Ordered Packet Scheduling in Wireless Ad Hoc Networks: Mechanisms and Performance Analysis [*]

V. Kanodia[†], C. Li[‡], A. Sabharwal[†], B. Sadeghi[†], and E. Knightly[†]

[†]ECE/CS Departments
Rice University
Houston, TX 77005
{kanodia,ashu,bahar,knightly}@rice.edu

[‡]CS Department
University of Virginia
Charlottesville, VA 22904
cl4v@cs.virginia.edu

ABSTRACT

Wireless *ad hoc* networks based on the IEEE 802.11 protocol can incur severe unfairness even in simple topologies. In particular, two topological properties that we define in a graph-theoretic framework and refer to as information asymmetry and perceived collisions result in significant performance degradations and unfairness. In this paper, we present the design and analysis of Distributed Wireless Ordering Protocol (DWOP), a distributed scheduling and media access algorithm targeted towards ensuring that packets access the medium in an order defined by an ideal reference scheduler such as FIFO, Virtual Clock, or Earliest Deadline First. In this way, DWOP enables QoS differentiation as well as fairness when combined with TCP. Our key technique is piggybacking head-of-line packet priorities in IEEE 802.11 control messages so that nodes can assess the relative priority of their own queued packets. With a graph-theoretic problem formulation, we design DWOP to achieve the exact reference ordering in fully connected graphs, and to have well-characterized deviations from the reference order in more complex topologies. A simple theoretical model indicates that the scheme attains rapid convergence for newly arriving nodes, and extensive simulations indicate that nearly exact reference ordering can be achieved, even in complex asymmetric and perceived-collision topologies.

1. INTRODUCTION

In *ad hoc* networks employing a CSMA/CA media access algorithm such as IEEE 802.11 [7, 14], even a simple topology with two flows and four nodes can result in near starvation. For example, as illustrated in Figure 1(a) and Reference [3], a topology in which the sender of one flow is out of radio range of the sender of another flow results in severe throughput degradation, and hence unfairness, for one of the flows. To address such performance problems and lack of fairness in IEEE 802.11 *ad hoc* networks, previous approaches have focused on introducing new protocols targeted at providing MAC-layer fairness [10-13,15-17].

Our approach is quite different. Rather than providing node- or flow-level fairness at the MAC layer, we target an ordering mechanism that can be used to approximate a set of reference schedulers in order to achieve either QoS differentiation or fairness. That is, our objective is to design a distributed MAC protocol in which, to the closest extent possible, packets are serviced in the order defined by a reference scheduler. Our technique applies to a broad class of schedulers in which packets are serviced in increasing order of a priority index that can be computed locally.[1] This class includes Earliest Deadline First, Virtual Clock [18], and FIFO. Both Earliest Deadline First and Virtual Clock schedulers target QoS differentiation, whereas FIFO combined with TCP provides proportional-fair bandwidth allocation.

In the remainder of this paper, we consider FIFO as the reference scheduler where priority indexes are set to packets' arrival times. We present Distributed Wireless Ordering Protocol (DWOP), a media access and scheduling scheme designed to achieve reference scheduling service order in wireless *ad hoc* networks via information sharing. Our contributions are as follows.

First, we study the performance of IEEE 802.11 from the perspective of *information sharing*. Specifically, we present several scenarios in which IEEE 802.11 diverges significantly from the reference service schedule (e.g. FIFO), resulting in severe performance degradations for a subset of flows. We describe how the root of these problems is incomplete information sharing which we classify into scenarios of *asymmetric information* and *perceived collisions*.

We next introduce a graph-theoretic formalism to explore the role of information sharing in *ad hoc* networks by generalizing the development of [13]. With a general flow-contention graph, scenarios of asymmetric information and perceived collisions are readily identified so that they can be incorporated into protocol design and analysis.

Within this context, we describe DWOP. The protocol exploits the broadcast nature of the wireless medium to piggyback the priority indexes (arrival times) of queued head-

---

[*]This research is supported by NSF Grants ANI-9730104 and ANI-0085842, a Texas Technology Development and Transfer Grant, a Sloan Fellowship, and a gift from Texas Instruments.

MOBIHOC'02, June 9-11, 2002, EPFL Lausanne, Switzerland.

[1] That is, the index must be computable using only flow and node state and not the state of remote flows.

of-line packets on existing hand-shake messages. As the targeted (global) FIFO schedule would transmit packets in order of these arrival times, each node builds a scheduling table based on overheard information of other packet's arrival times. With this information sharing, we devise a simple MAC rule such that a node contends for the medium only if it's locally queued packet has a smaller arrival time than all packets in its table, i.e., if the node has inferred that it possesses the next region-wide packet in the hypothetical reference FIFO schedule. Otherwise, if there are higher priority (lower arrival time) packets in the table, the node defers access. Contending nodes access the medium according to the IEEE 802.11 protocol, and deferring nodes can be viewed as setting an extended NAV (Network Allocation Vector) to wait for their turn. We show that DWOP attains a perfect FIFO schedule for networks with continuously backlogged flows and all nodes within (symmetric) radio range of each other. Moreover, we show that two additional table management techniques, receiver participation and stale entry elimination, limit DWOP's deviations from the reference FIFO schedule in more complex topologies characterized by flow graphs with asymmetric information or perceived collisions.

Realistic systems will have dynamic properties due to factors such as mobile nodes, sleeping nodes waking up, etc. With such behavior, nodes will not always have complete information about other nodes within their radio range. Hence, we develop a simple analytical model to study the transient characteristics and convergence properties of DWOP. In particular, a new node must hear from each other node within its radio range in order to have a complete scheduling table and be assured not to transmit a packet in non-FIFO order. We show that DWOP's convergence is sufficiently fast to allow (for example) high mobility speeds, and that DWOP's determinism results in significantly faster convergence than the time required to hear from each user in an analogous scenario for IEEE 802.11.

