Throughput-Optimal Broadcast in Wireless Networks with Point-to-Multipoint Transmissions

Abhishek Sinha and Eytan Modiano, *Fellow, IEEE*

Abstract—We consider the problem of efficient packet dissemination in wireless networks with point-to-multipoint wireless broadcast channels. We propose a dynamic policy, which achieves the broadcast capacity of the network. This policy is obtained by first transforming the original multi-hop network into a precedence-relaxed virtual single-hop network and then finding an optimal broadcasting policy for the relaxed network. The resulting policy is shown to be throughput-optimal for the original wireless network using a sample-path argument. We also prove the **NP**-completeness of the finite-horizon broadcasting problem, which is in contrast with the polynomial-time solvability of the problem with point-to-point channels. Illustrative simulation results demonstrate the efficacy of the proposed broadcast policy in achieving the full broadcast capacity with low delay.

Index Terms—Wireless broadcasting, scheduling, queueing theory, throughput optimality

1 Introduction and Related Work

The problem of disseminating packets efficiently from a set of source nodes to all nodes in a network is known as the *Broadcasting Problem*. Broadcasting is a fundamental network functionality, which is used frequently in numerous practical applications, including military communication [1], information dissemination in disaster management [2], in-network function computation [3], and efficient dissemination of control information in vehicular networks [4].

Due to its fundamental nature, the Broadcasting problem in wireless networks has been studied extensively in the literature. As a result, a number of different algorithms have been proposed for optimizing different efficiency metrics. Examples include minimum energy broadcasting [5], minimum latency broadcasting [6], broadcasting with minimum number of retransmissions [7], and throughput-optimal broadcasting [8]. A comprehensive study of different broadcasting algorithms proposed for Mobile Adhoc networks is presented in [9].

A fundamental feature of the wireless medium is the inherent point-to-multipoint nature of wireless links, where a packet transmitted by a node can be heard by all its neighbors. This feature, also known as the wireless broadcast advantage, is especially useful in network-wide broadcast applications, where the objective is to disseminate the packets among all nodes in the network efficiently. Additionally, because of inter-node interference, the set of simultaneous transmissions in a wireless network is restricted to the set of non-interfering feasible schedules. Designing a broadcast algorithm which efficiently utilizes the broadcast advantage, while respecting the interference constraints is a challenging problem.

The problem of throughput optimal multicasting in wired networks has been considered in [10]. In our recent works [11], [12], [13], we studied the problem of throughput optimal broadcasting in wireless networks with directed *point-to-point-links* and designed several efficient broadcasting algorithms. The problem of designing throughput optimal broadcasting policy in wireless networks with *point-to-multipoint links* was considered in [14], where the authors studied a highly restrictive "scheduling-free" model, where it is assumed that scheduling decisions are made by a central controller, acting independently of their algorithm. With this assumption, they obtained a randomized packet forwarding scheme, which requires a continuous exchange of control information among the neighboring nodes. This algorithm was proved to be throughput optimal *with respect to the given schedules*, using fluid limit techniques. There is also a large body of literature dedicated to the broadcasting/multicasting problem using network coding [15], [16], [17], [18]. However, in this work, we do not assume any coding capability of the nodes and stick to the simple copy-and-forward-based policies to (1) minimize the packet decoding delay, (2) eliminate the computational burden of coding on power-constrained wireless nodes, and (3) reduce the memory requirement for maintaining the system-state.

This paper considers the joint problem of throughput optimal scheduling *and* packet dissemination in wireless networks with point-to-multipoint links. Our approach uses the concept of *virtual network*, that we recently introduced in [19] for solving the generalized network flow problem with point-to-point links. To the best of our knowledge, this is the first known throughput optimal broadcasting algorithm in wireless networks with broadcast advantage.

- *A. Sinha is with the Department of Electrical Engineering, IIT Madras, Chennai, Tamil Nadu 600036, India. E-mail: abhishek.sinha@ee.iitm.ac.in.*
- *E. Modiano is with the Laboratory for Information and Decision Systems, MIT, Cambridge, MA 02139 USA. E-mail: modiano@mit.edu.*

Manuscript received 8 Jan. 2019; revised 22 Aug. 2019; accepted 2 Sept. 2019. Date of publication 10 Sept. 2019; date of current version 3 Dec. 2020. (Corresponding author: Abhishek Sinha.)
Digital Object Identifier no. 10.1109/TMC.2019.2940025

The main contributions of this paper are as follows:

- We propose a dynamic online policy for throughput-optimal broadcasting in wireless networks with point-to-multipoint links.
- We prove the *NP*-completeness of the corresponding finite horizon wireless broadcasting problem.
- We introduce a new control policy and proof technique by combining the stochastic Lyapunov drift theory with the deterministic adversarial queueing theory. This methodology enables us to derive a stabilizing control policy for a multi-hop network by solving the problem on a simpler *precedence-relaxed* virtual single-hop network.

The rest of the paper is organized as follows. In Section 2 we describe the system model and formulate the problem. In Section 3 we prove the hardness of the finite-horizon version of the problem. Next, in Section 4 we derive an optimal control policy for a related relaxed version of the wireless network. This control policy is then applied to the original unrelaxed network in Section 5, where we show that the resulting policy is throughput-optimal when used in conjunction with a priority-based packet scheduling policy. In Section 6, we demonstrate the efficacy of the proposed policy via numerical simulations. Finally, we conclude the paper in Section 8. A preliminary version of this paper appeared earlier in [20].

2 System Model and Problem Formulation

We consider the problem of efficiently disseminating packets, arriving randomly at source nodes, to all nodes in a wireless network. To clarify the terminology, in this paper, the term "broadcasting" refers to the act of routing packets to all nodes in a network in a *multi-hop* fashion. On the other hand, the term "point-to-multipoint transmission" refers to the multicasting nature of single-hop wireless links, where a transmitted packet is heard by all neighboring nodes of a transmitter due to the wireless broadcast advantage. The system model and the precise problem statement are described below.

2.1 Network Model

Consider a wireless network with its topology given by the directed graph $\mathcal{G}(V, E)$. The set V denotes the set of all nodes, with $|V| = n$. If the node j is within the transmission range of node i, there is a directed edge $(i, j) \in E$ connecting them. Due to the inherent point-to-multipoint broadcast nature of the radio channel, a transmitted packet can be heard by all out-neighbors of the transmitting node. In other words, the packets are transmitted over the *hyperedges*, where a hyperedge is defined to be the union of all outgoing edges from a node. The system evolves in a slotted time structure. External packets, which are to be broadcasted throughout the network, arrive at designated source nodes. For simplicity of exposition, we consider only static networks with a single source node `r`. However, the algorithm and its analysis presented in this paper extend to time-varying dynamic networks with multiple source nodes in a straightforward manner. We will consider time-varying networks in our numerical simulations.

2.2 Wireless Transmission Model

When a node $i \in V$ is scheduled for transmission, it can transmit any of its received packets at the rate of c_i packets per slot to *all* of its out-neighbors over its outgoing hyperedge. See Fig. 1. Due to the wireless interference constraint, only a selected subset of nodes can feasibly transmit over the hyperedges simultaneously without causing a collision. The wireless channel is assumed to be error-free otherwise. The set of all feasible transmission schedules may be concisely described using the notion of a *Conflict Graph* $\mathcal{C}(\mathcal{G})$. The set of vertices in the conflict graph is the same as the set of nodes in the network V. There is an edge between two nodes in the conflict graph if and only if these two nodes *cannot* transmit simultaneously without causing collision. Note that our *node-centric* definition of conflict graphs is a little different from the traditional *edge-centric* definition of conflict graph, which concerns point-to-point transmissions [21].

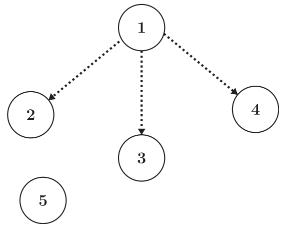


Fig. 1. An example of packet transmission over hyperedges—when the node 1 transmits a packet, assuming no interference, it is received simultaneously by the neighboring nodes 2, 3, and 4.

As the simplest example of the interference model, consider a wireless network where each node transmits on a separate channel, causing no inter-node interference. Hence, any subset of nodes can transmit at the same slot, and the conflict graph does not contain any edges. For another example, consider a wireless network subject to primary interference constraints. In this case, the edge (i, j) is *absent* in the conflict graph $\mathcal{C}(\mathcal{G})$ if and only if nodes i and j are *not* in the transmission range of each other and their out-neighbor-sets are disjoint. The set of all feasible transmission schedules $\mathcal{M}$ consists of the set of all *Independent Sets* in the conflict graph.

Note that the above definition of feasible schedules and conflict graph does not allow *any* collision in the network. The same assumption was also used in [14], where such schedules were called "interference-free". However, due to the point-to-multipoint nature of the wireless medium, it is possible (and sometimes beneficial) to consider schedules that allow some collisions, so that a transmitted packet may be correctly received only by a strict subset of neighbors. As it will be clear in what follows, it is straightforward to extend our algorithm to allow such general schedules, albeit at the expense of additional computational complexity. In order to present the main ideas in a simplified setting, in the following, we stick to the "interference-free" schedules, as defined above.

2.3 The Broadcast Policy-Space Π

We first recall the definition of a connected dominating set of a graph $\mathcal{G}$ [22].

Definition 1 (Connected Dominating Set). *A connected dominating set D of a graph $\mathcal{G}(V, E)$ is a subset of vertices with the following properties:*

- *The source node `r` is in D.*
- *The induced subgraph $\mathcal{G}(D)$ is connected.*
- *Every vertex in the graph either belongs to the set D or is adjacent to a vertex in the set D.*

A connected dominating set D is called *minimal* if $D \setminus \{v\}$ is not a connected dominating set for any $v \in D$. The set of all minimal connected dominating set is denoted by $\mathcal{D}$.

A packet p is said to have been *broadcasted* by time t if the packet p is present at every node in the network by time t.

It is evident that a packet p is broadcasted if it has been transmitted sequentially by every node in a connected dominating set D. An admissible broadcast policy π is a sequence of actions $\{\pi_t\}_{t\geq 0}$ executed at every slot t. The action at time slot t consists of the following three operations:

1) *Route Selection:* Assign a connected dominating set $D \in \mathcal{D}$ to every incoming packet at the source $\mathtt{r}$ for routing.
2) *Node Activation:* Activate a subset of nodes from the set of all feasible activations $\mathcal{M}$.
3) *Packet Scheduling:* Transmit packets from the activated nodes according to some scheduling policy.

The set of all admissible broadcast policies is denoted by Π. The actions executed at every slot may depend on any past or future packet arrival and control actions.

Assume that under the action of the broadcast-policy π, the set of packets received by node i at the end of slot T is $N_i^\pi(T)$. Then the set of packets $B(T)$ received by all nodes, at the end of time T is given by

$$B^\pi(T) = \bigcap_{i\in V} N_i^\pi(T). \tag{1}$$

2.4 Broadcast Capacity λ^*

Let $R^\pi(T) = |B^\pi(T)|$ denote the number of packets delivered to all nodes in the network up to time T, under the action of an admissible policy π. Also assume that the external packets arrive at the source node with expected rate of λ packets per slot. The policy π is called a *broadcast policy of rate* λ if

$$\lim_{T\to\infty} \frac{R^\pi(T)}{T} = \lambda, \quad \text{w.p.1.} \tag{2}$$

The broadcast capacity λ^* of the network is defined as

$$\lambda^* = \sup_{\pi\in\Pi}\{\lambda : \pi \text{ is a broadcast policy of rate } \lambda\}. \tag{3}$$

The Wireless Broadcasting problem is defined as finding an admissible policy π that achieves the Broadcast rate λ^*.

