

Parallel TDMA Scheduling for Multiple-Destination Wireless Networks

Gam D. Nguyen, Sastry Kompella, *Member, IEEE*, Jeffrey E. Wieselthier, *Fellow, IEEE*, and Anthony Ephremides, *Fellow, IEEE*

Abstract—We study transmission strategies in a multiple-source, multiple-destination wireless network. Each source transmits packets that are intended for a particular destination. However, a transmitted packet can cause interference at other destinations. Our primary performance measure is throughput, which we define to be the average number of packets that are successfully received per intended destination per time slot. The sources are first divided into groups, based on the intended destination of their packets. In our parallel method, each group operates according to its own local protocol (e.g., TDMA), concurrently with and independently of the other groups. Our results show the impact of transmission schedules, channel fading, receiver noise, and other-user interference on network performance. We then show that, for given channel statistics and topology configurations, the network performance can be significantly improved when the groups in the network coordinate their transmissions according to an optimal schedule. Further, in many cases, even the use of randomly generated parallel schedules can provide considerably higher performance than traditional TDMA.

Index Terms—Parallel TDMA, sequential TDMA, transmission scheduling, random schedules, distributed implementation, interference.

I. INTRODUCTION

WE study a multiple-source, multiple-destination wireless network, in which each of the K sources transmits to a designated one of the N destinations. The network operates in the presence of detrimental effects such as channel fading, attenuation, noise, and other-user interference (i.e., a node's transmission may cause interference at non-intended destinations). An example is a wireless sensor network, which consists of K sensor nodes transmitting data to N collection centers. Fig. 1 shows such a network in which $K = 15$ sources transmit to $N = 3$ destinations.

In this paper, we study a simple method for scheduling the transmissions between the sources and their destinations. First, we organize the network nodes into groups, which are defined based on the designated destination. Each group then operates

Manuscript received December 27, 2010; revised June 23, 2011; accepted August 29, 2011. The associate editor coordinating the review of this paper and approving it for publication was G. Xue.

This work was supported by the Office of Naval Research.

G.D. Nguyen and S. Kompella are with the Information Technology Division, Naval Research Laboratory, Washington, DC 20375 USA (e-mail: {gam.nguyen, sastry.kompella}@nrl.navy.mil).

J.E. Wieselthier is with Wieselthier Research, Silver Spring, MD 20901 USA (e-mail: jeff@wieselthier.com).

A. Ephremides is with the Electrical and Computer Engineering Department, University of Maryland, College Park, MD 20742 USA (e-mail: etony@umd.edu).

Digital Object Identifier 10.1109/TWC.2011.092711.102300

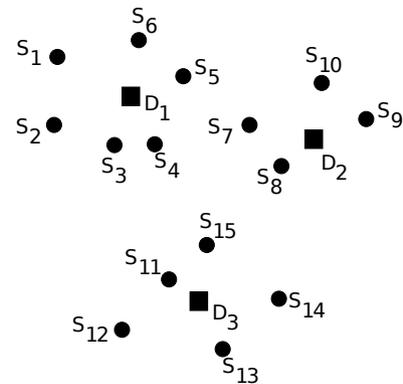


Fig. 1. A wireless network with 15 sources and 3 destinations.

according to a local TDMA-based protocol, concurrently with and independently of the other groups. A schedule is a rule that specifies which nodes are allowed to transmit in each time slot. We then develop methods for evaluating the throughput performance of this “parallel” method for an arbitrary schedule. Our results show the impact of channel statistics, receiver noise, and interference on network performance. Further, for a wide range of system parameters, randomly generated parallel schedules can provide considerably higher throughput than “sequential” TDMA (which permits only one source to transmit at a time). We next show that, in many cases, the network performance is improved significantly if the sources operate according to an optimized schedule (under which the sources coordinate their transmissions with sources in other groups).

Here, we study TDMA-based scheduling for a multiple-source multiple-destination wireless network that operates under the heavy-traffic condition, in which each source always has traffic to transmit. We focus on issues such as the tradeoffs between sequential and parallel transmissions, the opportunities for schedule optimization, and the effects of interference. Thus, our approach differs from other approaches such as CSMA [1], [2], CDMA [3], [4], or 802.11. Portions of this paper appear in [5].

II. NETWORK MODEL AND ASSUMPTIONS

We consider a stationary wireless network with K sources (denoted by S_1, S_2, \dots, S_K) transmitting their traffic to N destinations (denoted by D_1, D_2, \dots, D_N). Assume that $N \leq K$, and each source transmits to only one particular intended destination (but will cause interference at the other destinations). Logically, we can partition the sources into N groups

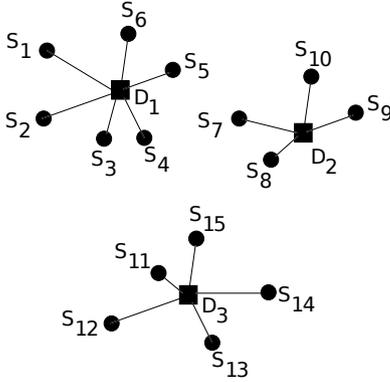


Fig. 2. The 15 sources are partitioned into 3 groups: $G_1 = \{S_1, \dots, S_6\}$, $G_2 = \{S_7, \dots, S_{10}\}$, and $G_3 = \{S_{11}, \dots, S_{15}\}$.

G_1, G_2, \dots, G_N , where G_i is the set of sources that transmit to destination D_i . For example, let us revisit the 15 sources in Fig. 1, and further assume that (i) S_1, S_2, \dots, S_6 are intended for D_1 , (ii) S_7, S_8, S_9, S_{10} are intended for D_2 , and (iii) $S_{11}, S_{12}, \dots, S_{15}$ are intended for D_3 . The sources can then be partitioned into the 3 groups as shown in Fig. 2.