Finally, we perform a set of *ns-2* simulations to evaluate the ability of DWOP to schedule packets in order of their priority indexes. We revisit the adverse scenarios in which IEEE 802.11 performs poorly and show that near-perfect FIFO is achieved by DWOP, with deviations limited to four packets for DWOP, as compared to practically unbounded deviations for IEEE 802.11. Moreover, we show that even in simple topologies in which all nodes are within radio range of each other, DWOP dramatically improves the packet transmission order and hence fairness as compared to IEEE 802.11.

The remainder of this paper is organized as follows. In Section II, we describe the role of information sharing in the poor and unfair performance obtained by IEEE 802.11 in *ad hoc* networks. In Section III, we present the DWOP protocol in the context of a graph-theoretic view of information sharing. Next, in Section IV, we present an analytical model used to explore the transient behavior of DWOP. Finally, in Section V we review related work and in Section VI we conclude.

2. INFORMATION SHARING IN IEEE 802.11

In this section, after briefly reviewing the IEEE 802.11 Distributed Coordination Function (DCF) protocol (see [7] for further details), we study the role of information sharing in the protocol's ability to provide a FIFO-like service. In particular, we present two topologies in which IEEE 802.11 obtains severe performance degradations, in one case due to asymmetry of information, and in the latter, due to "perceived collisions". While the former problem is previously documented in the context of fair bandwidth allocation [3, 13], our perspective of information sharing, presented in Section 3, addresses both problems and provides the context for distributed FIFO scheduling.

2.1 Review of IEEE 802.11 DCF

In this paper, we consider the IEEE 802.11 four-way handshake protocol depicted in Figure 4. (Note that the boxes marked CURRENT PACKET INFO and tables at the bottom are our proposed modifications to 802.11 and will be discussed in Section 3.) A node that intends to transmit a packet waits until the channel is sensed idle for a time period equal to Distributed InterFrame Spacing (DIFS). If the channel is sensed idle for DIFS seconds, the node generates a random backoff timer chosen uniformly from the range $[0, w-1]$, where w is referred to as the contention window. At the first transmission attempt, w is set to $CW_{\min}$ (minimum contention window). The backoff timer is decremented as long as the channel is sensed idle, stopped when a transmission is detected on the channel, and reactivated when the channel is sensed idle again for more than a duration DIFS.

After the backoff timer reaches 0, the node transmits a short request to send (RTS) message. When the receiving node detects an RTS, it responds after a time period equal to the Short InterFrame Spacing (SIFS) with a clear to send (CTS) packet. The sending node is allowed to transmit its actual data packet only if the CTS packet is correctly received. The RTS and CTS packets have information regarding the destination node and the length of the data packet to be transmitted. Any other node which hears either the RTS or CTS packet can use the data packet length information to update its network allocation vector (NAV) containing the information of the period for which the channel will remain busy. Thus, any hidden node[2] can defer transmission suitably to avoid collision. Finally, a binary exponential backoff scheme is used in IEEE 802.11 DCF: after each unsuccessful transmission, the value of w is doubled, up to the maximum value $CW_{\max} = 2^m CW_{\min}$, where m is the number of unsuccessful transmission attempts.

2.2 Full Information

In topologies such that all nodes are within radio range of each other, nodes have equal probability of capturing the channel since they have the same information regarding the system's state.[3] In particular, after transmission of an acknowledgment packet, each node sets its backoff timer according to the same distribution as described above. Thus, since nodes have equal probability of capturing the channel, nodes obtain equal shares of service in the long-term. However, over short time-scales, the binary exponential backoff mechanism can result in significantly unequal service shares to backlogged flows.

In any case, we will show that even under such simple topologies, the service order of IEEE 802.11 diverges significantly from a FIFO schedule due simply to the random

---

[2] Readers are referred to [3] for more discussion on hidden terminal problem.

[3] For clarity, we ignore the effects of transmission errors and propagation delay in the discussions below.

access nature of the protocol.

2.3 Asymmetric Information

In topologies where all nodes are *not* within radio range of each other, nodes can have different probability of channel capture due to one node hearing an RTS or CTS that another node does not hear. This unequal channel access probability can result in large differences in the net throughput achieved by individual nodes even over long time scales, thereby resulting in further deviations from the FIFO schedule. We refer to such scenarios as deriving from *asymmetric information* among nodes and provide an illustrative example as follows.

Consider the topology depicted in Figure 1(a) in which the receiver of Flow A (node 2) is in direct radio range of Flow B, whereas the sender (node 1) has no knowledge of Flow B. In the scenario of Figure 1(a), Flow B obtains a significantly higher throughput share as compared to Flow A, namely 95% vs. 5%.[4] The disparity in total share can be attributed to the fact that Flow B can hear packets from the receiver of Flow A, and hence knows exactly when to contend for the channel. Thus Flow A has equal probability of capturing the channel after *each* successful packet transmission by either of the flows. On the other hand, the transmitter of Flow A does not hear any packets from Flow B, and continually attempts to gain access to the channel via repeated RTS requests. The receiver node of Flow A cannot reply as it is either deferring access to Flow B or detecting a collision between a packet from Flow B and RTS from the transmitter of Flow A. In either case, the transmitter of Flow A times-out and doubles its contention window after each failed attempt. Thus the transmitter of Flow A has to discover an available time-slot randomly. Since the DATA packet size is much larger than the control packet size and the contention window can become quite large, the probability of Flow A capturing the channel is significantly less compared to Flow B. After Flow B has finished its packet transmission, it picks a backoff timer from a smaller sized initial congestion window (for the next packet) and thus is more likely to obtain channel access again. Therefore, whenever Flow B obtains the channel it tends to keep it for an extended period of time.