3 Hardness Results

Since a broadcast policy, as defined above, continues to be executed forever (compared to the finite termination property of standard algorithms), the usual notions of computational complexity theory do not directly apply in characterizing the complexity of these policies. Nevertheless, we show that the closely related problem of finite horizon broadcasting is NP-hard. Remarkably, this problem remains hard even if the node activation constraints are eliminated (i.e., all nodes can transmit packets at the same slot, which occurs e.g., when each node transmits over a distinct channel). Thus, the hardness of the problem arises from the combinatorial nature of distributing the packets among the nodes. This result is in sharp contrast with the polynomially solvable Wired Broadcast problem where the broadcast nature of the wireless medium is absent and different outgoing edges from a node can transmit different packets at the same slot [11], [13], [23].

Consider the following finite horizon problem called Wireless Broadcasting, with the input parameters $\mathcal{G}, M, T$.

- *INSTANCE:* A Graph $\mathcal{G}(V, E)$ with capacities C on the nodes. A set of $\mathcal{M}$ of packets with $|\mathcal{M}| = M$ at the source and a time horizon of T slots.
- *QUESTION:* Is there a scheduling algorithm π which routes all of these M packets to all nodes in the network by time $T \geq 2$, i.e., $B^\pi(T) = \mathcal{M}$?

We prove the following hardness result:

Theorem 3.1. Wireless Broadcast *is* NP*-complete.*

Proof of Theorem 3.1 is based on reduction from the the *NP*-complete problem Monotone Not All Equal 3 – SAT [24] to the Wireless Broadcasting problem. The complete proof of the Theorem is provided in Section 7.1.

Note that the problem for $T = 1$ is trivial as only the out-neighbors of the source receive $\min(C, M)$ packets at the end of the first slot. The problem becomes non-trivial for any $T \geq 2$. In our reduction, we show that the problem is hard even for $T = 2$. This reduction technique may be extended in a straightforward fashion to show that the problem remains *NP*-complete for any fixed $T \geq 2$.

The above hardness result is in sharp contrast with the efficient solvability of the broadcasting problem in the setting of point-to-point channels. In wired networks, the broadcast capacity can be achieved by routing packets using maximal edge-disjoint spanning trees, which can be efficiently computed using Edmonds' algorithm [23]. In a recent series of papers [11], [12], we proposed efficient throughput-optimal algorithms for wireless Directed Acyclic Graphs (DAG) in the static and time-varying settings. In a follow-up paper [13], the above line of work was extended to networks with arbitrary topology. In contrast, Theorem 3.1 and its corollary establishes that achieving the broadcast capacity in a wireless network with the broadcast channel is intractable even for simple network topology, such as a DAG. Also notice that this hardness result is inherently different from the hardness result of [25], where the difficulty stems from the hardness of max-weight node activations, which is an Independent Set problem. The above result should also be contrasted with the hardness of the minimum energy broadcasting problem [26].

4 Throughput-Optimal Broadcasting Policy for a Relaxed Network

In this section, we give a brief outline of the design of the proposed broadcast policy, which will be described in detail in the subsequent sections. At a high level, the proposed policy consists of two interdependent modules—a control policy for a *precedence-relaxed* virtual network described below, and a control policy for the actual physical network, described in Section 5. Although from a practical point of view, we are ultimately interested in the optimal control policy for the physical network, as we will soon see, this

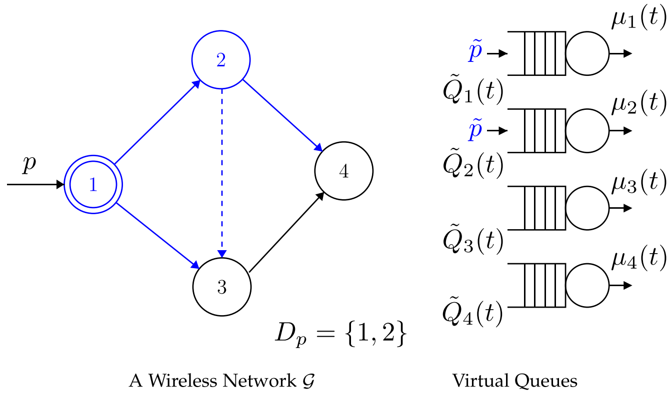


Fig. 2. Illustration of the virtual queue system for the four-node wireless network $\mathcal{G}$. Upon arrival, the incoming packet p is prescribed a connected dominating set $D_p = \{1,2\}$, which is used for its broadcasting. Relaxing the precedence constraint, packet p is counted as an arrival to the virtual queues $\tilde{Q}_1$ and $\tilde{Q}_2$ at the *same slot*. In the physical system, the packet p may take a while before reaching node 2, depending on the control policy.

control policy is intimately related to, and derived from the dynamics of the relaxed virtual network. The concept of precedence relaxed virtual network was first introduced in our recent paper [19].

4.1 Virtual Network and Virtual Queues

In this section we define and analyze the dynamics of an auxiliary virtual queueing process $\{\tilde{\boldsymbol{Q}}(t)\}_{t\geq 0}$. Our throughput-optimal broadcasting policy π^* will be described in terms of the virtual queues. We emphasize that virtual queues are not physical entities and they do not contain any physical packet. They are constructed solely for the purpose of designing a throughput-optimal policy for the physical network, which depends only on the value of the virtual queue lengths. More interestingly, the designed virtual queues correspond to a fairly natural *single-hop* relaxation of the *multi-hop* physical network, as detailed below.

4.1.1 A Precedence-Relaxed System

Consider an incoming packet p arriving at the source, which is to be broadcasted through a sequence of transmissions by nodes in a connected dominating set $D_p \in \mathcal{D}$. Appropriate choice of the set D_p is a part of our policy and will be discussed shortly. In reality, the packet p cannot be transmitted by a non-source node $v \in D_p$ at time t if it has not already reached the node v by the time t. This causality constraint is known as the *precedence constraint* in the literature [27]. We obtain the virtual queue process $\tilde{\boldsymbol{Q}}(t)$ by relaxing the precedence constraint, i.e., in the virtual queuing system, the packet p is made available for transmission by all nodes in the set D_p when the packet first arrives at the source. See Fig. 2 for an illustration.

4.1.2 Dynamics of the Virtual Queues

Formally, for each node $i \in V$, we define a virtual queue variable $\tilde{Q}_i(t)$. As described above, on the arrival of an external packet p at the source $\mathtt{r}$, the packet is replicated to a set of virtual queues $\{\tilde{Q}_i(t), i \in D_p\}$, where $D_p \in \mathcal{D}$ is a connected dominating set of the graph. Mathematically, this operation means that all virtual queue-counters in the set D_p are incremented by the number of external arrivals at the slot t. We will use the control variable $A_i(t)$ to denote the number of packets that were routed to the virtual queue $\tilde{Q}_i$ at time t. The service rate $\boldsymbol{\mu}(t)$ allocated to the virtual queues is required to satisfy the same interference constraint as the physical network, i.e., $\boldsymbol{\mu}(t) \in \mathcal{M}, \forall t$. Hence, we can write the one step dynamics of the virtual queues as follows:

$$\tilde{Q}_i(t+1) = (\tilde{Q}_i(t) + A_i(t) - \mu_i(t))^+, \qquad \forall i \in V. \tag{4}$$

4.2 Dynamic Control of the Virtual Queues

In this section, we design a dynamic control policy to stabilize the virtual queues for all arrival rates $\lambda < \lambda^*$. This policy takes action (choosing the routes of the incoming packets and selecting a feasible transmission schedule) by observing the virtual queue-lengths only and, unlike popular unicast policies such as *Backpressure*, does not require physical queue information. This control policy is obtained by minimizing one-step expected drift of an appropriately chosen Lyapunov function as described below. In the next section we will show how to combine this control policy for the virtual queues with an appropriate packet scheduling policy for the physical networks, so that the overall policy is throughput-optimal.

Consider the Lyapunov function $L(\cdot)$ defined as the Eucledian 2-norm of the virtual queue lengths, i.e.,

$$L(\tilde{\boldsymbol{Q}}(t)) = ||\tilde{\boldsymbol{Q}}(t)||_2 = \sqrt{\sum_i \tilde{Q}_i^2(t)}. \tag{5}$$

The one step drift $\Delta(t)$ of the Lyapunov function may be bounded as follows:

$$\begin{aligned}\Delta(t) &\equiv L(\tilde{\boldsymbol{Q}}(\boldsymbol{t+1})) - L(\tilde{\boldsymbol{Q}}(\boldsymbol{t}))\\ &= \sqrt{\sum_i \tilde{Q}_i^2(t+1)} - \sqrt{\sum_i \tilde{Q}_i^2(t)}.\end{aligned} \tag{6}$$

To bound this quantity, notice that for any $x \geq 0$ and $y > 0$, we have

$$\sqrt{x} - \sqrt{y} \leq \frac{x-y}{2\sqrt{y}}. \tag{7}$$

The inequality above follows by noting that RHS minus LHS is non-negative. Substituting $x = ||\tilde{\boldsymbol{Q}}(t+1)||^2$ and $y = ||\tilde{\boldsymbol{Q}}(t)||^2$ in the inequality (7), we have the following bound on the one-step drift (6) for any $||\tilde{\boldsymbol{Q}}(t)|| > 0$

$$\Delta(t) \leq \frac{1}{2||\tilde{\boldsymbol{Q}}(t)||}\left(\sum_i \left(\tilde{Q}_i^2(t+1) - \tilde{Q}_i^2(t)\right)\right). \tag{8}$$

From the virtual queue dynamics (4), we have

$$\begin{aligned}\tilde{Q}_i(t+1)^2 &\leq (\tilde{Q}_i(t) - \mu_i(t) + A_i(t))^2\\ &= \tilde{Q}_i^2(t) + A_i^2(t) + \mu_i^2(t) + 2\tilde{Q}_i(t)A_i(t)\\ &\quad - 2\tilde{Q}_i(t)\mu_i(t) - 2\mu_i(t)A_i(t).\end{aligned}$$

Since $\mu_i(t) \geq 0$ and $A_i(t) \geq 0$, we have

$$\begin{aligned}\tilde{Q}_i^2(t+1) - \tilde{Q}_i^2(t) &\leq A_i^2(t) + \mu_i^2(t)\\ &\quad + 2\tilde{Q}_i(t)A_i(t) - 2\tilde{Q}_i(t)\mu_i(t).\end{aligned} \tag{9}$$

Hence, combining Eqns. (8) and (9), the one-step Lyapunov drift, conditional on the current virtual queue-length $\tilde{\boldsymbol{Q}}(t)$,

TABLE 1
Comparison of the Broadcasting Policy Proposed in this Paper with that of [19]