We assume the following:

- The nodes, whose locations are known and fixed, are equipped with omnidirectional antennas.
- Each destination can receive at most one successful transmission at a time. However, it is possible to extend this analysis to nodes with multiple reception capability.
- Each source can communicate directly with its destination, i.e., it does not rely on other nodes to relay its traffic. However, our model can be extended to include multi-hop communication by letting some nodes be both sources and destinations, i.e., such nodes also act as relay nodes.
- Each source always has traffic to transmit, i.e., its transmission queue is never empty.
- Time is divided into slots, whose length equals that of one fixed-length packet. A frame consists of M_{frame} consecutive time slots.
- Our primary performance measure is throughput, which is the average number of packets that are successfully received per intended destination per time slot (packets received at a “wrong” destination do not contribute to throughput). We do not address issues such as time delays and stability analysis in this paper.
- The propagation delay among the nodes in the network is negligible.
- Nodes transmit according to a schedule, i.e., a node can transmit only in an assigned time slot. We require that each source transmits at least once in each frame, and that the schedule repeats from frame to frame. Thus, it is sufficient to study the performance in any one frame.

Definition 1 A *schedule* is a tuple $(H_1, H_2, \dots, H_{M_{\text{frame}}})$, where H_k is the set of sources that simultaneously transmit in time slot k . \square

Thus, a schedule is completely determined when the frame length M_{frame} and the sets H_k are determined, $k = 1, 2, \dots, M_{\text{frame}}$. Clearly, the number of all possible schedules

is very large for a general network. Later in the paper, we impose some structure on the schedules to make the problem more tractable.

Suppose that we are given a set H of sources that simultaneously transmit in the same time slot. Let $S \in H$ and $P_{\text{rx}}(S, D)$ be the signal power received from node S by node D . Let $\text{SINR}(S, D)$ be the signal-to-interference-plus-noise ratio (SINR) at node D for the transmission from node S , i.e.,

$$\text{SINR}(S, D) = \frac{P_{\text{rx}}(S, D)}{P_{\text{noise}}(D) + \sum_{U \in H \setminus \{S\}} P_{\text{rx}}(U, D)}$$

where $P_{\text{noise}}(D)$ denotes the receiver noise power at node D . We assume that a packet is successfully received, even in the presence of interference and noise, as long as its SINR exceeds a given threshold [6], [7], i.e., a packet transmitted by S is successfully received by D if it is intended for D and

$$\text{SINR}(S, D) > \beta_D \quad (1)$$

where $\beta_D \geq 0$ is a threshold at node D , which is determined by application requirements and the properties of the network [8]. When $\beta_D < 1$ (e.g., in spread-spectrum networks), it is possible for more than one transmission to satisfy (1) simultaneously [9].

The wireless channel is affected by fading, as described below. First, let $P_{\text{tx}}(S)$ be the transmit power at node S , and $r(S, D)$ be the distance between nodes S and D . When node S transmits, the power received by node D is modeled by

$$P_{\text{rx}}(S, D) = A(S, D)g(S, D)$$

where $A(S, D)$ is a random variable that incorporates the channel fading, and $g(S, D)$ is the received signal power in the absence of fading. We refer to $g(S, D)$, which depends on $r(S, D)$ and $P_{\text{tx}}(S)$, as the “received power factor.” For far-field communication (i.e., when $r(S, D) \gg 1$), we have

$$g(S, D) = P_{\text{tx}}(S)r(S, D)^{-a} \quad (2)$$

where a is the path-loss exponent whose typical values are between 2 and 6. A simple approximate model for both near-field (i.e., when $r(S, D) < 1$) and far-field communication is

$$g(S, D) = P_{\text{tx}}(S)[r(S, D) + 1]^{-a} \quad (3)$$

where the term $r(S, D) + 1$ is used to ensure that $g(S, D) \leq P_{\text{tx}}(S)$, i.e., the received power is not greater than the transmitted power when there is no fading. Under Rayleigh fading, it is well known that $A(S, D)$ is exponentially distributed [10].

Our goal is to study methods for scheduling the transmissions between the sources and destinations, and to evaluate the resulting performance. Under the well-known (sequential) TDMA method, there is exactly one transmission in each time slot, i.e., the frame length equals K , the number of sources. Thus, no other-user interference is present. In this paper we consider a “parallel” approach, as described in the following, under which the local groups simply operate their protocols (e.g., TDMA) in parallel.

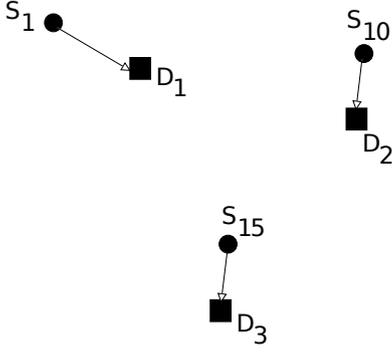


Fig. 3. Under the parallel method, sources S_1 , S_{10} and S_{15} transmit in the same time slot.

Recall that the sources are partitioned into N groups G_1, G_2, \dots, G_N , where G_i is the set of sources that are intended for destination D_i . We assume that the nodes in each group operate according to a “local” protocol (e.g., TDMA) that involves only members of that group. Thus, there are N transmissions in each slot, each of which is intended for a different destination. One approach to schedule construction is for each group to randomly generate its own schedule, independently of all of the other groups, i.e., there is no coordination among the different groups. However, as seen later, when these groups coordinate among themselves to obey an optimal schedule, the throughput performance can be significantly improved.

Let us revisit Fig. 2, which shows the network with $N = 3$ groups. According to our rule, there are 3 transmissions in each time slot. An example is shown in Fig. 3, which shows 3 transmissions ($S_1 \rightarrow D_1$, $S_{10} \rightarrow D_2$, and $S_{15} \rightarrow D_3$) in some time slot. Note that each transmission will cause interference at all of the unintended destinations.

