MAC protocol. Assume any traffic flows can be routed into the WMN according to any optimization criteria selected by the network administrator, as described in section 3. The proposed scheduling algorithm and MAC protocol will apply given any admissible routed communication pattern with a doubly stochastic traffic rate matrix.

An upward communication tree leading from each BS (2..16) and ending at BS(1) is also statically routed into the WMN in Fig. 1b. In Fig. 1b, the upward tree (ie red) follows the same topology as the downward tree, although this is not essential. In the tree in Fig 1b, we assume each BS adds 3 Mbps of bandwidth demand to the upward and downward tree. The embedding of any multicast trees into the WMN results in the specification of a traffic rate matrix for the WMN, which reflects the resulting loads on all the wireless links due to the multicast trees.

We assume additional point-to-point traffic between selected BSs is added, as represented by the green line in Fig 1b. In this example, the point-to-point traffic is generated between pairs of Base-Stations, to saturate every BS, ie every BS is essentially 100% saturated with traffic. This point-to-point traffic results in the specification of a second traffic rate matrix, which reflects the resulting loads on all the wireless links due to the point-to-point traffic.

The final traffic rate matrix is given by the sum of the multicast tree traffic rate matrix and the point-to-point traffic rate matrix described above. The resulting traffic rate matrix must be doubly substochastic or stochastic. This resulting matrix can then be decomposed to yield a set of F permutations or partial permutations, each of size N, as described in section 3. Each permutation specifies a set of up to N transmitting BSs, and a set of up to N receiving BSs per time-slot.

4.2 Experimental Results

Fig. 4a illustrates the observed normalized service lead/lag for every GR flow in the WMN of Fig. 1b, based upon the decomposition of the *16x16* traffic rate matrix derived in section 4.1, with *F=2048,* assuming the dual-channel WMN. The ideal service is represented by the bold diagonal line in Fig. 4a. Each single red line denotes the normalized service times observed for one GR flow which traverses its end-to-end path through the WMN. The individual red service lines for GR flows are indistinguishable, due to the large number of flows plotted on the same graph. However, the observed service closely tracks the ideal service. In Fig. 4a, the dashed green lines above and below the main diagonal correspond to service leads/lags of *3 IIDT* time-slots. The *X*-axis denotes the cell arrival time, expressed in terms of the IIDT for every flow *f*. The *Y*-axis denotes the cell number. The minimum and maximum Service Lead/Lags are visible from this graph. According to Fig. 4a, the observed Service Lead/Lags are within $K \cdot IIDT$ time-slots, as established in [33,34].

Fig. 4b plots the experimentally observed IDT PDF for cells leaving any BS, based upon the dual-channel WMN network model. According to theorem 3, all cells leaving a BS will exhibit a service lead/lag $\leq O(K \cdot IIDT)$ time-slots. Fig. 4b illustrates this property experimentally, ie the delay jitter remains bounded.

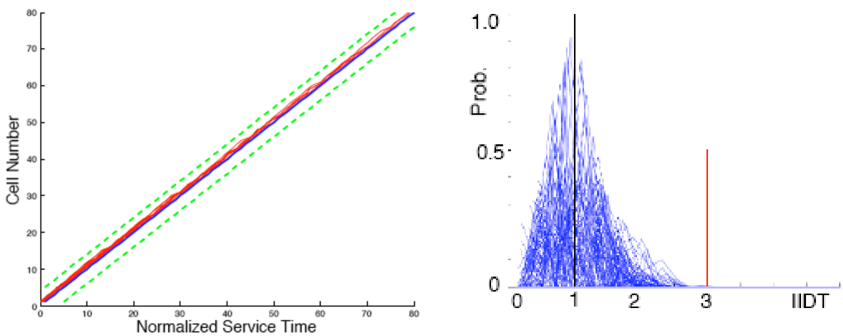

Fig. 4. (a) Service Lead/Lag over all Base-Stations. (b) IDT PDF over all Base-Stations.

The average and maximum number of cells per flow queued in every BS was also recorded by the network simulator. On average, every GR flow buffers less than 1 cell in each BS, indicating that each cell typically receives service within 1 *IIDT*, so that queuing is near-minimal. The maximum number of queued cells per flow in each BS is ≤ 6 cells, in this simulation, consistent with theorem 1. The simulator also verified that no cells were ever dropped for any GR flow.