The broadcasting policy proposed in [19]	The broadcasting policy proposed in this paper
1. This paper considers the generalized flow problem, including the broadcasting problem with point-to-point links.	1. This paper considers the broadcasting problem with point-to-multipoint wireless links.
2. The broadcast-route for a packet corresponds to a Spanning Tree.	2. The broadcast-route for a packet corresponds to a Connected Dominating Set.
3. The routing part of the broadcasting problem is in **P**.	3. The routing part of the broadcasting problem is **NP-Complete**.
4. In the proposed policy, the virtual queues are associated with links.	4. In the proposed policy, the virtual queues are associated with nodes.

under the action of an admissible policy π is upper-bounded as

$$\begin{aligned}
&\mathbb{E}(\Delta^\pi(t)|\tilde{\boldsymbol{Q}}(t)=\tilde{\boldsymbol{Q}})\\
&\stackrel{(\text{def})}{=}\mathbb{E}\big(L(\tilde{\boldsymbol{Q}}(t+1))-L(\tilde{\boldsymbol{Q}}(t))|\tilde{\boldsymbol{Q}}(t)=\tilde{\boldsymbol{Q}}\big)\\
&\leq\frac{1}{2||\tilde{\boldsymbol{Q}}||}\Bigg(B+2\underbrace{\sum_i\tilde{Q}_i(t)\mathbb{E}\big(A_i^\pi(t)|\tilde{\boldsymbol{Q}}(t)=\tilde{\boldsymbol{Q}}\big)}_{(a)}\\
&\quad-2\underbrace{\sum_i\tilde{Q}_i(t)\mathbb{E}\big(\mu_i^\pi(t)|\tilde{\boldsymbol{Q}}(t)=\tilde{\boldsymbol{Q}}\big)}_{(b)}\Bigg),
\end{aligned} \tag{10}$$

where the constant $B=\sum_i(\mathbb{E}A_i^2(t)+\mathbb{E}\mu_i^2(t))\leq n(\mathbb{E}A^2+c_{\max}^2)$. By minimizing the upper-bound on drift from Eqn. (10), and exploiting the separable nature of the objective, we obtain the following control policy for the virtual queues:

***U**niversal **M**ax **W**eight (*UMW*) Policy for the Virtual Queues*

1. *Route Selection*. We minimize the term (a) in the above with respect to all feasible controls to obtain the following routing policy: Route the incoming packet at time t along the minimum-weight connected dominating set (MCDS) $D^{\text{UMW}}(t)$, where the nodes are weighted by the virtual queue-lengths $\tilde{\boldsymbol{Q}}(t)$, i.e.,

$$D^{\text{UMW}}(t)=\arg\min_{D\in\mathcal{D}}\sum_{i\in V}\tilde{Q}_i(t)1(i\in\mathcal{D}). \tag{11}$$

2. *Node Activations*. We maximize the term (b) in the above with respect to all feasible controls to obtain the following node scheduling policy: At time t activate a feasible schedule $\boldsymbol{\mu}^{\text{UMW}}(t)$ having the maximum weight, where the nodes are weighted by the virtual queue-lengths $\tilde{\boldsymbol{Q}}(t)$, i.e.,

$$M^{\text{UMW}}(t)=\arg\max_{M\in\mathcal{M}}\sum_{i\in V}\tilde{Q}_i(t)c_i1(i\in M). \tag{12}$$

In connection with the virtual queue systems $\tilde{\boldsymbol{Q}}(t)$, we establish the following theorem which will be essential in the proof of the throughput-optimality of the overall algorithm involving physical queues.

Theorem 4.1. *For any arrival rate $\lambda<\lambda^*$ the virtual queue process $\{\boldsymbol{Q}(t)\}_{t\geq 0}$ is positive recurrent under the action of the* UMW *policy and*

$$\max_i\tilde{Q}_i(t)=\mathcal{O}(\log t)^1,\qquad \text{w.p. } 1.$$

The proof of Theorem 4.1 involves construction of an efficient randomized policy and using it with a sharper form of the Foster-Lyapunov theorem by Hajek [28]. This leads to the desired sample path result. The proof is provided in Section 7.2.[1]

Discussion of the Result: Even though the virual queue process is positive recurrent under the action of the UMW policy, it is not true that they are uniformly bounded almost surely. Theorem 4.1 states that, instead, the virtual queue lengths increase at most logarithmically with time almost surely. Theorem 4.1 also strengthens the result of Theorem 2.8 of [29], where an almost sure $o(t)$ bound was established for the queue lengths.[2]

In the rest of the paper, we will primarily focus on the typical sample paths $\mathcal{E}$ of the virtual queue process satisfying the above almost sure bound. Formally, we define the set $\mathcal{E}$ to be

$$\max_i\tilde{Q}_i(\omega,t)=\mathcal{O}(\log(t)),\qquad\forall\omega\in\mathcal{E}, \tag{13}$$

where $\mathbb{P}(\mathcal{E})=1$ from Theorem 4.1.

Given the apparent similarity between this algorithm and the algorithm presented in our earlier work [19], it is worth noting their difference as highlighted in Table 1.

4.3 Bound on the Virtual Queue Size

Recall that the random variable $A_i(t)$ denotes the total number of packets injected to the virtual queue $\tilde{Q}_i$ at time t. Similarly, the random variable $\mu_i(t)$ denotes the service rate from the virtual queue $\tilde{Q}_i$ at time t. Hence, the total number of packets that have been injected into any virtual queue $\tilde{Q}_i$ within the time interval $[t_1,t_2)$, $t_1\leq t_2$ is given by

$$A_i(t_1,t_2)=\sum_{\tau=t_1}^{t_2-1}A_i(\tau). \tag{14}$$

Similarly, the total amount of service offered to the virtual queue $\tilde{Q}_i$ within the time interval $[t_1,t_2)$ is given by

$$S_i(t_1,t_2)=\sum_{\tau=t_1}^{t_2-1}\mu_i(\tau). \tag{15}$$

Using the well-known Skorokhod representation theorem [30] of the Queueing recursion (4), we have[3]

$$\tilde{Q}_i(t)=\sup_{1\leq\tau\leq t}\big(A_i(\tau,t)-S_i(\tau,t)\big)^+. \tag{16}$$

1. Recall that, $f(t)=\mathcal{O}(g(t))$ if there exists a positive constant c and a finite time t_0 such that $f(t)\leq cg(t),\forall t\geq t_0$.

2. We say $f(t)=o(g(t))$ if $\lim_{t\to\infty}\frac{f(t)}{g(t)}=0$.

3. Note that, for simplicity of notation and without any loss of generality, we have assumed the system to be empty at time $t=0$.

Since the virtual queues $\tilde{Q}$ are controlled by the UMW policy, combining Eqn. (13) with (16), we have for all typical sample paths $\omega \in \mathcal{E}$

$$A_i(\omega;\tau,t) \leq S_i(\omega;\tau,t) + F(\omega,t), \qquad \forall \tau \leq t, i \in V, \tag{17}$$

where $F(\omega,t) = \mathcal{O}(\log t)$. In other words, Equn. (17) states that under the UMW policy, for any packet arrival rate $\lambda < \lambda^*$, the total number of packets that are routed to any virtual queue $\tilde{Q}_i$ may exceed the total amount of service offered to that queue in any time interval $[\tau, t)$ by at most an additive term of $\mathcal{O}(\log t)$ almost surely. In the following section, we will show that this arrival condition enables us to design a throughput-optimal broadcasting policy.

5 Control of the Physical Network

With the help of the one-hop virtual queue structure designed in the previous section, we now focus our attention on designing a throughput-optimal control policy for the multi-hop physical network. Recall from Section 2 that a broadcast policy for the physical network is specified by the following three components: (1) Route Selection, (2) Node Activation, and (3) Packet Scheduling. In our proposed broadcast policy, components (1) and (2) for the physical network are identical to the corresponding components in the virtual network. In other words, an incoming packet p at time t is prescribed a route (i.e., a connected dominating set) given by Eqn. (11) and the set of nodes given by Eqn. (12) are scheduled for transmission in that slot. Note that, both these decisions are based on the instantaneous virtual queue lengths $\tilde{\boldsymbol{Q}}(t)$. In particular, it is possible that a particular node, with positive virtual queue length, is scheduled for transmission in a slot, even though it does not have any packets to transmit in its physical queue. The surprising fact that will follow from Theorem 5.2 is that this kind of wasted transmissions are rare and *they do not affect throughput*.

Packet Scheduling. There are many possibilities for the component (3), i.e., Packet scheduling in the physical network. Recall that, the packet scheduling component selects packet(s) to be transmitted (subject to the node capacity constraint) when multiple packets contend for transmission by an active node and plays a role in determining the physical queuing process. In this paper, we consider a priority based scheduler which gives priority to the packet which has been transmitted by the nodes *the least number of times*. We call this scheduling policy **L**east **T**ransmitted **F**irst or LTF. The LTF policy is inspired from the *Nearest To Origin* policy of Gamarnik [31], where it was shown to stabilize the queues for the unicast problem in wired networks in a deterministic adversarial setting. In spite of the high-level similarities, however, we emphasize that these two policies are different, as the LTF policy works in the broadcast setting with point-to-multipoint transmissions and involves packet duplications.

Definition 2 (The policyLTF). *If multiple packets are available for transmission by an active node at the same time slot t, the* LTF *scheduling policy gives priority to a packet which has been transmitted the smallest number of times among all other contending packets.*

See Fig. 3 for an illustration of the LTF policy.

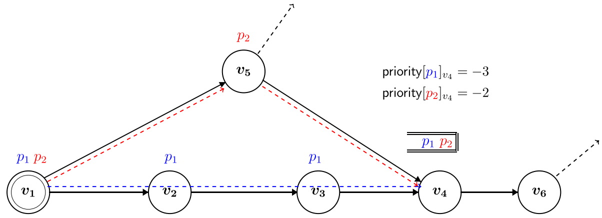


Fig. 3. A schematic diagram depicting the scheduling policy LTF in action. The packet p_1's broadcast route consists of the nodes $\{v_1, v_2, v_3, v_4, \ldots\}$ and the packet p_2's broadcast route consists of the nodes $\{v_1, v_5, v_4, \ldots\}$ as shown in the figure. At node v_4, according to the LTF policy, the packet p_2 has higher priority than the packet p_1 for transmission.

5.1 Stability of the Physical Queues

Let us denote the length of the physical queue at node i at time t by $Q_i(t)$. Note that the number of packets which arrive at the source in the time interval $[\tau, t)$ and whose prescribed route contains the node i, is equal to the corresponding arrival in the virtual network $A_i(\tau, t)$, given by Eqn. (14). Similarly, total service offered by the physical node i in the time interval $(\tau, t]$ is given by $S_i(\tau, t)$, defined in Eqn. (15). Thus, the bound in Eqn. (17) may be interpreted in terms of the packets arriving to the physical network. This leads to the following theorem:

Theorem 5.1. *Under the action of the* UMW *policy with* LTF *packet scheduling, we have for any arrival rate $\lambda < \lambda^*$*

$$\sum_{i \in V} Q_i(t) = \mathcal{O}(\log t), \qquad \text{w.p. } 1.$$

This implies that

$$\lim_{t \to \infty} \frac{\sum_{i \in V} Q_i(t)}{t} = 0, \qquad \text{w.p. } 1,$$

i.e., the physical queues are rate-stable.

Theorem 5.1 is established by combining the key sample path property of arrivals and departures from Eqn. (17), with an adversarial queueing theoretic argument of Gamarnik [31]. The complete proof of the Theorem is provided in Section 7.3.