The unequal bandwidth shares obtained in the topology of Figure 1(a) were also observed in References [3] and [13]. In [3], the authors propose an additional control packet termed RRTS (Retransmit RTS) as a mechanism to address this issue. Although successful in the above topology, the RRTS mechanism leads to unfair throughput allocations in other topologies (an example is given in [3]), and is not a part of the IEEE 802.11 standard.

In [13], the authors attribute the above behavior to the asymmetry in information available to each flow.[5] Flow B has exact information through the receiver of Flow A, whereas the transmitter of Flow A has no information.

2.4 Perceived Collision

While in the above example, more information helps Flow B, this does not imply that more information always increases a node's throughput share. Consider the topology shown in Figure 1(b) where Flow B has information about Flow A and Flow C, while Flow A and Flow C have no information about any other flow in the system. In this case, Flow B attains 28% of the total bandwidth share, whereas Flows A and C get 36% each. Even though Flow B has information about the other two flows in the system, it obtains a smaller share of the total throughput.

The reason that Flow B obtains a smaller bandwidth share is described as follows. Whenever either Flow A or Flow C captures the channel, Flow B sets the NAV accordingly upon hearing the CTS. Due to spatial reuse, Flows A and C can capture the channel simultaneously, thus causing Flow B to set consecutive NAVs. In this case, more information at Flow B about contending flows requires it to defer access to more flows. By extension, as the number of contending flows around Flow B increases, Flow B's share can decrease. In this scenario, Flow B gains access whenever both Flow A and C are simultaneously in backoff or control packets of Flow A and C collide at Flow B. After acquiring the channel, Flow B can retain access to the channel for multiple packet transmissions for the reasons discussed for the topology in Figure 1(b). Note that as the number of contending flows increases, the probability of simultaneous backoffs reduces, but the probability of control packet collision increases. We term this phenomenon of control packet collision as *perceived collision*, as it reduces the amount of information at Flow B about other flows.

In summary, the node with more information gains from its knowledge when it has already acquired the channel but loses when it is deferring to other flows. The phenomenon of perceived collisions assists the nodes with more information to transition from no access to an access state by temporarily removing the information.

3. INFORMATION SHARING AND THE DWOP PROTOCOL

In this section, we first present a graph-theoretic framework to describe the main mechanisms leading to unfair allocation in 802.11, asymmetric information and perceived collisions. We propose a two-step procedure to convert any topology into a flow-contention graph, using the concept of shared information among contending flows as the central idea. We then present the Distributed Wireless Ordering Protocol (DWOP) to closely approximate FIFO in wireless *ad hoc* networks, even with complex topologies involving asymmetric graphs. We demonstrate that in simple topologies that have all nodes within radio range of each other, the protocol achieves perfect FIFO when all nodes are continuously backlogged. For more complex topologies we characterize the discrepancy between the DWOP schedule and the true global FIFO schedule in several special cases.

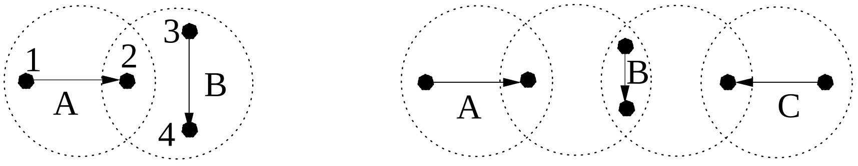
3.1 Graph-theoretic Formalism

Here, we formalize the notion of shared information via a simple graph theoretic framework. In particular, the spatially distributed nature of *ad hoc* networks naturally leads to incomplete information about the other nodes of the network. The problems arising from the spatial separation of nodes can be captured in the following framework, which follows a development similar to [13], but with an important difference to highlight asymmetric node information.

---

[4] All simulations were done using *ns*-2. Details about the simulation setup are presented in Section 5.

[5] The information is the time when a flow should contend for the channel. In IEEE 802.11, virtual carrier sense (NAV time) contains that information.

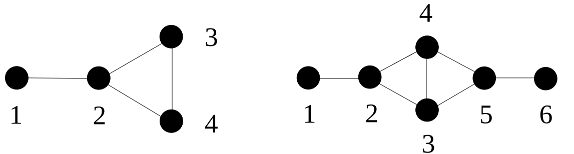


(a) Asymmetric Information (b) Perceived Collision

Figure 1: Illustrative example topologies

3.1.1 *Framework*

As the first step in the proposed framework, the geographical map of the nodes is converted into a connectivity graph based on the radio range of different nodes. In the second step, the connectivity graph is converted into a flow-contention graph with the knowledge of transmitter-receiver pair of every flow.[6] For the subsequent development, assume a network of N nodes, denoted by the set $\mathcal{N} = \{n_i : i = 1, \ldots, N\}$. Each flow constitutes a transmitter-receiver pair represented by a tuple (n_i, n_j) with $i \neq j$ and $n_i, n_j \in \mathcal{N}$. The set of K flows is represented by $\mathcal{F} = \{f_k : k = 1, \ldots, K\}$. The two steps in the procedure are formally defined as follows:



(a) Asymmetric information (b) Perceived-collision

Figure 2: **Connectivity graphs for example topologies**

1. *Connectivity graph* $G = (V, E)$: The set of vertices, V, in the connectivity graph G represent the nodes in the network, *i.e.*, $v \in V = \mathcal{N}$. An edge, $e \in E$, exists between vertices v_i and v_j if the nodes n_i and n_j are within the radio range of each other.[7]
2. *Flow-contention graph* $G' = (V', E'_s, E'_w)$: The vertices of G' represent the flows in the network, $V' = \mathcal{F}$. There is a directed *strong edge* $e'_s \in E'_s$ from vertex v'_i to v'_j if the transmitter of flow j is in the radio range of at least one of the constituent nodes (transmitter or receiver or both) of flow i. A directed *weak edge* from vertex v'_m to v'_n exists if only the receiver of flow n is in the radio range of at least one of the nodes of flow m. Note that if there is a strong edge between two flows, it immediately implies that there is a strong or weak opposite edge between the same two flows. Strong edges are denoted by solid lines and weak edges by dashed lines.