The MAC protocol and scheduling algorithm applies to any arbitrary WMN topology with any number of BSs. The scheduling experiment has been repeated for WMNs with varying sizes and with varying topologies, with up to *1K* Base-Stations. The results are consistent with the theory established in [33,34,39] and with Fig. 4, ie the end-to-end delay and jitter are near-minimal and bounded. Additional results for other topologies are presented in [37,38].

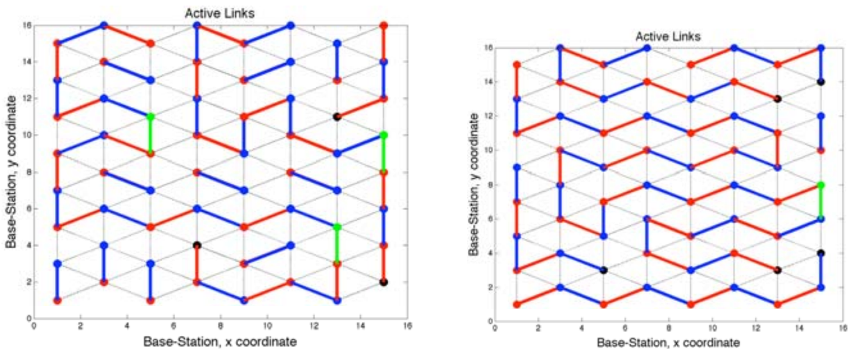

Fig. 5. Active links, 64 node WMN, 2 time-slots

Fig. 5 illustrates 2 typical transmission and reception sets, for a multi-channel hexagonal WMN with 64 nodes, arranged in an 8x8 mesh. Each BS has 2 transceivers and can simultaneously receive and transmit with 2 directed neighbors. A communication tree routed at BS(1) which provides each BS with a fixed rate of non-broadcast traffic was embedded into the WMN, as shown in Fig 1b. Additional point-to-point traffic between BSs was added, to saturate every wireless link. The resulting 64x64 traffic rate matrix was decomposed to yield a frame transmission schedule. A transmitting BS is denoted by a bold colored circle with a bold line to the receiver. In each time-slot, a greedy graph coloring algorithm is used to assign frequencies to the radio channels. Three frequencies are used in Fig. 5, while for these particular time-slots only 2 colors are necessary. In a general planar graph corresponding to hexagonal WMN, 3 colors are sufficient to color the graph.

4.3 Scalability

The proposed scheduling algorithm finds a solution for multihop WMNs which achieves 100% throughput and bounds the queue sizes per flow, end-to-end queuing delay and delay jitter to very small amounts, with computational time *O(NFlogNF)*, where *N* is the number of Base-Stations, and *F* is the number of time-slots per scheduling frame. The runtime complexity is the same as that of the well-known Fast-Fourier Transform (FFT) algorithm with *NF* points. The author estimates that the proposed algorithm runs in about $1/5^{th}$ the time of a comparable size FFT problem. A serial version of the proposed algorithm will solve the multi-hop WMN scheduling problem for reasonable *N* (8..128) with reasonable frame sizes *F* (*1K-4K* time-slots per frame) in milliseconds, faster than the WiMAX frame rate (typically 1-10 milliseconds). A parallel version of the proposed scheduler will solve larger scheduling problems much faster. Therefore, the algorithm is scalable to very large WMNs. The proposed scheduling algorithm was first proposed for use in packet-switched Internet routers in [34], which have considerably more demanding time constraints.

5 Conclusions

A collision-free low-jitter guaranteed-rate scheduling algorithm and MAC protocol for WMNs was presented. Within the WMN, fixed-size cells move between the BSs. IP packets are disassembled at the ingress BS, delivered through the WMN and reassembled once at the egress BS. It was shown that the scheduling problem in IQ switches can be transformed into a scheduling problem for WMNs. Therefore, the low-jitter GR scheduling algorithm for IQ switches presented in [33,34,35] can be used to find conflict-free low-jitter GR transmission sets in a WMN. Extensive simulations of hundreds of simultaneous competing GR traffic flows traversing many randomly generated WMNs were performed. In the single-channel WMN, admissible GR multimedia traffic can be provisioned and scheduled to achieve up to 67% of the system capacity. The remaining 33% of capacity can be used for Best-Effort traffic. In dual-channel WMNs with doubly stochastic traffic rate matrices, admissible GR multimedia traffic can be provisioned to achieve up to 100% of system capacity. For all provisioned GR flows, the number of cells per flow queued in each BS is near-minimal and bounded, and was experimentally observed to be between 1 and 2 cells per flow on average. The end-to-end delay for a flow along a path with H hops is also near-minimal and bounded, and was experimentally observed to be less than $2H \cdot IIDT$ time-slots. The end-to-end jitter for a flow along a path with H hops is also near-minimal and bounded, and was experimentally observed to be less than $O(K \cdot IIDT)$ time-slots. The cell loss rate due to primary scheduling conflicts is zero. These experimental results are consistent with the theoretical bounds reported in [34,39].