As a direct consequence of Theorem 5.1, we have the main result of this paper:

Theorem 5.2. UMW *is a throughput-optimal wireless broadcast policy.*

Proof. The total number of packets $R(t)$, received by all nodes in common up to time t may be bounded in terms of the physical queue lengths as follows:

$$A(0,t) - \sum_{i \in V} Q_i(t) \stackrel{(*)}{\leq} R(t) \leq A(0,t), \tag{18}$$

where the inequality $(*)$ follows from the observation that if a packet p has not reached at all nodes in the network, then at least one copy of it must be present in some physical queue.

Dividing both sides of Eqn. (18) by t, taking limits and using the Strong Law of Large Numbers and Theorem 5.1, we conclude that

$$\lim_{t\to\infty} \frac{R(t)}{t} = \lambda, \quad \text{w.p.1.}$$

Hence, from the definition (2.4), we conclude that UMW is throughput-optimal. □

5.2 Efficient Implementation

It is evident from the description of the UMW policy that the routing and node activation decisions are made using the virtual queue lengths $\tilde{Q}(t)$, whereas the physical packet scheduling decisions are based on the contents of the physical queues at each node. In the following, we discuss an efficient implementation of each of the three components in detail.

5.2.1 Routing

A broadcast route (MCDS) is computed for each packet immediately upon its arrival according to Eqn. (11), and copied into its header field. The route selection involves solving an MCDS problem with the nodes weighted by the corresponding virtual queue lengths, which is *NP*-hard [32]. This hardness result is consistent with the hardness of the wireless broadcasting problem proved earlier in Theorem 3.1. Assuming bi-directional wireless links, a polynomial time $\mathcal{O}(\log n)$ approximation algorithm for this problem is known for general graphs [33]. Furthermore, constant factor approximation algorithms for this problem are known for unit disk graphs [34].

5.2.2 Node Activation

At every slot a non-interfering subset of nodes is activated by choosing a maximum weight independent set in the conflict graph $\mathcal{C}(\mathcal{G})$, where the nodes are weighted by their corresponding virtual queue lengths, see Eqn. (12). The problem of finding a maximum weight independent set in a general graph is known to be *NP*-hard [32]. However, for the special case, such as unit disk graphs, constant factor approximation algorithms are available [35]. Note that, the same issue arises in the standard max-weight policies [36].

By a similar analysis, it can be shown that using an $\alpha \geq 1$ approximation algorithm for routing and $\beta \geq 1$ approximation algorithm for node activation, we can achieve $\frac{1}{\max(\alpha,\beta)}$ fraction of the optimal broadcast capacity of the network.

5.2.3 Packet Scheduling

The LTF policy can be efficiently implemented by maintaining a *min-heap* data-structure per node. The initial priority of each incoming packet at the source is set to zero. Once a packet p is received at a node i and the node i is included in its list of the required transmitting node, its priority is decreased by one, and it is inserted to the min-heap maintained at node i. Naturally, a node discards multiple receptions of the same packet.

5.3 Extensions

In this subsection, we briefly describe how to extend the proposed broadcasting policy in the case of (1) multiple source nodes, and (2) time-varying topology.

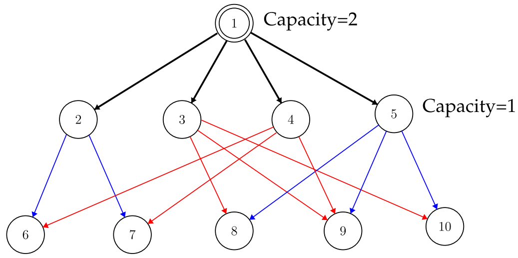


Fig. 4. A wireless network with non-interfering channels. The broadcast capacity of the network is $\lambda^* = 2$.

5.3.1 Multiple Source Nodes

The proposed policy works *as-it-is* for multiple source nodes in a static wireless network, and preserves its throughput-optimality. More explicitly,

1) An incoming packet to some source node s selects its broadcast-route by computing the Minimum-Weight Connected Dominating Set (MCDS) with s being the root node (Eqn. (11)).
2) The virtual queues are updated with the arrival of a packet at any source as in Eqn. (4). This requires either a centralized implementation or coordination among all source nodes.
3) The Max-Weight node activation policy remains unchanged (Eqn. (12)).
4) In the *LTF* physical packet scheduling policy, each packet computes its traversed hop-distances from its respective source.

By following essentially the same analysis, it can be shown that the proposed policy with multiple source nodes is also throughput-optimal.

5.3.2 Dynamic Networks

The problem becomes more challenging when the network topology changes frequently. In our numerical simulations, we consider an instance of *semi-dynamic* network where the nodes could be in either ON or OFF states, but the overall topology remains time-invariant. We compared the performance of the proposed policy against a standard heuristic broadcasting policy in Figs. 7 and 8.

We believe that under certain regularity assumptions on the nature of time-variation of the underlying topology (e.g., Markovian dynamics with a positive recurrent chain), a variation of the proposed broadcasting policy can be used. However, we postpone this generalization to future work.

6 Simulation Results

6.1 Interference-Free Network

As a proof of concept, we first simulate the *UMW* policy in a simple wireless network with known broadcast capacity. Consider the network shown in Fig. 4. Here node 1 is the source having a transmission capacity $C_1 = 2$. All other nodes in the network have unit transmission capacity. Assume that the channels are non-interfering, i.e., all nodes can transmit in a slot (this holds, e.g., if the nodes transmit on different frequencies). Since the broadcast capacity of any wireless network is upper-bounded by the capacity of the source, we readily have $\lambda^* \leq 2$. Also, it can be seen from

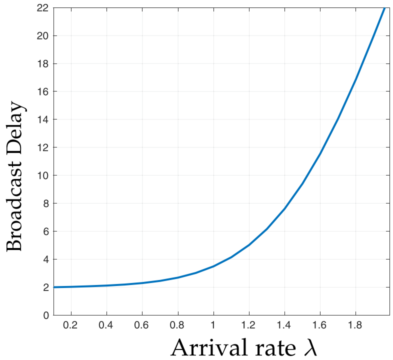


Fig. 5. Plot of the broadcast delay incurred by the UMW policy as a function of the arrival rate λ in the network shown in Fig. 4.

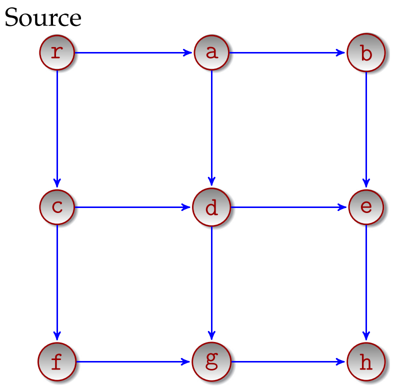


Fig. 6. A 3×3 wireless grid network with primary interference constraints. The wireless broadcasting capacity (λ^*) of the network is at most $\frac{1}{3}$ (Proposition 6.1).

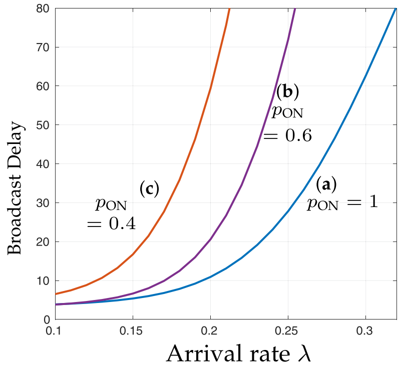


Fig. 7. Plot of the broadcast delay incurred by the UMW policy as a function of the arrival rate λ in the 3×3 wireless grid network shown in Fig. 6.

Fig. 4 that by transmitting the even numbered packets from nodes 2 and 5 (shown in blue) and the odd numbered packets from nodes 3 and 4, a broadcast rate of 2 packets per slot can be achieved. Hence, the broadcast capacity of the network is $\lambda^* = 2$.

Fig. 5 shows the average broadcast delay with the packet arrival rate λ in this network under the action of the proposed UMW policy. Note that the minimum delay is at least 2 as it takes at least two slots for any arriving packet to reach the nodes in the third layer. The plot confirms that the dynamic policy achieves the full broadcast capacity.

6.2 Networks with Interference Constraints

Consider the 3×3 wireless grid network, shown in Fig. 6. Assume that the transmissions are limited by primary interference constraints, i.e, two nodes cannot transmit together if the transmissions interfere at any node in the network. Assume that any node, if activated, has a transmission rate of one packet per slot. In this setting we have the following upper-bound on the broadcast capacity of the network.

> **Proposition 6.1.** *The broadcast capacity of the 3×3 grid network is at most $\frac{1}{3}$.*

The proof of the proposition is provided in Section 7.4.

In Fig. 7 we show the broadcast delay as a function of the packet arrival rate, under the action of the UMW policy on the right most curve marked (a). From the plot, we observe that the delay-throughput curve has a vertical asymptote approximately along the straightline $\lambda = \frac{1}{3}$. This, together with Lemma 6.1, immediately implies that the broadcast capacity of the network is $\lambda^* = \frac{1}{3}$ and confirms the throughput-optimality of the UMW policy.

6.2.1 Broadcasting in a Fleet of Moving Vehicles

Next, we simulate the performance of the UMW broadcasting policy on a fleet of vehicles moving together as a square grid network (Fig. 6). Due to the time-varying nature of wireless channels, the nodes (vehicles) are not always available for transmission. In particular, we assume a simplified model where each vehicle is active for potential transmission at a slot independently with some fixed but unknown probability p_{ON}. The controller takes scheduling decision by observing the current channel states. The delay performance of the proposed UMW broadcast policy is shown in Figs. 7b and 7c for two cases, $p_{\mathrm{ON}} = 0.6$ and $p_{\mathrm{ON}} = 0.4$ respectively. Following similar analysis as in the preceding sections, it can be shown that the UMW policy is also throughput-optimal for time-varying networks. Hence, from the plot it follows that the broadcast capacities of the time-varying 3×3 wireless grid network are ≈ 0.26 and ≈ 0.22 packets per slot, for the activity parameter $p_{\mathrm{ON}} = 0.6$ and $p_{\mathrm{ON}} = 0.4$ respectively.

6.2.2 Numerical Comparison with an Existing Wireless Broadcasting Protocol

In our final simulation experiment, we compare the performance of the proposed UMW broadcasting policy with the widely-studied Multipoint Relaying algorithm (MPR) [37], [38], which is also a part of the Optimized Link State Routing Protocol (OLSR) described in the RFC. The principal difference between the UMW and the MPR algorithm is that UMW selects the route (i.e., a connected dominating set or CDS) for each packet *dynamically* depending on the current network-wide loading condition, whereas the MPR algorithm broadcasts along a *fixed* CDS, specially chosen with 2-hop neighbor information. Due to space constraint, we do not give details of the MPR algorithm in this paper (see [9], [38]). In this simulation, we use the simple 2-hop topology, shown in Fig. 4, with the node 1 generating packets according to the Poisson distribution of rate λ packets per slot. The wireless transmissions are assumed

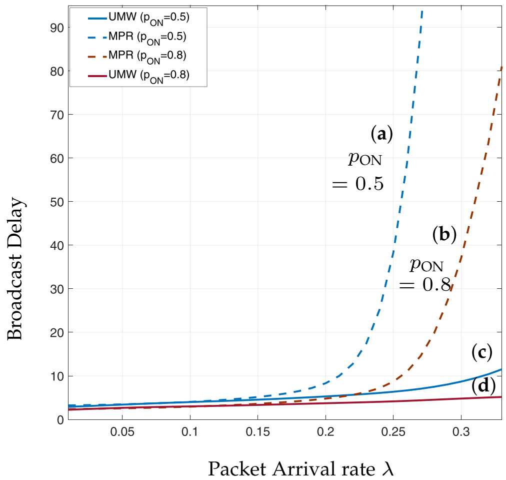


Fig. 8. Comparison of the broadcast delay incurred by the MPR and the UMW policy as a function of the arrival rate λ in the wireless network shown in Fig. 4 for two different values of $p_{\mathtt{ON}}$.