The flow-contention graphs for the asymmetric-information topology (Figure 1(a)) and perceived-collision topology (Figure 1(b)) are shown in Figures 3(a) and 3(b), respectively. The connectivity graphs of the example topologies are shown in Figures 2(a) and (b) respectively. Note that the flow-contention graph is independent of the particular scheduling or media access protocols and is rather a flow-specific representation of a connectivity graph. However, the form of the information and specific use of it is dependent on a particular choice of a protocol. The flow-contention graph can thus be used to classify the flows which are directly affected by asymmetric information and perceived collisions.

3.1.2 *Examples*

Due to the passive nature of the receiver in IEEE 802.11 (the receiver only sends CTS or ACK, but does not influence the transmitter's decision regarding when to contend), receiver information regarding timing of contending nodes is not used in medium access protocols. Thus, in 802.11, receiver information is "weaker" than the same information at the transmitter of a flow, which is why transmitter information is labeled as strong. Thus, two flows which have a directed strong edge in one direction and a weak edge in the other direction have *asymmetric information* about each other. The flow-contention graph of Figure 3(a) clearly shows the asymmetric information between Flow A and Flow B, where the shared information is the exact completion of a successful packet transmission.

The flows that have an incoming strong degree of at least two, and an outgoing weak degree of at least one are affected by *perceived collisions*. Recall that perceived collisions may not affect the flows which are transmitting the packets, but the nodes which are deferring to the active flows. Perceived collisions become more probable as the incoming strong information at a flow increases relative to the node's outgoing weak information (proportional to number of strong incoming edges), and hence can hurt the nodes with more information about other nodes.

The flow contention graph only captures the information exchange between the flows. Combined with a specific protocol, which defines the actions taken at each information exchange, the graph-theoretic framework can also be used to study the role of information sharing in other medium-access protocols.

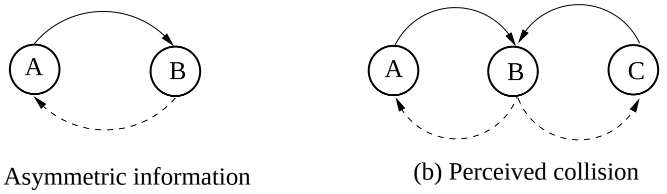
3.2 DWOP: Distributed Wireless Ordering Protocol

In this section, we present DWOP, a protocol that approximates reference scheduler in wireless *ad hoc* networks by exploiting overheard information from other nodes to estimate when to contend for the channel. We first describe how priority indexes can be communicated via piggybacking so that nodes can build local scheduling tables based on overheard information. We then describe how to exploit

---

[6] In this conversion, for simplicity we assume that the carrier sense range is the same as the packet detect range, ignoring the "double ring" effect explored in detail in [9].

[7] Two nodes are considered to be within radio range of each other if they can decode each other's packets reliably.



(a) Asymmetric information (b) Perceived collision

Figure 3: Flow-contention graphs for example topologies

the piggybacked information to obtain a reference (FIFO) schedule within the framework of IEEE 802.11 for a topology where all nodes are within radio range of each other. We then show that a receiver's scheduling table information can be effectively used in more complex topologies, via *receiver out-of-order notification* which reduces the information disparity among nodes. Finally, since the local scheduling table can potentially have stale entries due to perceived collisions, mobility and channel errors, we propose a distributed stale entry detection method enabling a quick recovery to steady state.

3.2.1 Distributing Arrival Times

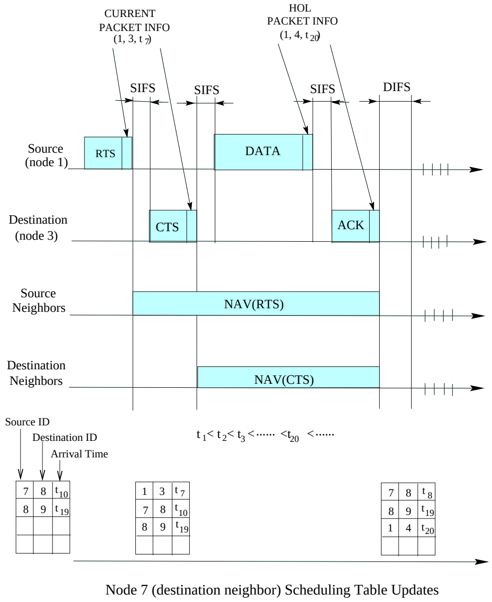


Figure 4: Piggybacking on IEEE 802.11 four-way handshake, and the updating of scheduling tables.

A FIFO schedule is realized by servicing packets in order of their arrival times to the network. To achieve a distributed FIFO schedule among multiple nodes in an *ad hoc* network, we communicate the arrival times of queued packets to other nodes via piggy-backing. Analogous to [8], all nodes will maintain a scheduling table based on overheard information to assess whether or not they possess the next packet for service in the distributed FIFO scheduler. In particular, as illustrated in Figure 4, each packet has an associated arrival time. When a node issues an RTS message, it piggybacks the arrival time of its current packet. Nodes that overhear this RTS insert an entry into a local scheduling table. When the receiving node grants a CTS, it also appends the arrival time in the CTS frame to allow the hidden nodes (node 7 in Figure 4) which are unable to hear the RTS to add an entry in their scheduling tables upon hearing the CTS. Next, when the node transmits the DATA packet, it piggybacks the arrival time of its head-of-line (highest priority) packet, not including the one in transmission, which is also inserted in the local table by overhearing nodes.

With this information, each node can assess the priority of its own head-of-line packet in relation to its (necessarily partial) list of other head-of-line packets. In this way, nodes have the potential to approximate a "global" FIFO reference schedule in a distributed way.