References

[1] Akyildiz, I.F., Wang, X.: A Survey on Wireless Mesh Networks. IEEE Radio Communications, S23–S30 (2005)

[2] Cao, M., Wang, X., Kim, S.K., Madihian, M.: Multi-Hop Wireless Backhaul Networks: A Cross-Layer Design Paradigm. IEEE JSAC 25(4), 738–748 (2007)

[3] Xergias, S., Passas, N., Salkintzis, A.K.: Centralized Resource Allocation for Multimedia Traffic in IEEE 802.16 Mesh Networks. Proc. IEEE 96(1), 54–63 (2008)

[4] Sharma, G., Mazumda, R.R., Shroff, N.B.: On the Complexity of Scheduling in Wireless Networks. In: IEEE Mobicom 2006 (2006)

[5] Tassiulas, L., Ephremides, A.: Stability Properties of Constrained Queueing Systems and Scheduling Policies for Maximum Throughput in Multihop Radio Networks. IEEE Trans. Automatic Control 37(12), 1936–1948 (1992)

[6] Tassiulas, L.: Linear Complexity algorithms for maximum throughput in radio networks and input queued switches. IEEE, Los Alamitos (1998)

[7] Tassiulas, L., Ephremides, A.: Joint optimal routing and scheduling in packet radio networks. IEEE Trans. Inform. Theory 38, 165–169 (1992)

[8] Jain, K., Padhye, J., Padmanabhan, V., Qiu, L.: Impact on Interference on multi-hop wireless networks performance. In: ACM Mobicom 2003 (2003)

[9] Jain, K., Padhye, J., Padmanabhan, V., Qiu, L.: Impact of Interference on Multi-Hop Wireless Network Performance. Wireless Networks 11, 471–487 (2005)