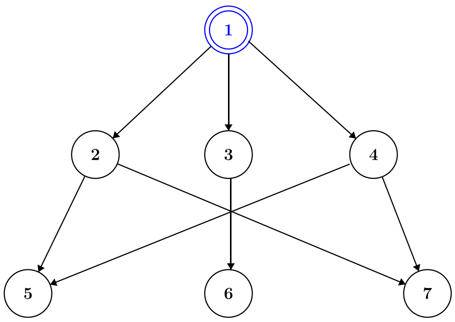


Fig. 9. Network Topology used for comparing Broadcasting with Point-to-Point and Point-to-Multipoint Transmissions

to be limited by the primary-interference constraint. The node-states are time-varying—a node is in ON state at a slot w.p. p_{ON}, and is in OFF state w.p. $1 - p_{\mathrm{ON}}$, independent of everything else. At the active state, a node can transmit one packet per-slot, subject to the interference constraint.

in Fig. 8, the dotted lines (marked (a) and (b)) and the solid lines (marked (c) and (d)) show the delay-performances of the MPR policy and the UMW policy respectively, as a function of the offered load λ, for two different values of p_{ON}. It can be observed that UMW outperforms MPR both in terms of delay and achievable throughput. The gain with the UMW policy becomes more prominent with heavy traffic as the UMW policy explores all feasible routing strategies by load-balancing among multiple CDS, whereas the MPR policy is constrained by a fixed (non-adaptive) routing strategy (single CDS) only.

6.2.3 Numerical Performance Comparison with the Algorithm in [19]

We pick a simple network topology, shown in Fig. 9, to illustrate the difference in the broadcasting efficiency for wireless networks with point-to-point links [15], and with point-to-multipoint links that come with the wireless broadcast advantage. We consider the following two scenarios:

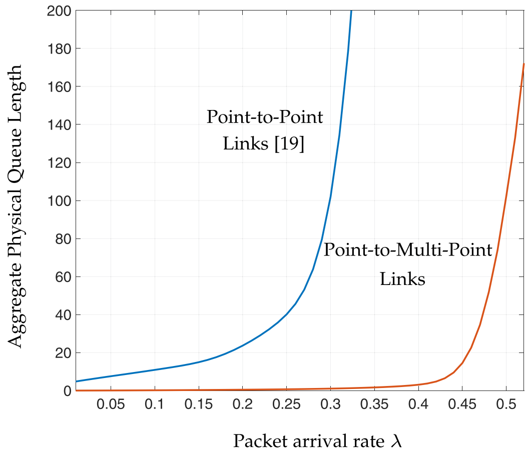


Fig. 10. Comparison of the broadcasting performance for wireless network in Fig. 9 with Point-to-Point and Point-to-Multi-Point transmissions.

Point-to-Point Links. With point-to-point wireless links, a directed edge from node a to node b signifies that node a can transmit packets to node b without error provided that there is no inter-channel interference. Point-to-Point wireless links arise in 5G mm-wave communication, e.g., when the transmitter beam-forms its transmitted signal to the receiver node to mitigate the high propagation loss prevalent at high frequencies. In the simulation result shown in Fig. 10, we assumed a primary interference constraint as in [15]. With this interference model, the set of all links which can be activated simultaneously is the set of all *Matchings* of the given network.

Point-to-Multipoint Links. With point-to-multipoint links, all out-neighbors of a transmitting node receives the same packet, provided no inter-channel interference. This type of wireless links are common in sub-6 GHz systems, e.g., 4G, and WiFi. In the plot shown in Fig. 10, we used the same node-activation constraint as discussed in Section 2, namely, two nodes a and b can simultaneously transmit at the same slot provided they do not interfere at any other common outgoing node.

Discussion. In the numerical simulation result shown in Fig. 10, we observe that broadcasting with point-to-multipoint links outperforms broadcasting with point-to-point links in that topology. This may be understood by noting that with point-to-point links, it takes three transmissions to disseminate a packet from the source node 1 to its out-neighbors (nodes 2, 3, and 4), whereas, with point-to-multipoint transmissions, all three out-neighbors of the source nodes receive a packet simultaneously from the source node in a single transmission.

7 Proof of Results

7.1 Proof of Hardness of the Wireless Broadcasting Problem

We start with the following lemma

Lemma 1. Wireless Broadcasting *is in* NP.

Proof. From the formulation, the problem Wireless *Broadcasting* is a Decision Problem. Also, if it has a Yes answer, then there is a scheduling algorithm which serves as a certificate. Hence the problem belongs to NP. □

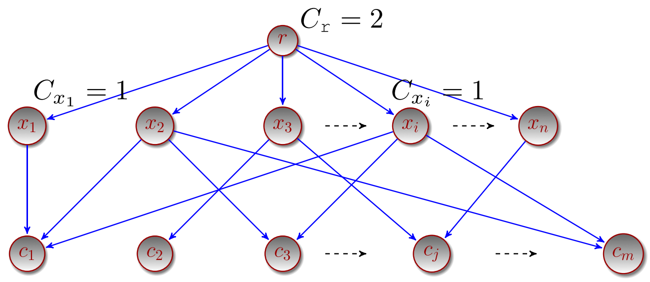


Fig. 11. The Gadget used for the hardness proof.

Next we show that an *NP*-complete problem, named Monotone Not All Equal 3-SAT (MNAE-3SAT) reduces to the problem Wireless Broadcasting in polynomial time. This will complete the reduction.

We begin with the description of the problem MNAE-3SAT:

7.1.1 The Problem MNAE-3SAT

- *INSTANCE:* Set U of boolean variables, collection $\mathcal{C}$ of clauses over U such that each clause $c \in \mathcal{C}$ has $|c| = 3$ variables and none of the clauses contain complemented variables (*Monotonicity*).
- *QUESTION:* Is there a truth assignment for U such that each clause in $\mathcal{C}$ has at least one true literal and at least one false literal ?

It is known that the problem MNAE $-$ 3SAT is *NP*-complete [24].

7.1.2 Reduction: MNAE-3SAT $\overset{\text{poly}}{\Rightarrow}$ Wireless Broadcasting

Suppose we are given an instance of the problem MNAE-3SAT $(U, \mathcal{C})$. Let $|U| = n$ and $|\mathcal{C}| = m$. Denote n boolean variables by $\{x_i, i = 1, 2, \ldots, n\}$. For this instance of MNAE-3SAT, we consider the following instance $\mathcal{G}(V, E)$ of Wireless Broadcasting as shown in Fig. 11. The construction is done as follows:

- There are a total of $n + m + 1$ nodes. The nodes are divided into three layers as shown in Fig. 11.
- Let $\mathtt{r} \in V$ be the source node in the first layer. The capacity of the source node is 2. This means that, the source node can transmit 2 packets per slot to its out-neighbours.
- There are n nodes in the second layer of the Fig. 11, all of which are out-neighbors of the source node $\mathtt{r}$. Each of these nodes correspond to a variable x_i in MNAE-3SAT instance. Capacity of each of these nodes in the second layer is one.
- There are m nodes in the third layer, each corresponding to a clause $c \in \mathcal{C}$ in the MNAE-3SAT instance. The edges incoming to a node c_j are defined as follows: if the clause c_j is expressed as $c_j = x_{i_1} \vee x_{i_2} \vee x_{i_3}$, then we add three edges $(x_{i_1}, c_j), (x_{i_2}, c_j), (x_{i_3}, c_j)$ in the graph $\mathcal{G}(V, E)$. Capacities of each node in the third layer is taken to be 1.

Now consider the following instance of Wireless *Broadcasting* on the constructed graph $\mathcal{G}(V, E)$. There are $M = 2$ packets at the source with a deadline of $T = 2$ slots. We claim that the following two questions are equivalent, i.e., Question 1 has a YES answer iff the Question 2 has a YES answer.

- *Question 1:* Is the MNAE-3SAT instance $(U, \mathcal{C})$ satisfiable ?
- *Question 2:* Does the constructed Wireless Broadcasting instance have a Yes Solution?

To show this, let us denote the packets sent by the source $\mathtt{r}$ at the beginning of the slot by $\{0, 1\}$. Since the capacity of the source $\mathtt{r}$ is 2, all nodes $x_1, x_2, \ldots, x_n$ receive this packet at every slot. Since the capacities of the nodes x_i is only unity, they can only transmit either packet 0 or the packet 1 at that slot. We can denote this choice by the binary variable x_i, i.e., $x_i = 0$ if the node x_i sends packet 0 and is 1 if it sends packet 1.

Note that the node c_j will receive both the packets if the corresponding clause contains at least one 0 and at least one 1. For a broadcast capacity of 2, all nodes must receive both packets at every slot. This is exactly the condition for the existence of a satisfying assignment of the MNAE-3SAT instance. This proves the intended hardness result.

Corollary. As a direct consequence of the above reduction, it follows that the problem Wireless Broadcasting remains *NP*-complete even with the following additional restrictions:

1) The wireless transmissions are non-interfering.
2) The graph $\mathcal{G}(V, E)$ is a DAG (c.f. [11], [13]).
3) The node capacities may take at most two values.
4) The in-degree of each node is at most 3.

7.2 Proof of Stability of the Virtual Queues

The proof of positive-recurrence and the sample path result is divided into several parts. First we describe and develop some general tools and then apply these tools to the virtual-queue Markov Chain $\{\tilde{Q}(t)\}_{t\geq 1}$.

7.2.1 Mathematical Tools

The key to our proof is a stronger version of the Foster-Lyapunov drift theorem, obtained by Hajek [28] in a more general context. The following statement of the result, quoted from [39], will suffice our purpose. First, we recall the definition of a Lyapunov function:

Definition 3 (Lyapunov Function). *Let $\mathcal{X}$ denote the state space of any process. We call a function $L : \mathcal{X} \to \mathbb{R}$ a* Lyapunov *function if the following conditions hold:*

- *(1) $L(x) \geq 0, \forall x \in \mathcal{X}$ and,*
- *(2) the set $S(M) = \{x \in \mathcal{X} : L(x) \leq M\}$ is finite for all finite M.*

Theorem A.1 (Hajek '82). *For an irreducible and aperiodic Markov Chain $\{X(t)\}_{t\geq 0}$ over a countable state space $\mathcal{X}$, suppose $L : \mathcal{X} \to \mathbb{R}_+$ is a Lyapunov function. Define the drift of L at X as*

$$\Delta L(X) \overset{\Delta}{=} \big(L(X(t+1)) - L(X(t))\big)\mathcal{I}(X(t) = X),$$

where $\mathcal{I}(\cdot)$ is the indicator function. Thus, $\Delta Z(X)$ is a random variable that measures the amount of change in the value of Z

in one step, starting from the state X. Assume that the drift satisfies the following two conditions:

- (C1) *There exists an $\epsilon > 0$ and a $B < \infty$ such that*

$$\mathbb{E}(\Delta L(X)|X(t) = X) \leq -\epsilon, \quad \forall X \in \mathcal{X} \quad \text{with} \quad Z(X) \geq B.$$

- (C2) *There exists a $D < \infty$ such that*

$$|\Delta L(X)| \leq D, \quad \text{w.p.} \quad 1, \quad \forall X \in \mathcal{X}.$$

Then, the Markov Chain $\{X(t)\}_{t\geq 0}$ is positive recurrent. Furthermore, there exists scalars $\theta^* > 0$ and a $C^* < \infty$ such that

$$\limsup_{t\to\infty} \mathbb{E}\big(\exp(\theta^* L(X(t))\big) \leq C^*.$$

We now establish the following technical lemma, which will be useful later.