III. THROUGHPUT EVALUATION

Consider a transmission schedule $(H_1, H_2, \dots, H_{M_{\text{frame}}})$, where H_k is the set of sources that transmit in time slot k (see Definition 1). For a given time slot k , let $C_{H_k}(S, D)$ be the probability that a packet from source S is successfully received by destination D , given that all the nodes in H_k simultaneously transmit in this time slot. Let $C_{\text{success}}(k)$ be the average total number of successful transmissions in time slot k . We then have

$$C_{\text{success}}(k) = \sum_{S \in H_k} C_{H_k}(S, D^S) \quad (4)$$

where D^S denotes the destination of S .

We now define *throughput* T to be the average number of packets that are successfully received per intended destination per time slot. Because each destination can receive at most one packet in a time slot, T is also the probability that a packet is successfully received by its intended destination in a time slot.

The throughput of the parallel method can be computed as follows. Recall that there are M_{frame} time slots in a frame, and there are N parallel transmissions in each time slot, i.e.,

$|H_k| = N$. Thus, the total number of transmissions in each frame is NM_{frame} . The throughput is then

$$T = \frac{1}{NM_{\text{frame}}} \sum_{k=1}^{M_{\text{frame}}} C_{\text{success}}(k).$$

Substituting (4) into the above expression yields

$$T = \frac{1}{NM_{\text{frame}}} \sum_{k=1}^{M_{\text{frame}}} \sum_{S \in H_k} C_{H_k}(S, D^S). \quad (5)$$

For the case of Rayleigh fading, the following result (whose proof is given in [11], [5]) provides the exact formula for $C_{H_k}(S, D^S)$, which depends on the receiver noise, channel fading, receiver threshold, and other-user interference.

Theorem 1 Suppose that the fading between a transmitting node S and a receiving node D is modeled as a Rayleigh random variable Y_S with parameter $v(S, D)$. For $S \neq U$, assume that Y_S and Y_U are independent. Let $g(S, D)$ denote the received power factor, which depends on the distance and the transmit power, e.g., $g(S, D) = P_{\text{tx}}(S)[r(S, D) + 1]^{-\alpha}$. Given that all the nodes in H_k simultaneously transmit in time slot k , the probability that a packet from S is successfully received by D is

$$C_{H_k}(S, D) = \frac{\exp\left(-\frac{\beta P_{\text{noise}}(D)}{v(S, D)g(S, D)}\right)}{\prod_{U \in H_k \setminus \{S\}} \left[1 + \beta \frac{v(U, D)g(U, D)}{v(S, D)g(S, D)}\right]}$$

where β and $P_{\text{noise}}(D)$ are the required SINR threshold and the receiver noise power at D , respectively.

Remark 1 We have $C_{H_k}(S, D^S) \leq 1$ for all schedules $(H_1, H_2, \dots, H_{M_{\text{frame}}})$. It can then be shown that the throughput in (5) for the parallel method is bounded by 1, i.e., $T \leq 1$. For any given schedule, this upper bound is achieved when $\beta = 0$. \square

Remark 2 For a given schedule, we can analytically compute the throughput T in (5). The computation of T requires a double sum that adds the $M_{\text{frame}}N$ terms of the form $C_{H_k}(S, D^S)$, where M_{frame} is the frame length and N is the number of destinations. The computation of $C_{H_k}(S, D^S)$ in turn requires a product of N terms (by Theorem 1). The overall computational complexity for computing T is then $O(M_{\text{frame}}N^2)$, which, for a given value of N , is minimized when M_{frame} is minimized. Thus, it is desirable to minimize M_{frame} . \square

Remark 3 Recall that, under the sequential TDMA method, there is exactly one transmission in each time slot, i.e., there is no other-user interference and $|H_k| = 1$ for all time slot k . The throughput T for the parallel method is given in (5). Similarly,

it can be shown that the throughput for the sequential TDMA method is

$$T_{\text{TDMA}} = \frac{1}{KN} \sum_{i=1}^K C_{\{S_i\}}(S_i, D^{S_i})$$

where D^{S_i} denotes the destination of source S_i , K is the number of sources, and N is the number of destinations. We must have $T_{\text{TDMA}} \leq 1/N$, and $T_{\text{TDMA}} = 1/N$ under the ideal condition $P_{\text{noise}}(D^{S_i}) = 0$ for all i . \square

IV. TRANSMISSION SCHEDULING FOR THE PARALLEL METHOD

Recall from Definition 1 that a *schedule* is a tuple

$$(H_1, H_2, \dots, H_{M_{\text{frame}}})$$

where M_{frame} is the frame size (i.e., the number of time slots in the frame) and H_k is the set of sources that simultaneously transmit in time slot k . In this section, we provide the structure for the schedules and their enumerations.

A. Schedule Specifications

Recall that there are N destinations, and the K sources are partitioned into N groups. Group G_i consists of the sources that are intended for destination D_i . These N groups transmit simultaneously in each time slot, i.e., there are N simultaneous transmissions, each of which is intended for a different destination and $|H_k| = N$ for each time slot k . To ensure fairness among the sources that belong to the same group, we require that they have the same number of transmissions in each frame. For group G_i , this number is denoted by h_i . However, different groups may have different number of transmissions, i.e., we may have $h_i \neq h_j$ for some $i \neq j$. Let m_i be the number of sources in group i , i.e., $m_i = |G_i|$. We must have

$$h_i = \frac{M_{\text{frame}}}{m_i}. \quad (6)$$

From Remark 2, to simplify the computation of the throughput T in (5), we now require that the frame length M_{frame} be the minimum value that will permit each source in the same group to transmit the same number of times. From (6), M_{frame} must be a common multiple of m_1, m_2, \dots, m_N . Thus, M_{frame} is minimized only if it is the *least* common multiple (LCM) of m_1, m_2, \dots, m_N , i.e.,