3.2.2 Using Shared Information

The key idea for DWOP is that each node should contend for the channel only when it has the packet with the lowest arrival time (highest priority) among all the nodes within its radio range. The proposed design philosophy is in contrast with that of IEEE 802.11, in which nodes contend for the medium without consideration or knowledge of the arrival times of queued packets at other nodes.

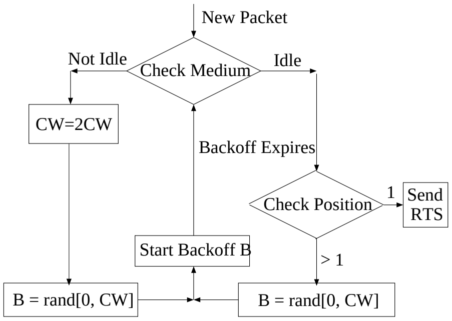


Figure 5: Flowchart for DWOP.

The operation of the DWOP protocol within the framework of IEEE 802.11's backoff policy is depicted in Figure 5. When a node has a packet to transmit, it checks its local scheduling to make a decision regarding if it should contend for the channel. If the node determines (perhaps incorrectly) that its locally queued packet is the highest priority packet in the region (i.e., the node's packet has an arrival time less than all entries in the scheduling table), then the node will contend for the channel as it would in IEEE 802.11. On the other hand, if the node's HOL packet is lower in priority than an entry in its local scheduling table, the node will back off, thereby deferring access to the higher priority packet.

For more complex topologies in which all nodes are not within radio range of each other, asymmetry of information between nodes (Section 2) will prevent DWOP from achieving a perfect FIFO transmission order since not all transmitting nodes are aware of all other nodes' packet arrival times. Consider again the topology in Figure 1(a). In Section 2, it was shown how information asymmetry in IEEE 802.11 causes Flow B to have a larger share of throughput than Flow A. For DWOP, the effect of information asymmetry on the throughput share for each flow is reversed; Flow A achieves a higher share compared to Flow B. The transmitter of Flow A has no knowledge about arrival times of packets queued at the transmitter of Flow B and thus always infers that it has the highest priority packet in the system. Therefore the channel access mechanism for Flow A defaults to IEEE 802.11 and it attempts to gain channel access continuously.

On the other hand, the transmitter of Flow B is aware of Flow A packet arrival times through the receiver of Flow A and thus defers access whenever there is a higher priority packet queued at the transmitter of Flow A. Thus, Flow B is less aggressive in channel access, and in case of the same arrival pattern as at Flow A, always defers channel access after one successful packet transmission. In this case, information asymmetry may cause Flow A to obtain a higher share of bandwidth than Flow B. Note that receiver of Flow A is aware of packet priorities of both the flows in the system, and can thus "forward" the information about Flow B to the transmitter of Flow A. We use the above observation to exploit receiver scheduling table information with an aim to ameliorate information asymmetry.

3.2.3 Receiver Participation

To address the problems introduced by information asymmetry inherent to *ad hoc* networks, we propose an *out-of-order notification* piggybacked on a control packet sent by the receiver to the transmitter, on every FIFO violation according to the receiver's scheduling table.[8] In [1], it was shown that two-hop information, if available, is sufficient to achieve perfect FIFO and eliminate all contention. In essence, the proposed receiver participation technique passes information that is two-hops away from its transmitter, but does so only when needed to avoid the overhead of propagating topology information. Since the notification is sent only when needed, the proposed technique cannot ensure perfect global FIFO rather provides an approximation to FIFO.

On receiving an RTS for a packet that is out-of-order with respect to the receiver's local scheduling table,[9] the receiver can send back a feedback message to the transmitter informing the transmitter of its actual rank with respect to the receiver's local scheduling table. The transmitter, on reception of such an out-of-order notification from the receiver, goes into a backoff after completing the current packet transmission for a time, given by

$$T_{\text{backoff}} = R(EIFS + DIFS + T_{\text{success}} + CW_{\min})$$

where R is the rank of transmitter in the receiver scheduling table, and T_{success} is the longest possible time required to transmit a data packet successfully including handshake (RTS+CTS+DATA+ACK).

The purpose of T_{backoff} is to allow the higher priority packets in the radio range of the receiver to complete transmission. To ensure perfect FIFO, an alternate mechanism to achieve this would be to have the receiver not reply to any RTS that carries a priority tag larger than the smallest entry in the receiver's scheduling table. This would effectively force the transmitter to timeout and backoff, thus avoiding any out of order packet transmission. However, since the transmitter has already expended system resource while transmitting the RTS successfully for the out-of-order packet, we instead allow the present transmission to complete and the receiver piggybacks the out-of-order notification on the CTS/ACK. Hence, the transmitter reacts to the out of order notification *after* the completion of the out-of-order packet. This is a tradeoff between achieving perfect FIFO scheduling and high system utilization. As we show in Section 5 the deviation from the ideal FIFO schedule caused by allowing an out-of-order packet to proceed is minor.

In DWOP, nodes access the channel solely based on their rank in their or the receiver's scheduling table. Thus, the performance of the protocol depends critically on maintaining the consistency of the scheduling tables. We next show how perceived collisions can cause stale entries in the scheduling table and present a novel stale entry detection method.

3.2.4 Perceived Collisions and Stale Entry Detection

Recall that an overhearing node adds entries to its table upon hearing RTS/CTS and removes entries when it hears the successful completion of a packet through DATA/ACK. Thus, the only reason a table can have stale entries is if after hearing an RTS/CTS a node fails to hear the succeeding ACK. In the absence of channel errors and mobility, the reason a node would not hear an ACK after hearing an RTS is because of a collision at that node. This can happen in case the other colliding node was not aware of the previous RTS and thus was not deferring access. This scenario can happen in graphs characterized by perceived collisions as described in Sections II and III.

The effect of inconsistent tables is a possible large deviation from the ideal FIFO schedule if the inconsistency is not corrected. In the worst case, stale entries could lead a node to completely stop transmitting, as it defers access to the stale entry in its table.