Lemma 2. *Let $\{Y(t)\}_{t\geq 0}$ be a stochastic process taking values on the nonnegative real line. Supppose that there exists scalars $\theta^* > 0$ and $C^* < \infty$ such that*

$$\limsup_{t\to\infty} \mathbb{E}(\exp(\theta^* Y(t))) \leq C^*. \tag{19}$$

Then

$$Y(t) = \mathcal{O}(\log(t)), \qquad \text{w.p.1}$$

Proof. Define the positive constant $\eta^* = \frac{2}{\theta^*}$. We will show that

$$\mathbb{P}(Y(t) \geq \eta^* \log(t), \qquad infinitely\ often) = 0.$$

For this, define the event E_t as

$$E_t = \{Y(t) \geq \eta^* \log(t)\}. \tag{20}$$

From the given condition (19), we know that there exists a finite time t^* such that

$$\mathbb{E}(\exp(\theta^* Y(t))) \leq C^* + 1, \quad \forall t \geq t^*. \tag{21}$$

We can now upper-bound the probabilities of the events $E_t, t \geq t^*$ as follows:

$$\begin{aligned}
\mathbb{P}(E_t) &= \mathbb{P}(Y(t) \geq \eta^* \log(t)) \\
&= \mathbb{P}\big(\exp(\theta^* Y(t)) \geq \exp(\theta^* \eta^* \log(t))\big) \\
&\overset{(a)}{\leq} \frac{\mathbb{E}(\exp(\theta^* Y(t)))}{t^2} \\
&\overset{(b)}{\leq} \frac{C^* + 1}{t^2}.
\end{aligned}$$

The inequality (a) follows from the Markov inequality and the fact that $\theta^* \eta^* = 2$. The inequality (b) follows from Eqn. (21). Thus, we have

$$\begin{aligned}
\sum_{t=1}^{\infty} \mathbb{P}(E_t) &= \sum_{t=1}^{t^*-1} \mathbb{P}(E_t) + \sum_{t=t^*}^{\infty} \mathbb{P}(E_t) \\
&\leq t^* + (C^* + 1) \sum_{t=t^*}^{\infty} \frac{1}{t^2} \\
&\leq t^* + (C^* + 1) \frac{\pi^2}{6} < \infty.
\end{aligned}$$

Finally, using the Borel Cantelli Lemma, we conclude that

$$\mathbb{P}(\limsup Y_t \geq \eta^* \log t) = \mathbb{P}(E_t \quad \text{i.o.}) = 0.$$

This proves that $Y_t = \mathcal{O}(\log t), \text{w.p.1}$. □

Combining Theorem A.1 with Lemma 2, we have the following corollary

Corollary A.2. *Under the conditions (C1) and (C2) of Theorem A.1, we have*

$$L(X(t)) = \mathcal{O}(\log t), \qquad \text{w.p.1}$$

7.2.2 Construction of a Stationary Randomized Policy for the Virtual Queues $\{\tilde{Q}(t)\}_{t\geq 1}$

Let $\mathcal{D}$ denote the set of all Connected Dominating Sets (CDS) in the graph $\mathcal{G}$ containing the source $\mathtt{r}$. Since the broadcast rate $\lambda < \lambda^*$ is achievable by a stationary randomized policy, there exists such a policy π^* which executes the following:

- There exist non-negative scalars $\{a_i^*, i = 1, 2, \ldots, |\mathcal{D}|\}$ with $\sum_i a_i^* = \lambda$, such that each new incoming packet is routed independently along a CDS $D_i \in \mathcal{D}$ with probability $\frac{a_i^*}{\lambda}, \forall i$. The packet routed along the CDS D_i corresponds to an arrival to the virtual queues $\{Q_j, j \in D_i\}$.

  As a result, packets arrive to the virtual queue Q_j i.i.d. at an expected rate of $\sum_{i:j\in D_i} a_j^*, \forall j$ per slot.
- A feasible schedule $\boldsymbol{s}_j \in \mathcal{M}$ is selected for transmission with probability p_j $j = 1, 2, \ldots, k$ i.i.d. at every slot. By Caratheodory's theorem, the value of k can be restricted to at most $n + 1$. This results in the following expected service rate vector from the virtual queues:

$$\boldsymbol{\mu}^* = \sum_{j=1}^{n+1} p_j c_j \boldsymbol{s}_j.$$

Since each of the virtual queues must be stable under the action of the policy π^*, from the theory of the GI/GI/1 queues, we know that there exists an $\epsilon > 0$ such that

$$\mu_i^* - \sum_{j:i\in D_j} a_j^* \geq \epsilon, \qquad \forall i \in V. \tag{22}$$

Next, we will verify that the conditions *C1* and *C2* in Theorem A.1 holds for the Markov Chain of the virtual queues $\{\tilde{\boldsymbol{Q}}(t)\}_{t\geq 1}$ under the action of the *UMW* policy, with the Lyapunov function $L(\tilde{\boldsymbol{Q}}(t)) = ||\tilde{\boldsymbol{Q}}(t)||$ at any arrival rate $\lambda < \lambda^*$.

7.2.3 Verification of Condition (C1)- Negative Expected Drift

From the definition of the policy *UMW*, it minimizes the RHS of the drift upper-bound (10) from the set of all feasible policies Π. Hence, we can upper-bound the conditional drift of the *UMW* policy by comparing it with the stationary policy π^* described in Section 7.2.2 as follows:

$$\begin{aligned}
&\mathbb{E}(\Delta^{\mathbf{UMW}}(t)|\tilde{\boldsymbol{Q}}(t)=\tilde{\boldsymbol{Q}})\\
&\leq \frac{1}{2||\tilde{\boldsymbol{Q}}||}\Big(B+2\sum_{i\in V}\tilde{Q}_i(t)\mathbb{E}\big(A_i^{\mathrm{UMW}}(t)|\tilde{\boldsymbol{Q}}(t)=\tilde{\boldsymbol{Q}}\big)\\
&\quad -2\sum_{i\in V}\tilde{Q}_i(t)\mathbb{E}\big(\mu_i^{\mathrm{UMW}}(t)|\tilde{\boldsymbol{Q}}(t)=\tilde{\boldsymbol{Q}}\big)\Big)\\
&\overset{(a)}{\leq} \frac{1}{2||\tilde{\boldsymbol{Q}}||}\Big(B+2\sum_{i\in V}\tilde{Q}_i(t)\mathbb{E}\big(A_i^{\pi^*}(t)|\tilde{\boldsymbol{Q}}(t)=\tilde{\boldsymbol{Q}}\big)\\
&\quad -2\sum_{i\in V}\tilde{Q}_i(t)\mathbb{E}\big(\mu_i^{\pi^*}(t)|\tilde{\boldsymbol{Q}}(t)=\tilde{\boldsymbol{Q}}\big)\Big)\\
&= \frac{1}{2||\tilde{\boldsymbol{Q}}||}\Big(B-2\sum_{i\in V}\tilde{Q}_i(t)\big(\mathbb{E}\mu_i^*(t)-\mathbb{E}A_i^*(t)\big)\Big)\\
&= \frac{1}{2||\tilde{\boldsymbol{Q}}||}\Big(B-2\sum_{i\in V}\tilde{Q}_i(t)\big(\mu_i^*-\sum_{j:i\in D_j}a_j^*\big)\Big)\\
&\overset{(b)}{\leq} \frac{B}{2||\tilde{\boldsymbol{Q}}||}-\epsilon\frac{\sum_{i\in V}\tilde{Q}_i(t)}{||\tilde{\boldsymbol{Q}}||},
\end{aligned} \tag{23}$$

where inequality (a) follows from the definition of the *UMW* policy and inequality (b) follows from the stability property of the randomized policy given in Eqn. (22). Since the virtual-queue lengths $\tilde{\boldsymbol{Q}}(t)$ is a non-negative vector, it is easy to see that (e.g., by squaring both sides)

$$\sum_{i\in V}\tilde{Q}_i(t)\geq\sqrt{\sum_i\tilde{Q}_i^2(t)}=||\tilde{\boldsymbol{Q}}||.$$

Hence, from Eqn. (23) in the above chain of inequalities, we obtain

$$\mathbb{E}(\Delta^{\mathbf{UMW}}(t)|\tilde{\boldsymbol{Q}}(t)=\tilde{\boldsymbol{Q}})\leq\frac{B}{2||\tilde{\boldsymbol{Q}}||}-\epsilon. \tag{24}$$

Thus

$$\mathbb{E}(\Delta^{\mathbf{UMW}}(t)|\tilde{\boldsymbol{Q}}(t))\leq-\frac{\epsilon}{2},\qquad\forall||\tilde{\boldsymbol{Q}}||\geq B/\epsilon.$$

This verfies the negative expected drift condition *C1* in Theorem A.1.

7.2.4 Verification of Condition (C2)- Almost Surely Bounded Drift

To show that the magnitude of one-step drift $|\Delta L(\tilde{\boldsymbol{Q}})|$ is almost surely bounded, we compute

$$\begin{aligned}
|\Delta L(\tilde{\boldsymbol{Q}}(t))|&=|L(\tilde{\boldsymbol{Q}}(t+1))-L(\tilde{\boldsymbol{Q}}(t))|\\
&=\big|||\tilde{\boldsymbol{Q}}(t+1)||-||\tilde{\boldsymbol{Q}}(t))||\big|.
\end{aligned}$$

Now, from the dynamics of the virtual queues (4), we have for any virtual queue $\tilde{Q}_i$

$$|\tilde{Q}_i(t+1)-\tilde{Q}_i(t)|\leq|A_i(t)-\mu_i(t)|.$$

Thus

$$||\tilde{\boldsymbol{Q}}(t+1)-\tilde{\boldsymbol{Q}}(t)||\leq||\boldsymbol{A}(t)-\boldsymbol{\mu}(t)||\leq\sqrt{n}(A_{\max}+c_{\max}).$$

Hence, using the triangle inequality for the ℓ_2 norm, we obtain

$$|\Delta L(\tilde{\boldsymbol{Q}}(t))|=\big|||\tilde{\boldsymbol{Q}}(t+1)||-||\tilde{\boldsymbol{Q}}(t))||\big|\leq\sqrt{n}(A_{\max}+c_{\max}),$$

which verifies the condition *C2* of Theorem A.1.