$$M_{\text{frame}} = \text{LCM}(m_1, m_2, \dots, m_N). \quad (7)$$

Example 1 Consider a wireless network with $K = 7$ sources and $N = 3$ destinations. Assume that (i) S_1 and S_2 are intended for D_1 , (ii) S_3, S_4 , and S_5 are intended for D_2 , and (iii) S_6 and S_7 are intended for D_3 . Thus, the sources are partitioned into $G_1 = \{S_1, S_2\}$, $G_2 = \{S_3, S_4, S_5\}$, and $G_3 = \{S_6, S_7\}$. The cardinalities of the groups are $m_1 = 2$, $m_2 = 3$, $m_3 = 2$, and hence $M_{\text{frame}} = \text{LCM}(2, 3, 2) = 6$. From Definition 1, each schedule is specified by the tuple (H_1, H_2, \dots, H_6) . Two example schedules are shown in Fig. 4. For Schedule 1, we have $H_1 = \{S_1, S_3, S_6\}$, $H_2 =$

Time slot	1	2	3	4	5	6
Schedule 1	S_1	S_2	S_1	S_2	S_1	S_2
	S_3	S_4	S_5	S_3	S_4	S_5
	S_6	S_7	S_6	S_7	S_6	S_7
Schedule 2	S_1	S_1	S_1	S_2	S_2	S_2
	S_3	S_3	S_4	S_4	S_5	S_5
	S_6	S_6	S_7	S_7	S_6	S_7

Fig. 4. Two example schedules.

$\{S_2, S_4, S_7\}, \dots, H_6 = \{S_2, S_5, S_7\}$. For Schedule 2, we have $H_1 = \{S_1, S_3, S_6\}$, $H_2 = \{S_1, S_3, S_6\}, \dots, H_6 = \{S_2, S_5, S_7\}$. In each frame of 6 slots, each member of G_1 transmits 3 times, each member of G_2 transmits 2 times, and each member of G_3 transmits 3 times, i.e., $h_1 = 3$, $h_2 = 2$, and $h_3 = 3$. \square

B. Schedule Enumerations

Two schedules $(H_1, H_2, \dots, H_{M_{\text{frame}}})$ and $(H'_1, H'_2, \dots, H'_{M_{\text{frame}}})$ are identical when $H_k = H'_k$ for $k = 1, 2, \dots, M_{\text{frame}}$. Otherwise, they are said to be *different*. Two schedules are said to be *equivalent* when one schedule is a permutation of another.

Theorem 2 Let f and e be the number of different schedules and the number of non-equivalent schedules, respectively. We have

$$f = \prod_{i=1}^N \frac{M_{\text{frame}}!}{(h_i!)^{m_i}}$$

and

$$\frac{f}{M_{\text{frame}}!} \leq e \leq \frac{f}{M_{\text{frame}}!} (h_i!)^{m_i}$$

for all i , $1 \leq i \leq N$.

Proof Let f_i be the number of different schedules generated by group i . By assumption, each source in group i transmits h_i times in each frame. Because there are M_{frame} transmissions in each frame, and there are m_i sources in group i , we have

$$f_i = \frac{M_{\text{frame}}!}{h_i! \dots h_i!} = \frac{M_{\text{frame}}!}{(h_i!)^{m_i}}.$$

Thus, the total number of different schedules generated by the N groups $1, 2, \dots, N$, is $f = f_1 f_2 \dots f_N$. The set of f different schedules can be partitioned into the e sets of equivalent schedules: E_1, E_2, \dots, E_e . Because $|E_j| \leq M_{\text{frame}}!$, we have $e \geq \frac{f}{M_{\text{frame}}!}$.

Let A be a schedule generated by group i , and X_A be the set of different schedules generated by the N groups under the restriction that the schedule generated by group i is A . Note that $|X_A| = \prod_{j=1, j \neq i}^N f_j = f/f_i$. Let B be a schedule generated by group i . By permuting the order of the transmissions of X_B , it becomes identical to X_A , i.e., the schedules in X_A and X_B are equivalent. Thus, we have

$$e \leq |X_A| = \frac{f}{f_i} = \frac{f}{\frac{M_{\text{frame}}!}{(h_i!)^{m_i}}} = \frac{f}{M_{\text{frame}}!} (h_i!)^{m_i}. \quad \square$$

To summarize, we can compute the throughput T in (5) for each (parallel) schedule. The number of schedules is given by Theorem 2. As discussed later in Remark 5, possible methods for generating schedules include (i) distributed operation (in which each group chooses its schedule randomly, and independently of the other groups) and (ii) exhaustive search (which determines the optimal, i.e., maximum throughput) schedule.

Remark 4

(1) The upper bound for e in Theorem 2 is tighter for smaller h_i . Suppose that $h_i = 1$ for some i . Theorem 2 then implies that

$$e = \frac{f}{M_{\text{frame}}!}. \quad (8)$$

For example, let $m_1 = 4$ and $m_2 = 2$. From (6) and (7), we have $h_1 = 1, h_2 = 2$, and $M_{\text{frame}} = 4$. Using Theorem 2 and (8), we have $f = 144$ and $e = f/M_{\text{frame}}! = 144/4! = 6$.