Figure 1(b) depicts a topology where stale entries can occur, where both transmitter and receiver of Flow B can have stale entries. Observe that in this case, Flows A and C have no stale entries and continue transmitting. This causes the local scheduling tables at Flow B to continuously update its table by adding and deleting new entries although the position of its HOL packet remains fixed. This is because the new additions/deletions for Flows A and C occur below the position occupied by the HOL packet of Flow B.

Thus, we observe that an indicator of stale table entries is when a node's own packet position remains fixed with entries below the HOL packet entry changing.[10] We use the above observation as a stale entry detection method to estimate when a given node may have stale entries in its local scheduling table. Thus, when a node observes that its position remains fixed although packets with priority below its HOL packet are being transmitted, it immediately concludes that it has one or more stale entries in its table. In this way

---

[8] Note that this is in contrast to IEEE 802.11 where the receiver plays more of a passive role, replying to only packets sent by the transmitter.

[9] Note that since the transmitter chooses to send an RTS, this implies that the packet has a higher priority than the highest priority packet in the transmitter's scheduling table.

[10] Note that a node's position could also be fixed with entries above the position of the HOL packet changing in the normal (no stale entry) case.

each node can independently identify the existence of one or more stale entries in its local scheduling table.

To remove the stale entry from the table after detection, we propose a simple heuristic that a node simply delete the oldest (smallest arrival time) entry assuming that this was the table's stale entry. As we later confirm through experiments in Section 5, the oldest entry is actually the stale entry in most cases. If it is not, it will be detected and removed in subsequent transmissions via the same mechanism, thus ensuring eventual removal of all stale entries so that the flow will eventually be ranked one in its table and resume contention for the channel.

In Sections 3.3 and Section 5, we confirm that the four mechanisms introduced in Sections *3.2.1-3.2.4* allow DWOP to closely approximate a FIFO schedule.

3.3 Protocol Analysis

In practical systems, nodes typically have no initial knowledge of the system. In Section 4, it is shown that DWOP converges to the information steady state where all nodes have heard from their neighboring nodes. In this section, we will study the protocol FIFO behavior in information steady state. To proceed, we require the following two definitions.

DEFINITION 1 (INFORMATION STEADY STATE). *The system is in steady state when it has information about all contending flows in its scheduling table, where the number of entries is equal to incoming strong degree for the transmitters and incoming weak degree for the receivers.*

DEFINITION 2 (TANDEM PACKET CAPTURE). *Assume that all nodes transmit equal size packets of duration* T_{length} *seconds. Further assume that all nodes are continuously backlogged. Let* $\{t_1, t_2, \ldots, t_k\}$ *represent the time of channel capture by an arbitrary subset of k nodes in the system, such that* $t_1 \leq t_2 \leq \cdots \leq t_k$*. The channel capture by the k nodes is said to be in tandem if* $t_1 + 2T_{length} \leq t_k - CW_{\min}$.

The following proposition shows that in a simple fully connected topology, DWOP achieves a perfect FIFO schedule in steady state.

PROPOSITION 1 (PERFECT FIFO CASE). *If all nodes are in within radio range of each other and are continuously backlogged, then the proposed protocol achieves perfect FIFO in the steady state.*

Proof: In steady state with constant backlogged flows, each node is aware of all HOL packets of every other flow. This complete information implies that every node is aware of their global position, and hence never contends if it is not rank one. Since there are no collisions, the top ranked node always acquires the channel, implying perfect FIFO. □

The above proposition is the motivation for preserving determinism in the protocol. In the presence of correct information, the protocol minimizes out of order transmissions. When the nodes are not within radio range of each other, perfect FIFO is not achievable in all cases. The following result partially characterizes the performance of DWOP for general topologies.

THEOREM 1 (FIFO BOUND). *Consider an arbitrary network of nodes with each node transmitting equal size packets*[11] *under constant backlog. Further assume that none of the nodes have any stale entries. In the information steady state, the transmission delay for a flow* f *with the highest priority packet is characterized as follows:*

1. *If flow* f *has no outgoing weak edges, it will transmit without any out-of-order transmissions before it.*
2. *If flow* f *has at least one outgoing weak edge and the tandem packet capture condition is not satisfied for the neighboring nodes with asymmetric connection, then* f *will have to wait for no more than one out-of-order packet transmission.*

Proof: From the hypothesis, stale entry detection is never triggered to delay the top ranked packet.

1. For flow f with no weak outgoing edge, all its outgoing edges have to be strong. Thus, none of the contending flows will attempt to capture the channel out of order, which will allow n to transmit immediately.

2. Let $\mathcal{A}$ denote the set of flows with weak incoming edge from flow f with the highest priority packet. Due to receiver out-of-order notification, all competing flows in set $\mathcal{A}$ will backoff for at least one packet duration. Thus, if the tandem packet condition is not satisfied for flows in $\mathcal{A}$, then after successful transmissions by them, flow f will transmit.[12]

If the tandem packet capture condition is satisfied for the neighboring nodes with asymmetric connectivity, then the node may have to wait more than two packet transmissions but will eventually send the packet with probability increasing (due to random choice of backoff counters by each node) with time.

In the presence of stale entries, the delay in transmission of the highest priority packet depends on the time it takes to detect the stale entry. Furthermore, in the absence of the backlogging condition for some of the contending nodes, the time for successful transmission can deviate from the bounds in Theorem 1. The impact of traffic variations will be analyzed in the next section.

4. SYSTEM DYNAMICS AND TRANSIENT BEHAVIOR OF INFORMATION SHARING

In realistic scenarios, each node will not have perfect information about other nodes' head-of-line packets. For example, when a mobile node moves to within radio range of a new set of users, the new node does not have information about the others' HOL packets and vice versa. Similar conditions occur when a new user enters the system and when a sleeping node wakes up. In such cases, there is a transient time until the new node overhears sufficient information for the system to reach the information steady state, in which all nodes have information about all other backlogged nodes within their radio range.