[10] Chou, C.T., Qadir, J., Lim, J.G.: Advances and Challenges with Data Broadcasting in Wireless Mesh Networks. IEEE Comm. Magazine, 78–122 (2007)
[11] Sharma, G., Shroff, N.B., Mazumdar, R.R.: Maximum Weighted Matching with Interference Constraints. In: IEEE Int. Conf. Pervasive Computing and Comm. Workshop (2006)
[12] Madan, R., Cui, S., Lall, S., Goldsmith, A.J.: Cross-layer design for lifetime maximization in interference-limited wireless sensor networks. In: IEEE Infocom (2005)
[13] Kodlialam, M., Nandagopal, T.: Characterizing Achievable rates in multi-hop wireless mesh networks: the joint routing and scheduling problem. In: ACM Mobicom 2003, San Diego, California (2003)
[14] Cruz, R.L., Santhaman, A.V.: Optimal routing, link scheduling and power control in multi-hop wireless networks. In: Proc. IEEE Infocom, vol. 1, pp. 702–711 (2005)
[15] Xiao, L., Johansoon, M., Boyd, S.P.: Simultaneous routing and resource allocation via dual decomposition. IEEE Trans. Comm. 52(7), 1136–1144 (2004)
[16] Elliot, T., Ephremides, A.: Joint scheduling and power control for wireless ad hoc networks. IEEE Trans. Wireless Comm. 1, 74–85 (2004)
[17] Sharma, G., Shroff, N.B., Mazumdar, R.R.: Joint Congestion Control and Distributed Scheduling for Throughput Guarantees in Wireless Networks. In: IEEE Infocom, pp. 2072–2080 (2007)
[18] Parekh, A.K., Gallager, R.G.: A Generalized Processor Sharing Approach to Flow Control in Integrated Service Networks: the Single Node Case. IEEE/ACM Trans. Networking 1, 344–357 (1993)
[19] Parekh, A.K., Gallager, R.G.: A Generalized Processor Sharing Approach to Flow Control in Integrated Service Networks: the Multiple Node Case. IEEE/ACM Trans. Networking 2(2), 137–150 (1994)
[20] Jajszczyk, A.: Nonblocking, Repackable and Rearrangeable Clos Networks: Fifty Years of the Theory Evolution. IEEE Comm. Magazine, 28–33 (2003)
[21] McKeown, N., Mekkittikul, A., Anantharam, V., Walrand, J.: Achieving 100% Throughput in an Input Queued Switch. Trans. Comm. 47(8), 1260–1267 (1999)
[22] Lotfinezhad, M., Liang, B., Sousa, E.S.: Dynamic Control of Tunable Sub-optimal Algorithms for Scheduling of Time-varying Wireless Networks. In: IEEE iWQoS Conf., Enschede, Netherlands, pp. 153–163 (2008)
[23] Gourgy, A., Szymanski, T.H., Down, D.: On Tracking the Behaviour of an Output Queued Switch using an Input Queued Switch. IEEE Trans. Networking (submitted)
[24] Koksal, C.E., Gallager, R.G., Rohrs, C.E.: Rate Quantization and Service Quality over Single Crossbar Switches. In: IEEE Infocom (2004)
[25] Gopya, P., Vin, H.M.: Generalized Guaranteed Rate Scheduling Algorithms: A Framework. IEE/ACM Trans. Networking 5(4), 561–571 (1997)
[26] Keslassy, I., Kodialam, M., Lakshamn, T.V., Stiliadis, D.: On Guaranteed Smooth Scheduling for Input-Queued Switches. IEEE/ACM Trans. Networking 13(6) (2005)
[27] Kodialam, M.S., Lakshman, T.V., Stilladis, D.: Scheduling of Guaranteed-bandwidth low-jitter traffic in input-buffered switches. US Patent Application #20030227901
[28] Chen, W.J., Chang, C.-S., Huang, H.-Y.: Birkhoff-von Neumann Input Buffered Crossbar Switches. IEEE Trans. Comm. 49(7), 1145–1147 (2001)
[29] Chang, C.-S., Chen, W.J., Huang, H.-Y.: On Service Guarantees for Input Buffered Crossbar Switches: A Capacity Decomposition Approach by Birkhoff and von Neuman. In: IEEE iWQoS 1999, pp. 79–86 (1999)
[30] Mohanty, S.R., Bhuyan, L.N.: Guaranteed Smooth Switch Scheduling with Low Complexity. In: IEEE Globecom, pp. 626–630 (2005)

[31] Sun, W., Shin, K.G.: End-to-End Delay Bounds for Traffic Aggregates Under Guaranteed-Rate Scheduling Algorithms. IEEE/ACM Trans. Networking 11(3), 1188–1201 (2005)
[32] Giaccone, P., Leonardi, E., Shat, D.: Throughput Region of Finite-Buffered Networks. IEEE Trans. PDS 11(12), 251–263 (2007)
[33] Szymanski, T.H.: QoS Switch Scheduling using Recursive Fair Stochastic Matrix Decomposition. In: IEEE Int. Conf. HPSR, pp. 417–424 (2006)
[34] Szymanski, T.H.: A Low-Jitter Guaranteed-Rate Scheduling Algorithm for Packet-Switched IP Routers, accepted (with revision). IEEE Trans. on Comm. (2008)
[35] Szymanski, T.H.: Method and Apparatus to Schedule Packets through a Crossbar Switch with Delay Guarantees, US Patent Application (2006)
[36] Szymanski, T.H.: Method and Apparatus to Schedule Packets through a Wireless Mesh Network with Near Minimal Delay and Jitter, US Provisional Patent App. (2008)
[37] Szymanski, T.H., Gilbert, D.: Delivery of Guaranteed Rate Internet Traffic with Very Low Delay Jitter. In: IEEE Pacific Rim Conf. on Comm. Comp. and Signal Processing, Canada, pp. 450–455 (2007).
[38] Szymanski, T.H., Gilbert, D.: Low-Jitter Guaranteed-Rate Communications for Cluster Computing Systems. Pacific Rim Special Issue. Int. Journal of Computer Networks and Distributed Systems (to appear) (2008)
[39] Szymanski, T.H.: Bounds on End-to-End Delay and Jitter in Input-Buffered and Internally Buffered IP Networks (submitted) (2008)