(2) Consider the special case where all the groups have the same size. We then have $m_i = |G_i| = K/N$, $M_{\text{frame}} = K/N$, and $h_i = 1$ for all i . Using Theorem 2 and (8), the number of non-equivalent schedules for this special case is

$$e = (M_{\text{frame}}!)^{N-1} = \left(\frac{K}{N}\right)^{N-1}. \quad \square$$

V. PERFORMANCE EVALUATION

In this section, we evaluate and compare the throughput performance, by numerical examples, for the sequential method and the parallel method (with random schedules and optimal schedules). We show the impact of SINR threshold, receiver noise level, other-user interference, network topology, and schedules on performance. We also show that the performance of the random schedules is almost as good as that of the optimal schedules under certain conditions. We assume the following:

- The path-loss exponent is $a = 3$.
- The SINR threshold and the receiver noise power are the same at each destination D , i.e., we can now write $\beta_D = \beta$ and $P_{\text{noise}}(D) = P_{\text{noise}}$.
- The wireless channel is affected by Rayleigh fading with Rayleigh parameter $v(S, D) = 1$.
- The received power factor is given by (3), i.e., $g(S, D) = P_{\text{tx}}(S)[r(S, D) + 1]^{-a}$.
- The transmit power is $P_{\text{tx}}(S) = 1$ for all sources S , i.e., there is no power control.

We now study a stationary wireless network that has $K = 24$ sources and $N = 6$ destinations. The 6 destinations are located on a 10×20 rectangle, as shown in Fig. 5, i.e., 4 destinations at the 4 corners and the other 2 destinations at the mid-points of the longer sides of the rectangle. Fig. 5 implies that the neighboring destinations are separated by the distance of either 10 or $10\sqrt{2}$. Assume that sources $S_{4i-3}, S_{4i-2}, S_{4i-1}$, and S_{4i} are intended for destination D_i , and they are randomly located in the circle centered at D_i and of radius R , $1 \leq i \leq 6$. For example, S_1, S_2, S_3 , and S_4 are intended for D_1 , and they are randomly located in the circle centered at D_1 and of radius

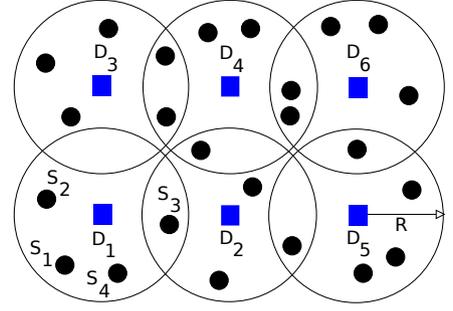


Fig. 5. A wireless network with 24 sources S_i and 6 destinations D_j .

R . Note that R determines the distribution of the location of the sources that are associated with each particular destination. However, a source's transmission will cause interference at the other destinations outside these circles. The level of other-user interference increases when R increases.

We assume that the network operates according to a TDMA-based protocol, i.e., we are interested in the performance of the sequential TDMA and parallel TDMA. Note, however, that our model can also be extended to other types of protocols.

The throughput T for the parallel method for a given schedule is computed by (5). To specify the transmission policy under the parallel method, we need to determine the groups and the schedules (see Section IV-A). There are $N = 6$ groups: $G_1 = \{S_1, S_2, S_3, S_4\}$, $G_2 = \{S_5, S_6, S_7, S_8\}$, ..., $G_6 = \{S_{21}, S_{22}, S_{23}, S_{24}\}$. In this example, all the 6 groups have the same size, i.e., $m_1 = m_2 = \dots = m_6 = 4$. From Remark 4, the frame length is $M_{\text{frame}} = 4$, and the number of non-equivalent schedules is $e = (4!)^6 = 7,962,624$. As noted earlier, under the parallel method we consider 2 approaches to scheduling: exhaustive search (which produces the optimal throughput) and random scheduling (see Remark 5). The throughput values obtained using these schemes, respectively, are:

- The maximum throughput, T_{max} , which is produced by the optimal schedules.
- The average throughput, T_{ave} , which is obtained by averaging the throughput values produced by the e schedules. We have $T_{\text{ave}} \leq T_{\text{max}}$.

Under the parallel method, in order for the nodes in the network to obey a particular schedule, they must coordinate among themselves to meet the specifications of the schedule. Clearly, it is desirable for the network to operate according to one of the best schedules (that yield the maximum throughput T_{max}). A simple form of distributed implementation is to use randomly chosen schedules, in which there is no coordination among the groups. The long-term throughput performance under such random schedules approaches T_{ave} (see Remark 5). As seen in the following, even these random schedules can significantly outperform the sequential TDMA method in many cases. Recall that the sources that are intended for the same destination are located in the circle of radius R centered at that destination. The results are shown in Figs. 6, 7, and 8 for $R = 5$, and in Figs. 9, 10, and 11 for $R = 10$. For each case, we show the throughput performance versus the SINR

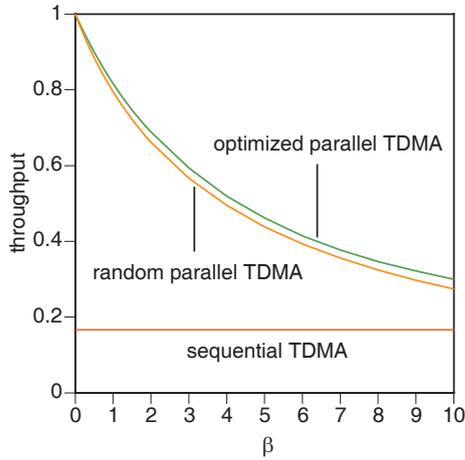


Fig. 6. Throughput vs SINR threshold: $R = 5, P_{\text{noise}} = 0$.

threshold β for $P_{\text{noise}} \in \{0, 10^{-4}, 10^{-3}\}$. Note that β is the required SINR threshold, and higher values of β correspond to higher quality of service requirements (e.g., lower bit error rates).

The general observation is that smaller values of β , R , and P_{noise} yield higher throughput, as expected. The sequential TDMA throughput T_{TDMA} is given in Remark 3. We have $T_{\text{TDMA}} \leq 1/N = 1/6$. As seen in Fig. 6, $T_{\text{TDMA}} = 1/6$ for the ideal case of $P_{\text{noise}} = 0$, independent of β and R . For small values of β (e.g., $\beta < 1$), the performance of random parallel schedules is not very different from that of the optimal parallel schedules, and these parallel schedules significantly outperform the sequential TDMA schedules.