7.2.5 Almost Sure Bound on Virtual Queue Lengths

Finally, we invoke Corollary A.2 to conclude that

$$\limsup_t||\tilde{\boldsymbol{Q}}(t)||=\mathcal{O}(\log t),\qquad\text{w.p.1.}$$

This implies that

$$\max_i\tilde{\boldsymbol{Q}}_i(t)=\mathcal{O}(\log t),\qquad\text{w.p.1.}$$

7.3 Proof of Theorem 5.1

Throughout this proof, we will consider only the typical sample point $\omega\in\mathcal{E}$ defined in Eqn. (13). For the sake of notational simplicity, we will drop the argument ω for evaluating a random variable X at ω, i.e., the deterministic sample path $X(\omega,t),\omega\in\mathcal{E}$ will be simply denoted by $X(t)$. We now make a simple observation which will be useful in the proof of the theorem:

Lemma 3. *Consider a function $F:\mathbb{Z}_+\to\mathbb{Z}_+$, where $\mathbb{Z}_+$ is the set of non-negative integers. Assume that $F(t)=\mathcal{O}(\log t)$. Define $M(t)=\sup_{0\leq\tau\leq t}F(\tau)$. Then,*

1) *$M(t)$ is non-decreasing in t and $M(t)\geq F(t)$.*
2) *$M(t)=\mathcal{O}(\log t)$.*

Proof. Clearly, $M(t)\sup_{0\leq\tau\leq t}F(\tau)\geq F(t)$ and

$$M(t+1)=\sup_{0\leq\tau\leq t+1}F(\tau)\geq\sup_{0\leq\tau\leq t}F(\tau)=M(t).$$

To prove the second claim, let $t_{\max}(t)=\arg\max_{0\leq\tau\leq t}F(\tau)$. Clearly, $t_{\max}(t)\leq t$. Hence, for large enough t, we have

$$M(t)=F(t_{\max}(t))=\mathcal{O}(\log t_{\max}(t))=\mathcal{O}(\log t).$$

□

As a consequence of Lemma 3 applied to Eqn. (17), we have almost surely

$$A_i(t_0,t)\leq S_i(t_0,t)+M(t),\qquad\forall i\in V,\quad\forall t_0\leq t, \tag{25}$$

for some non-decreasing function $M(t)=\mathcal{O}(\log t)$. We now return to the proof of the main result, Theorem 5.1.

Proof. Our proof technique is inspired by an adversarial queueing theory argument, given in [31]. We remind the reader that we are analyzing a typical sample path satisfying Eqn. (25), which holds almost surely. In the following argument, each copy of a packet is counted separately.

Without any loss of generality, assume that we start from an empty network at time $t=0$. Let $R_k(t)$ denote the total number of packets waiting to be transmitted further at time t, which have already been forwarded *exactly* k

times by the time t. We call such packets "layer k" packets. As we have mentioned earlier, if a packet is duplicated multiple times along its assigned route D (which is a connected dominating set (or CDS, in short)), each copy of the packet is counted separately in the variable $R_k(t)$, i.e.,

$$R_k(t) = \sum_{D \in \mathcal{D}} \sum_{i \in D_k} R_{(i,D)}(t), \tag{26}$$

where the variable $R_{(i,D)}(t)$ denotes the number of packets following the CDS D, that are waiting to be transmitted by the node $i \in D$ at time t and D_k is the set of nodes in the CDS D, which are exactly kth hop away from the source along the CDS D. We show by induction that $R_k(t)$ is *almost surely* bounded by a function, which is $\mathcal{O}(\log t)$.

Base Step $k = 0$. Consider the source node $i = \mathtt{r}$ and an arbitrary time t. Let $t_0 \leq t$ be the largest time at which no packets of layer 0 (packets which are present only at the source and have never been transmitted before) were waiting to be transmitted by the source. If no such time exists, set $t_0 = 0$. During the time interval $(t_0, t]$, as a consequence of the property in Eqn. (25) of the *UMW* policy, at most $S_{\mathtt{r}}(t_0, t) + M(t)$ external packets have arrived to the source $\mathtt{r}$ for broadcasting. Also, by the choice of the time t_0, the source node $\mathtt{r}$ was always having packets to transmit during the entire time interval $(t_0, t]$. Since LTF packet scheduling policy is followed in the physical network, layer 0 packets have priority over all other packets (in fact, there is packet of other layers present *only* at the source, but this is not the case at other nodes which we will consider in the induction step). Hence, it follows that the total number of layer 0 packets at time t satisfies

$$\begin{aligned} R_0(t) = \sum_{D \in \mathcal{D}} \sum_{i \in D_0} R_{(i,D)}(t) &\leq S_{\mathtt{r}}(t_0, t) + M(t) - S_{\mathtt{r}}(t_0, t) \\ &\leq M(t). \end{aligned} \tag{27}$$

Define $B_0(t) \stackrel{\text{def}}{=} M(t)$. Since $M(t) = \mathcal{O}(\log t)$, we have $B_0(t) = \mathcal{O}(\log t)$. Note that, since $M(t)$ is non-decreasing from Lemma (3), so is $B_0(t)$.

Induction Step. As our inductive assumption, suppose that, for some non-decreasing functions $B_j(t) = \mathcal{O}(\log t)$, $j = 0, 1, 2, \ldots, k-1$, we have $R_j(t) \leq B_j(t)$, for all time t. We next show that there exists a non-decreasing function $B_k(t) = \mathcal{O}(\log t)$ such that $R_k(t) \leq B_k(t)$ for all time t.

To prove the above assertion, fix a node i and an arbitrary time t. Let $t_0 \leq t$ denote the largest time before t, such that there were no layer k packets waiting to be transmitted by the node i. Set $t_0 = 0$ if no such time exists. Hence the node i was always having packets to transmit during the time interval $(t_0, t]$ (packets in layer k or lower). The layer k packets that wait to be transmitted by the node i at time t are composed only of a subset of packets which were in layers $0 \leq j \leq k-1$ at time t_0 or packets that arrived during the time interval $(t_0, t]$ and include the node i as *one of their kth transmitter* along the route followed. By our induction assumption, the first group of packets has a size bounded by $\sum_{j=0}^{k-1} B_j(t_0) \leq \sum_{j=0}^{k-1} B_j(t)$, where we have used the fact (using our induction step) that the functions $B_j(\cdot)$'s are monotonically non-decreasing. The size of the second group of packets is given by $\sum_{D: i \in D_k} A_D(t_0, t)$. We next estimate the number of layer k packets that crossed the edge e during the time interval $(t_0, t]$. Since the LTF packet scheduling policy is used in the physical network, layer k packets were not processed only when there were packets in layers up to $k-1$ that included the node i in its routing CDS. The number of such packets is bounded by $\sum_{j=0}^{k-1} B_j(t_0) \leq \sum_{j=0}^{k-1} B_j(t)$, which denotes the total possible number of packets in layers up to $k-1$ at time t_0, *plus* $\sum_{j=0}^{k-1} \sum_{D: i \in D_j} A_D(t_0, t)$, which is the number of new packets that arrived in the interval $(t_0, t]$ and includes the node i as a transmitter within their first $k-1$ hops. Thus, we conclude that at least

$$\max\left\{0, S_i(t_0, t) - \sum_{j=0}^{k-1} B_j(t) - \sum_{j=0}^{k-1} \sum_{D: i \in D_j} A_D(t_0, t)\right\}, \tag{28}$$

packets of layer k have been transmitted by the node i during the time interval $(t_0, t]$. Hence, the total number of layer k packets present at node i at time t is given as

$$\begin{aligned} \sum_{D: i \in D_k} R_{(i,D)}(t) &\leq \sum_{j=0}^{k-1} B_j(t) + \sum_{D: i \in D_k} A_D(t_0, t) \\ &- \big(S_i(t_0, t) - \sum_{j=0}^{k-1} B_j(t) - \sum_{j=0}^{k-1} \sum_{D: i \in D_j} A_D(t_0, t)\big) \\ &= 2 \sum_{j=0}^{k-1} B_j(t) + \sum_{j=0}^{k} \sum_{D: i \in D_j} A_D(t_0, t) - S_i(t_0, t) \\ &\stackrel{(a)}{\leq} 2 \sum_{j=0}^{k-1} B_j(t) + A_i(t_0, t) - S_i(t_0, t) \\ &\stackrel{(b)}{\leq} 2 \sum_{j=0}^{k-1} B_j(t) + M(t), \end{aligned}$$

where the inequality (a) follows from the fact that each packet gets routed to a node i for transmission only once and hence

$$A_i(t_0, t) = \sum_{j=0}^{n-1} \sum_{D: i \in D_j} A_D(t_0, t), \qquad \forall i \in V.$$

The inequality (b) follows from the property of the typical sample paths, stated in Eqn. (25). Hence, the total number of layer k packets at time t is bounded as

$$R_k(t) = \sum_i \sum_{D: i \in D_k} R_{(i,D)}(t) \leq 2n \sum_{j=0}^{k-1} B_j(t) + nM(t).$$

Define $B_k(t)$ to be the RHS of the above equation, i.e.,

$$B_k(t) \stackrel{(\text{def})}{=} 2n \sum_{j=0}^{k-1} B_j(t) + nM(t). \tag{29}$$

Using our induction assumption and Eqn. (29), we conclude that $B_k(t) = \mathcal{O}(\log t)$, and it is easily seen to be non-decreasing. This completes the proof of the induction step.

To conclude the proof of the theorem, observe that the sum of the lengths of the physical queues at time t may be alternatively written as

$$\sum_{i\in V} Q_i(t) = \sum_{k=1}^{n-1} R_k(t). \tag{30}$$

Since the previous inductive argument shows that for all k, we have $R_k(t) \leq B_k(t)$ where $B_k(t) = \mathcal{O}(\log t)$ *a.s.*, we have $\sum_{i\in V} Q_i(t) = \mathcal{O}(\log t)$, and hence

$$\lim_{t\to\infty} \frac{\sum_{i\in V} Q_i(t)}{t} = 0, \qquad \text{w.p. } 1. \tag{31}$$

□

7.4 Proof of Proposition 6.1

Proof. Observe that, due to the primary interference constraints, the nodes $\mathtt{r}, a$ and d can not be activated at the same slot. Consider any arbitrary policy, which activates the node i for a fraction $f_i, i \in V$ times. Hence, we have the constraint that

$$f_{\mathtt{r}} + f_a + f_c \leq 1. \tag{32}$$

On the other hand, if the policy π achieves a broadcast rate of λ, it must be that

$$\begin{aligned} &\lambda \leq f_{\mathtt{r}}, && \text{(considering broadcast rate at node } a) \\ &\lambda \leq f_a, && \text{(considering broadcast rate at node } b) \\ &\lambda \leq f_c, && \text{(considering broadcast rate at node } f). \end{aligned}$$

Adding the above three equations, we have

$$3\lambda \leq f_{\mathtt{r}} + f_a + f_c \overset{(a)}{\leq} 1, \tag{33}$$

where the inequality (a) follows from the constraint (32). Since the policy π is assumed to be arbitrary, we conclude that the broadcast capacity of the 3×3 grid network is at most $\frac{1}{3}$. □

8 Conclusion

In this paper, we obtained the first throughput-optimal broadcasting policy for wireless networks with point-to-multipoint links and arbitrary scheduling constraints. The policy is derived using the powerful framework of *precedence-relaxed virtual network*, which we used earlier for designing throughput-optimal policies for networks with point-to-point links. Packet routing and scheduling decisions are made by solving standard optimization problems on the network, weighted by the virtual queue lengths. The policy is proved to be throughput optimal by a combination of the Lyapunov method and a sample path argument using adversarial queueing theory. Extensive simulation results demonstrate the efficiency of the proposed policy in both static and dynamic network settings. There exist several interesting directions to extend this work. First, in our simplified model, we assumed that interference-free wireless transmissions are also error-free. A more accurate wireless channel model would incorporate the possibility of packet losses associated with individual receiving nodes, due to fading and receiver noise [12]. Second, it remains unknown whether the UMW policy is still throughput optimal if the routing and node activations are made using the corresponding *physical queue* lengths as compared to the virtual queues. A positive result in this direction would lead to a more efficient implementation.