In this section, we study this transition time for two cases. For the transient case, we compute the time to reach the steady state for a scenario in which no node has information about any other node. For the perturbation case, we compute the time to return to the steady state when a new node enters a system consisting of a number of nodes that

---

[11] The analysis easily extends to unequal size packets with a modification to tandem packet capture condition

[12] In the worst case, $t_k - t_1$ is approximately equal to one packet transmission.

had previously reached the steady state. In both cases, our analytical model indicates that the system converges significantly faster than the time required for an analogous IEEE 802.11 system to allow each user to transmit.[13]

4.1 Model Description

In the following analysis, we consider a region of m mobile nodes, all within radio range of each other, such that each node has at least two packets in its queue. Hence each transmitted packet always carries information of the next HOL packet in the node. Also in order to simplify the analysis, we assume that the values of random timer backoffs of all nodes are independent and uniformly distributed in the range of $[0, C_w]$ and ignore collisions.

4.2 Relationships Among Backoff Timers

Here, we compute two distributions regarding the ordering of backoff timers, which play an important role in system dynamics.

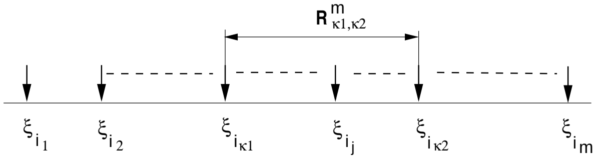


Figure 6: Ordered Backoff Timers

Let $\xi_i, i = 1, \cdots, m$ denote the backoff timers chosen by the m nodes. We rearrange $\xi_i, i = 1, \cdots, m$ in increasing order $\xi_{i_1} \leq \xi_{i_2} \leq \cdots \leq \xi_{i_m}$ (see Figure 6). According to [4], we have that the probability distribution of the random variable $R^m_{k_1,k_2} = \xi_{i_{k_2}} - \xi_{i_{k_1}}$ $(k_2 > k_1)$ is given by:

$$P[R^m_{k_1,k_2} = r] = \frac{m! r^{k_2-k_1-1}(C_w - r)^{m-k_2+k_1}}{(k_2 - k_1 - 1)!(m - k_2 + k_1)! C_w^m}. \qquad (2)$$

Denote Z^h_m as the probability that node i_1 consecutively transmits h packets before node i_2 transmits a packet. This probability is equivalent to the probability that node i_1, after sending out its first packet, consecutively picks up h backoff timers $\chi^1_{i_1}, \cdots, \chi^h_{i_1}$ such that $\sum_{k=1}^{h-1} \chi^k_{i_1} \leq R^m_{1,2}$ and $\sum_{k=1}^{h} \chi^k_{i_1} > R^m_{1,2}$ (see Figure 7).

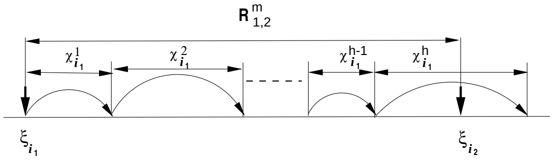


Figure 7: Consecutively Selected Random Numbers

Thus, we have

$$\begin{aligned} Z^h_m &= P[\textstyle\sum_{k=1}^{h-1} \chi^k_{i_1} \leq R^m_{1,2}, \sum_{k=1}^{h} \chi^k_{i_1} > R^m_{1,2}] \\ &= \sum_{r=0}^{C_w} P[\textstyle\sum_{k=1}^{h-1} \chi^k_{i_1} \leq R^m_{1,2}, \sum_{k=1}^{h} \chi^k_{i_1} > R^m_{1,2} | R^m_{1,2} = r] P[R^m_{1,2} = r] \\ &= \sum_{r=0}^{C_w} P[\textstyle\sum_{k=1}^{i-1} \chi^k_{i_1} \leq r, \sum_{k=1}^{i} \chi^k_{i_1} > r] P[R^m_{1,2} = r]. \qquad (3) \end{aligned}$$

Let A_h denote a random variable given by the sum of node i_1's first $h - 1$ backoff timers, i.e., $A_h = \{\sum_{k=1}^{h-1} \chi^k_{i_1} \leq r\}$ and $\bar{A}_h = \{\sum_{k=1}^{h-1} \chi^k_{i_1} > r\}$, then

$$P[\sum_{k=1}^{h-1} \chi^k_{i_1} \leq r, \sum_{k=1}^{h} \chi^k_{i_1} > r] = P[A_h \bar{A}_{h+1}].$$

Since $\chi^k_{i_1}$, $k = 1, 2, \cdots, h$ are independent and identically distributed uniform random variables in the range $[0, C_w]$, we have

$$\begin{aligned} P[A_h] &= \frac{1}{C_w^{h-1}} \int_0^r \int_0^{r-x_1} \cdots \int_0^{r-\sum_{k=1}^{h-2} x_k} dx_{h-1} \cdots dx_2 dx_1 \\ &= \frac{1}{C_w^{h-1}} \frac{r^{h-1}}{(h-1)!}. \qquad (4) \end{aligned}$$

Furthermore, since $\sum_{k=1}^{h-1} \chi^k_{i_1} \leq \sum_{k=1}^{h} \chi^k_{i_1}$, we have that $A_{h+1} \subseteq A_h$, so that

$$P[A_h \bar{A}_{h+1}] = P[A_h - A_{h+1}] = \frac{r^{h-1}}{C_w^h (r-1)!} \left[1 - \frac{r}{C_w h}\right] \qquad (5)$$

Substituting Equations (5) and (2) into Equation (3), we have

$$Z^h_m = \sum_{r=0}^{C_w} [\frac{1}{C_w^{h-1}} \frac{r^{h-1}}{(h-1)!} - \frac{1}{C_w^h} \frac{r^h}{h!}] \frac{m(C_w - r)^{m-1}}{C_w^m}. \qquad (6)$$

4.3 Transient Behavior from the Initial State

Here, we use the above distributions to study the system's transient behavior from an initial state in which all nodes have no information about the HOL packets of any other nodes. With information sharing, the system will evolve from this initial state into the steady state after each node transmits at least one packet.