The main advantage of the sequential method is that there is no other-user interference. The main advantage of the parallel method is the possibility of simultaneous reception of multiple packets (at different destinations) in each time slot, although other-user interference does reduce the probability of successful reception. In both methods, receiver noise and fading play a role. As seen in the following, when the groups are sufficiently separated, or when the receiver noise is sufficiently high, the parallel method typically outperforms the sequential method, because the *relative* impact of other-user interference is reduced (see also Remark 6).

First, consider the case of $R = 5$, for which throughput results are shown in Figs. 6, 7 and 8, where T_{max} and T_{ave} are plotted for the optimal and random parallel schedules, respectively. Recall that the neighboring destinations are separated by the distance of either 10 or $10\sqrt{2}$. Although the circles of radius $R = 5$ are just touching (rather than overlapping), there still exists other-user interference in this case. The parallel TDMA method (for both the optimal and random schedules) significantly outperform the sequential TDMA method for all shown values of β . This can be explained by noting that the sources in the groups are not very close to destinations other than their own in this case. This separation reduces the other-user interference at each destination, and increases the chance that multiple packets (one packet for each intended destination) are received successfully under the parallel TDMA. Note also that the average performance of the

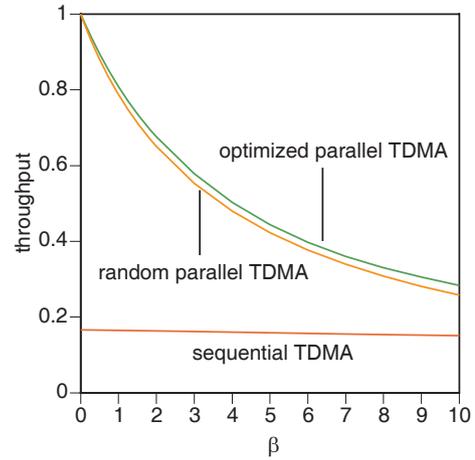


Fig. 7. Throughput vs SINR threshold: $R = 5, P_{\text{noise}} = 10^{-4}$.

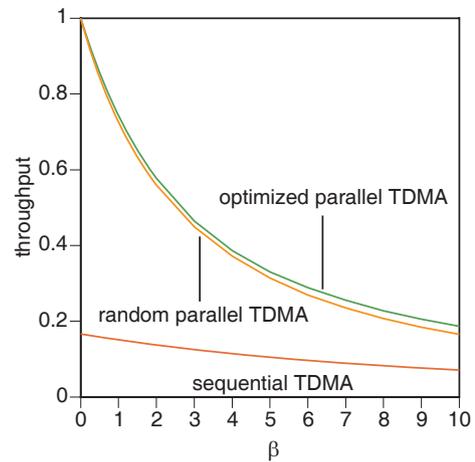
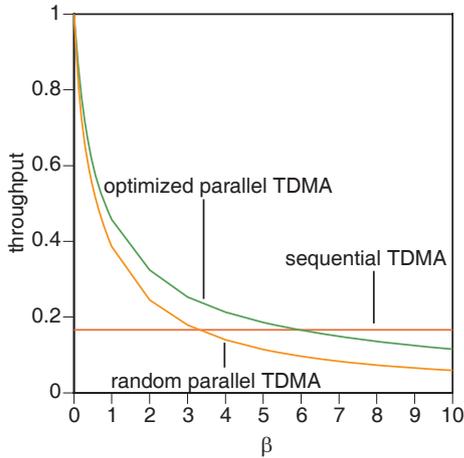
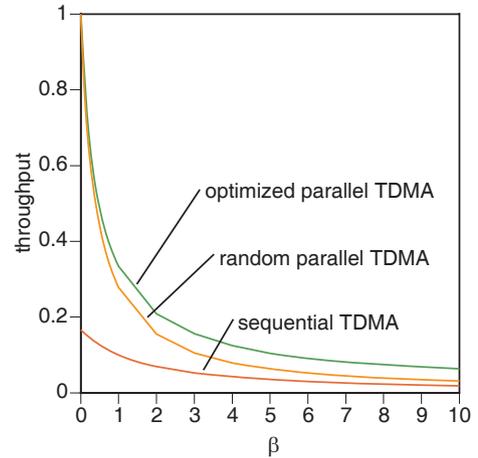
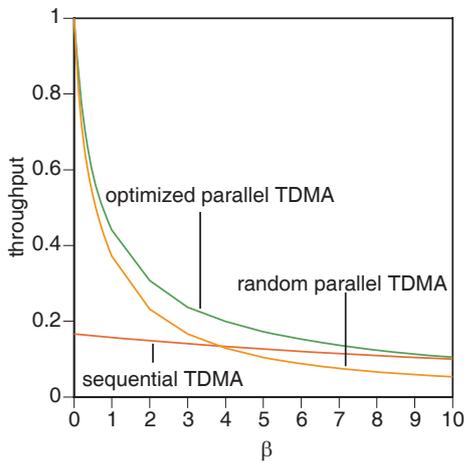


Fig. 8. Throughput vs SINR threshold: $R = 5, P_{\text{noise}} = 10^{-3}$.

random parallel schedules is not far from that of the optimal parallel schedules. This result is consistent with the intuitive observation that when the groups are reasonably far apart, performance does not depend strongly on the schedules of neighboring groups. In other words, there is little interaction among the groups, and hence little opportunity to improve performance by coordinating transmission schedules. Thus, schedule optimization yields only modest improvement, and random schedules perform sufficiently well for the case of $R = 5$. That is, the random schedules (a simple form of distributed implementation for the parallel method) perform well when the groups are sufficiently separated, even when the nodes move at very high speeds (see also Remark 5.2).