Acknowledgments

The first author would like to thank the P.G. Senapathy Center for Computing Resource at IIT Madras for providing the Virgo Super Cluster facility where a part of the numerical simulation work was carried out. The first author was supported by Qualcomm gift IND-417880 and the second author was supported by US National Science Foundation Grant CNS-1217048, CNS-1524317, ONR Grant N00014-12-1-0064, and ARO MURI Grant W911NF-08-1-0238.

References

[1] W. Na, N.-N. Dao, and S. Cho, "Reliable multicasting service for densely deployed military sensor networks," *Int. J. Distrib. Sensor Netw.*, vol. 2015, p. 6:6, Jan. 2015.

[2] W. R. Heinzelman, J. Kulik, and H. Balakrishnan, "Adaptive protocols for information dissemination in wireless sensor networks," in *Proc. 5th Annu. ACM/IEEE Int. Conf. Mobile Comput. Netw.*, 1999, pp. 174–185.

[3] M. Dietzfelbinger, "Gossiping and broadcasting versus computing functions in networks," *Discrete Appl. Math.*, vol. 137, no. 2, pp. 127–153, 2004.

[4] R. Chen, W.-L. Jin, and A. Regan, "Broadcasting safety information in vehicular networks: Issues and approaches," *IEEE Netw.*, vol. 24, no. 1, pp. 20–25, Jan./Feb. 2010.

[5] I. Maric and R. D. Yates, "Cooperative multihop broadcast for wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 22, no. 6, pp. 1080–1088, Aug. 2004.

[6] R. Gandhi, A. Mishra, and S. Parthasarathy, "Minimizing broadcast latency and redundancy in ad hoc networks," *IEEE/ACM Trans. Netw.*, vol. 16, no. 4, pp. 840–851, Aug. 2008.

[7] G. K. W. Wong, H. Liu, X. Chu, Y.-W. Leung, and C. Xie, "Efficient broadcasting in multi-hop wireless networks with a realistic physical layer," *Ad Hoc Netw.*, vol. 11, no. 4, pp. 1305–1318, 2013.

[8] L. Massoulie, A. Twigg, C. Gkantsidis, and P. Rodriguez, "Randomized decentralized broadcasting algorithms," in *Proc. 26th IEEE Int. Conf. Comput. Commun.*, 2007, pp. 1073–1081.

[9] B. Williams and T. Camp, "Comparison of broadcasting techniques for mobile ad hoc networks," in *Proc. 3rd ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2002, pp. 194–205.

[10] S. Sarkar and L. Tassiulas, "A framework for routing and congestion control for multicast information flows," *IEEE Trans. Inf. Theory*, vol. 48, no. 10, pp. 2690–2708, Oct. 2002.

[11] A. Sinha, G. Paschos, C.-P. Li, and E. Modiano, "Throughput-optimal broadcast on directed acyclic graphs," in *Proc. IEEE Conf. Comput. Commun.*, 2015, pp. 1248–1256.

[12] A. Sinha, G. Paschos, C.-P. Li, and E. Modiano, "Throughput-optimal multihop broadcast on directed acyclic wireless networks," *IEEE/ACM Trans. Netw.*, vol. 25, no. 1, pp. 377–391, Feb. 2017.

[13] A. Sinha, G. Paschos, and E. Modiano, "Throughput-optimal multi-hop broadcast algorithms," in *Proc. 17th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2016, pp. 51–60.

[14] D. Towsley and A. Twigg, "Rate-optimal decentralized broadcasting: The wireless case," in *Proc. ACITA*, Citeseer, 2008, pp. 323–333.

[15] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. Inf. Theory*, vol. 46, no. 4, pp. 1204–1216, Jul. 2000.

[16] T. Ho and H. Viswanathan, "Dynamic algorithms for multicast with intra-session network coding," *IEEE Trans. Inf. Theory*, vol. 55, no. 2, pp. 797–815, Feb. 2009.

[17] M. Médard and A. Sprintson, *Network Coding: Fundamentals and Applications*. New York, NY, USA: Academic, 2011.

[18] T. Ho and D. Lun, *Network Coding: An Introduction*. Cambridge, U.K.: Cambridge Univ. Press, 2008.

[19] A. Sinha and E. Modiano, "Optimal control for generalized network-flow problems," *IEEE/ACM Trans. Netw.*, vol. 26, no. 1, pp. 506–519, Feb. 2018.

[20] A. Sinha and E. Modiano, "Throughput-optimal broadcast in wireless networks with point-to-multipoint transmissions," in *Proc. 18th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2017, pp. 3:1–3:10.

[21] K. Jain, J. Padhye, V. N. Padmanabhan, and L. Qiu, “Impact of interference on multi-hop wireless network performance,” *Wireless Netw.*, vol. 11, no. 4, pp. 471–487, 2005.

[22] D. B. West, et al., *Introduction to Graph Theory*, vol. 2. Upper Saddle River, NJ, USA: Prentice Hall, 2001.

[23] R. Rustin, *Combinatorial Algorithms*. Algorithmics Press, New York, 1973.

[24] T. J. Schaefer, “The complexity of satisfiability problems,” in *Proc. 10th Annu. ACM Symp. Theory Comput.*, 1978, pp. 216–226.

[25] D. Shah, N. C. David, and J. N. Tsitsiklis, “Hardness of low delay network scheduling,” *IEEE Trans. Inf. Theory*, vol. 57, no. 12, pp. 7810–7817, Dec. 2011.

[26] A. E. F. Clementi, P. Crescenzi, P. Penna, G. Rossi, and P. Vocca, “On the complexity of computing minimum energy consumption broadcast subgraphs,” in *Proc. Annu. Symp. Theoretical Aspects Comput. Sci.*, 2001, pp. 121–131.

[27] J. K. Lenstra and A. H. G. Rinnooy Kan, “Complexity of scheduling under precedence constraints,” *Operations Res.*, vol. 26, no. 1, pp. 22–35, 1978.

[28] B. Hajek, “Hitting-time and occupation-time bounds implied by drift analysis with applications,” *Advances Appl. Probability*, vol. 14, pp. 502–525, 1982.

[29] M. J. Neely, “Stochastic network optimization with application to communication and queueing systems,” *Synthesis Lectures Commun. Netw.*, vol. 3, no. 1, pp. 1–211, 2010.

[30] L. Kleinrock, *Queueing Systems: Theory*, vol. 1. Hoboken, NJ, USA: Wiley, 1975.

[31] D. Gamarnik, “Stability of adaptive and non-adaptive packet routing policies in adversarial queueing networks,” in *Proc. 31st Annu. ACM Symp. Theory Comput.*, 1999, pp. 206–214.

[32] R. G. Michael and S. J. David, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. San Francisco, CA, USA: Freeman, 1979.

[33] S. Guha and S. Khuller, “Approximation algorithms for connected dominating sets,” *Algorithmica*, vol. 20, no. 4, pp. 374–387, 1998.

[34] M. T. Thai and D.-Z. Du, “Connected dominating sets in disk graphs with bidirectional links,” *IEEE Commun. Lett.*, vol. 10, no. 3, pp. 138–140, Mar. 2006.

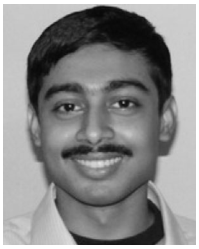
[35] G. K. Das, M. De, S. Kolay, S. C. Nandy, and S. Sur-Kolay, “Approximation algorithms for maximum independent set of a unit disk graph,” *Inf. Process. Lett.*, vol. 115, no. 3, pp. 439–446, 2015.

[36] L. Tassiulas and A. Ephremides, “Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks,” *IEEE Trans. Autom. Control*, vol. 37, no. 12, pp. 1936–1948, Dec. 1992.

[37] A. Qayyum, L. Viennot, and A. Laouiti, “Multipoint relaying for flooding broadcast messages in mobile wireless networks,” in *Proc. 35th Annu. Hawaii Int. Conf. Syst. Sci.*, 2002, pp. 3866–3875.

[38] T. Clausen and P. Jacquet, “Optimized link state routing protocol (OLSR),” RFC Editor, 2003.

[39] A. Eryilmaz and R. Srikant, “Asymptotically tight steady-state queue length bounds implied by drift conditions,” *Queueing Syst.*, vol. 72, no. 3/4, pp. 311–359, 2012.



Abhishek Sinha received the BE degree in electronics and telecommunication engineering from Jadavpur University, Kolkata, India, the ME degree in telecommunication engineering from the Indian Institute of Science, Bangalore, and the PhD degree from the Massachusetts Institute of Technology, in 2017, where he worked with the Laboratory for Information and Decision Systems (LIDS). He is currently an assistant professor with the Department of Electrical Engineering, IIT Madras. After his PhD, he worked as a senior engineer with Qualcomm Research, San Diego. He is a recipient of several awards including the Best Paper Award in INFOCOM 2018, the Best Paper Award in MobiHoc 2016, Prof. Jnansaran Chatterjee memorial gold medal and T. P. Saha Memorial gold centered silver medal from Jadavpur University, and Jagadis Bose National Science Talent Search (JBNSTS) scholarship, Kolkata, India. His areas of interests include network control, information theory, machine learning, and applied probability.

Eytan Modiano received the BS degree in electrical engineering and computer science from the University of Connecticut at Storrs, in 1986, and the MS and PhD degrees both in electrical engineering from the University of Maryland, College Park, Maryland, in 1989 and 1992, respectively. He was a naval research laboratory fellow between 1987 and 1992 and a national research council post doctoral fellow during 1992-1993. Between 1993 and 1999, he was with MIT Lincoln Laboratory. Since 1999, he has been on the faculty with MIT, where he is a professor and associate department head with the Department of Aeronautics and Astronautics, and associate director of the Laboratory for Information and Decision Systems (LIDS). His research is on communication networks and protocols with emphasis on satellite, wireless, and optical networks. He is the co-recipient of the INFOCOM 2018 Best Paper Award, the MobiHoc 2016 Best Paper Award, the Wiopt 2013 Best Paper Award, and the Sigmetrics 2006 Best Paper Award. He is the editor-in-chief of the *IEEE/ACM Transactions on Networking*, and served as associate editor of the *IEEE Transactions on Information Theory* and the *IEEE/ACM Transactions on Networking*. He was the technical program co-chair for IEEE Wiopt 2006, IEEE Infocom 2007, ACM MobiHoc 2007, and DRCN 2015. He is a fellow of the IEEE and an associate fellow of the AIAA, and served on the IEEE Fellows committee.