Next, consider the case of $R = 10$ (Figs. 9, 10, and 11), where the circles of radius R now overlap significantly. Thus, the impact of interference from members of other groups is also significant. For the parallel method, the performance can be noticeably improved through the appropriate choice of transmission schedules, i.e., schedule optimization noticeably improves the performance in this case. For the ideal case of $P_{\text{noise}} = 0$ (Fig. 9), the sequential TDMA outperforms the random parallel method for $\beta > 3.3$ and the optimal parallel method for $\beta > 6$. Thus, the sequential TDMA is


 Fig. 9. Throughput vs SINR threshold: $R = 10$, $P_{\text{noise}} = 0$.

 Fig. 11. Throughput vs SINR threshold: $R = 10$, $P_{\text{noise}} = 10^{-3}$.

 Fig. 10. Throughput vs SINR threshold: $R = 10$, $P_{\text{noise}} = 10^{-4}$.

more appropriate under the condition of very low receiver noise level and high interaction among the groups (i.e., when the network is interference-limited). When $P_{\text{noise}} = 10^{-4}$ (Fig. 10), the random parallel method outperforms the sequential TDMA for $\beta < 4$, and the optimal parallel method outperforms the sequential TDMA for all shown values of β . Optimal schedules also noticeably outperform the random schedules, i.e., schedule optimization is recommended for this case. When $P_{\text{noise}} = 10^{-3}$ (Fig. 11), the receiver noise level is relatively high and has a noticeable impact on throughput performance. In this case, the parallel method (random and optimal) outperforms the sequential method.

To summarize, the above results show that the sequential schedules outperform the parallel schedules when the receiver noise level is sufficiently low, the impact of other-user interference is high (i.e., R is sufficiently large), and the required SINR β is sufficiently high. Otherwise, the parallel schedules perform much better. The throughput is further improved by schedule optimization under the parallel method. Additionally, when the value of required SINR threshold β or the interaction among the groups is sufficiently low, the performance of the random schedules is not far from that of the optimal schedules. This is also true when the receiver noise power

P_{noise} is sufficiently high. In these cases, the random parallel method, which is a simple form of distributed implementation, performs only slightly below the optimal parallel method, even for networks with extreme mobility (see also Remark 5.2).

Remark 5

(1) A drawback of the random parallel method is that some schedules may provide throughput that is considerably lower than T_{ave} . However, this drawback can be avoided by using a different random schedule in each frame to yield long-term performance that approaches T_{ave} . Further, as long as mobility is not excessive, once a good schedule is determined (by appropriate exchange of information among the groups), it can be used in subsequent frames.

(2) The random schedules can be generated in a distributed manner as follows. Consider an arbitrary group G_i . The sources in this group transmit in time slots that are organized into frames, i.e., each frame consists of M_{frame} consecutive time slots. Note that the sources transmit according to a known order in each frame. By randomly permuting the transmission order from one frame to the next, random schedules can be generated. More specifically, the sources start in Frame 1 (which consists of the first M_{frame} consecutive time slots) whose transmission order is assumed to be known by each group member. Each source then uses a common computer random number generator (RNG) with the same initial seed to construct Frame 2 that is a random permutation of Frame 1. Thus, as long as the same RNG is used by all group members, the transmission order in Frame 2 is known by each group member. Using the same procedure, subsequent random frames can be constructed. Other groups can use a similar method for constructing their own random frames. To ensure the frames generated by different groups are uncorrelated, different initial seeds or different RNGs are used by different groups. Note that random schedules are constructed without requiring channel and topological information, i.e., they can be constructed even when nodes move at extreme speeds.

(3) To implement random parallel TDMA, each source in a group forms the next frame by randomly permuting the current frame. Randomly permuting a frame of length M_{frame}

time slots requires $O(M_{\text{frame}})$ computational steps [12]. Thus, each source requires $O(1)$ computational steps per time slot to generate a random schedule. Using exhaustive search, the optimal schedule is found by computing and comparing the throughput values for all schedules. From Remark 2, computing the throughput for each schedule has complexity $O(M_{\text{frame}}N^2)$, where N is the number of destinations. Thus, the overall computational complexity of the exhaustive search is $O(fM_{\text{frame}}N^2)$, where f is the number of schedules (see Theorem 2). \square

Remark 6 As discussed in the following, under certain conditions, the throughput performance under the parallel method is nearly independent of how the schedules are chosen. Thus, under these conditions, an arbitrary schedule (e.g., a randomly chosen schedule) then performs close to the optimal schedule.

(1) Assume that the SINR threshold $\beta \approx 0$. It then follows from Remark 1 that any parallel schedule (optimal or random) will produce throughput $T \approx 1$ (which can also be seen in Figs. 6 - 11).

(2) Assume that the groups operate in an ideal environment that has no interference among the groups (e.g., when the groups are sufficiently far from each other, or when other-user interference is negligible to receiver noise). Consider an arbitrary group G_i and an arbitrary schedule. Our assumption implies that the local throughput of G_i , denoted by T_i , is then independent of the other groups, as well as independent of the transmission order in each frame. Note that T_i does depend on the receiver noise and channel fading (but not on other-user interference). Thus, the total (normalized) network throughput for the N groups is $T = \sum_{i=1}^N T_i/N$, which is independent of how the schedule is chosen (optimal or random). \square

VI. SUMMARY

The focus of this paper is on the transmission-scheduling problem for multiple-destination wireless networks that operate under realistic conditions that include noise, fading, attenuation, and other-user interference. For a given set of sources, each with packets intended for a specific single destination, as well as knowledge of the network topology and the channel statistics, the goal is to find the transmission schedule that maximizes throughput. Our parallel TDMA scheduling (under which a TDMA schedule is established for each destination) provides higher throughput than sequential TDMA over a wide range of system parameters (e.g., low SINR threshold and sufficient separation among the groups). In addition, the use of randomly generated schedules (which are generated independently for each destination in a distributed manner) can provide throughput that is almost as high as that of the optimal parallel schedules (which are determined by exhaustive search) for such system parameters.

REFERENCES

[1] L. Kleinrock and F. A. Tobagi, "Packet switching in radio channels—part I: CSMA models and their throughput-delay characteristics," *IEEE Trans. Commun.*, vol. 23, no. 12, pp. 1400–1416, Dec. 1975.

- [2] F. A. Tobagi and L. Kleinrock, "Packet switching in radio channels—part II: the hidden terminal problem in CSMA and busy-tone solution," *IEEE Trans. Commun.*, vol. 23, no. 12, pp. 1417–1433, Dec. 1975.
- [3] M. B. Pursley, "The role of spread spectrum in packet radio networks," *Proc. IEEE*, vol. 75, no. 1, pp. 116–134, Jan. 1987.
- [4] S. Verdú, *Multuser Detection*. Cambridge University Press, 1998.
- [5] G. D. Nguyen, S. Kompella, J. E. Wieselthier, and A. Ephremides, "Optimization of transmission schedules in capture-based wireless networks," in *Proc. IEEE MILCOM*, Nov. 2008.
- [6] C. T. Lau and C. Leung, "Capture models for mobile packet radio networks," *IEEE Trans. Commun.*, vol. 40, no. 5, pp. 917–925, May 1992.
- [7] C. Namislo, "Analysis of mobile radio slotted ALOHA networks," *IEEE J. Sel. Areas Commun.*, vol. 2, no. 4, pp. 583–588, July 1984.
- [8] J. E. Wieselthier, G. D. Nguyen, and A. Ephremides, "Throughput (bits/sec/Hz) of capture-based random-access systems with SINR channel models," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, pp. 2401–2405, June 2007.
- [9] G. D. Nguyen, A. Ephremides, and J. E. Wieselthier, "Comments on 'Capture and retransmission control in mobile radio'," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 12, pp. 2340–2341, Dec. 2006.
- [10] J. Proakis, *Digital Communications*, 4th edition. McGraw-Hill, 2000.
- [11] S. Kandukuri and S. Boyd, "Optimal power control in interference-limited fading wireless channels with outage-probability specifications," *IEEE Trans. Wireless Commun.*, vol. 1, no. 1, pp. 46–55, Jan. 2002.
- [12] R. Durstenfeld, "Algorithm 235: Random Permutation," *Communications ACM*, vol. 7, no. 7, p. 420, July 1964.



Gam D. Nguyen received the Ph.D. degree in electrical engineering from the University of Maryland, College Park, in 1990. He is currently at the Naval Research Laboratory, Washington, DC. His research interests include communication systems and networks. Dr. Nguyen received the IEEE Fred W. Ellersick Award for the best unclassified paper at MILCOM 2000.



Sastry Kompella (S'04-M'07) received the BE degree in electronics and communication engineering from Andhra University, India, in 1996, the MS degree in electrical engineering from Texas Tech University in 1998, and the Ph.D. degree in electrical and computer engineering from the Virginia Polytechnic Institute and State University, Virginia, in 2006. Currently, he is the section head for the Wireless Network Research Section under the Information Technology Division at the Naval Research Laboratory, Washington, DC. His research

interests include cross-layer design, optimization, and scheduling in wireless as well as cognitive radio networks.



Jeffrey E. Wieselthier (S'67-M'69-SM'88-F'07) received the Ph.D. degree in electrical engineering from the University of Maryland, College Park, in 1979. He was employed at the Naval Surface Warfare Center, White Oak, Silver Spring, MD, from 1969 to 1979. From 1979 to 2007, he was with the Information Technology Division of the Naval Research Laboratory, Washington, DC, where he was a Senior Researcher and Head of the Wireless Network Theory Section of the Networks and Communication Systems Branch. Additionally, he was a program manager in communications and networking for the Office of Naval Research. He is currently a self-employed consultant with Wieselthier Research. Dr. Wieselthier was Lead Guest Editor of the 2005 special issue on Wireless Ad Hoc Networks in the *IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS*, and he was on the Editorial Board of Elsevier's journal *Ad Hoc Networks*. He was Technical Program Co-Chair of the Third IEEE Symposium on Computers and Communications in Athens, Greece, in 1998. He twice served as Treasurer of the IEEE International Symposium on Information Theory (1991 in Budapest, Hungary, and 2011 in Saint Petersburg, Russia). He won the IEEE Fred W. Ellersick Award for the best unclassified paper at MILCOM 2000. His current interests include wireless communication networks, with an emphasis on issues relating to energy-aware, cross-layer, and cooperative operation of ad hoc and sensor networks.



Anthony Ephremides (S'68-M'71-SM'77-F'84) received the Ph.D. degree in electrical engineering from Princeton University, Princeton, NJ, in 1971. He holds the Cynthia Kim Professorship of Information Technology at the Electrical and Computer Engineering Department of the University of Maryland in College Park, where he holds a joint appointment at the Institute for Systems Research. He served as President of the IEEE Information Theory Society in 1987 and as a member of the IEEE Board of Directors in 1989 and 1990. He

has been the General Chair and/or the Technical Program Chair of several

technical conferences. His research interests lie in the areas of communication systems and networks and all related disciplines, such as information theory, control and optimization, wireless networks, and energy efficient systems. Dr. Ephremides received the IEEE Donald E. Fink Prize Paper Award in 1991 and the first ACM Achievement Award for Contributions to Wireless Networking in 1996, the 2000 Fred W. Ellersick MILCOM Best Paper Award, the IEEE Third Millennium Medal, the 2000 Outstanding Systems Engineering Faculty Award from the Institute for Systems Research, and the Kirwan Faculty Research and Scholarship Prize from the University of Maryland in 2001. He also received the 2006 Aaron Wyner Award for Exceptional Service and Leadership to the IEEE Information Theory Society.