To address this issue, we compute the probability that the system enters the steady state after n packet transmissions. Let $S(n, m)$ denote the event that a system with m nodes is in the steady state after n packet transmissions. Note that $P[S(n, m)] = 0$ if $n < m$, and $P[S(n, m)] = 1$ if $m = 1$. Furthermore, after sending out their first packet, nodes i_1 and node i_2 will coordinate with each other for channel access since each knows the priority index of the other's next packet. From the performance analysis point of view, after node i_2 sends out its first packet, nodes i_1 and i_2 can be treated as a single "virtual node" and the system can be treated as consisting of $m - 1$ nodes still in the initial state. Therefore, we have following chart for the evolution of the system from the initial state to the steady state.

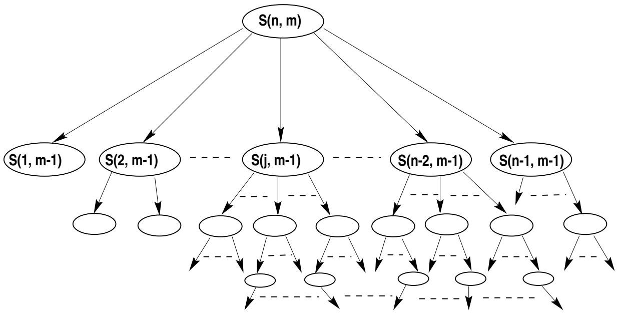


Figure 8: System Evolution from the Initial State

---

[13] Note that each user must transmit at least one packet to reach the steady state.

Thus, using the state diagram of Figure 8, we have $P[S(n,m)] = \sum_{h=1}^{n-m+1} Z_m^h P[S(n-h, m-1)]$, where Z_m^h is given in Equation (6). The same procedure yields

$$P[S(_{n-l,\, m-k})] = \sum_{h=1}^{n-l-m+k+1} Z_{m-k}^h P[S(_{n-l-h,\, m-k-1})], \quad (7)$$

for $k = 1, 2, \cdots, m-1$, where

$$Z_{m-k}^h = \sum_{r=0}^{C_w} [\frac{r^{h-1}}{C_w^{h-1}(h-1)!} - \frac{r^h}{C_w^h h!}] \frac{(m-k)(C_w - r)^{m-k-1}}{C_m^{m-k}}.$$

Therefore, $P[S(n,m)]$ can be computed using Equations (6)-(8).

We now present numerical and *ns-2* simulation investigations of the system's time to reach the steady state. Figure 9 depicts the transition period's probability distributions (in units of packets) for $m = 4$ and 8 nodes. We consider $C_w = 32$, the same as the minimum contention window size given in [7]. In order to highlight the impact of information sharing on the transient process, we also present the distributions of the duration for each node to send out at least one packet in the standard IEEE 802.11 MAC protocol.

We make following observations regarding the figure. First, after $2m$ packet transmissions, the probability for the system with information sharing to enter the steady state is more than 0.95. For example, in Figure 9(a), after 8 packets are transmitted in a system with 4 flows (4 source nodes and 4 destination nodes), the probability for the system to enter the steady state is 0.97 from the model's prediction and 0.98 from simulation. Second, observe that this duration is significantly less than the time required for the IEEE 802.11 MAC protocol to reach a state in which every node has transmitted at least one packet. For example, for this to occur in the system with 4 flows with probability 0.85, 5 packets must be transmitted with information sharing, whereas 9 packets must be transmitted in IEEE 802.11. Thus, information sharing accelerates the system into an steady state in which each node has transmitted at least one packet.

4.4 Transient Behavior from the Perturbation State

Mobility, sleeping nodes awakening, etc., will perturb the system from its steady state and cause multiple nodes to contend for the channel simultaneously. To evaluate the distribution of the duration for the system to return to the steady state, we treat the existing m nodes as a virtual node denoted by A, since there exists an order among these m nodes such that only one contends for the channel with with new nodes. To simplify analysis, we consider a single new node indexed by $m+1$. Without loss of generality, we assume that the priority of the k^{th} packet of virtual node A is higher than the priority of the first packet of node $m+1$ and the priority of the $(k+1)^{th}$ packet of node A is lower than the priority of the packet of node $m+1$. We also assume that node $m+1$ will know if node A has another packet with higher priority than itself packets after node A transmits a packet. This situation will occur if each original node has at least two packets with higher priorities than node $m+1$ packets or every node piggybacks the information about the packet with the second highest priority in its scheduling table as well as the information of its next packet.

Under this assumption, the probability that the virtual node A transmits $n-1$ packets before node $m+1$ transmits its first packet is given by $0.5Z_{2,1}^{n-k-1}$. With probability 0.5, virtual node A obtains a backoff timer smaller than that of node $m+1$ and sends out k consecutive packets. This will occur because the new node $m+1$ learns that virtual node A has packets with higher priorities and waits until virtual node A sends out the k^{th} packet. With probability Z_2^{n-k-1}, as defined in Equation (3), virtual node A will consecutively send out $n-k-1$ packets with lower priorities than the first packet of node $m+1$ before the perturbed system returns to the steady state. Similarly, the probability that node $m+1$ continuously transmits $n-1$ packets before the perturbed system returns to the steady state is given by $0.5Z_2^{n-1}$. Specifically, if $n \leq k$ and virtual node A first captures channel, then it is impossible for the system to return to the steady state after n packets having been transmitted. This is because virtual node A will continuously transmit k packets before node $m+1$ transmits its first packet.

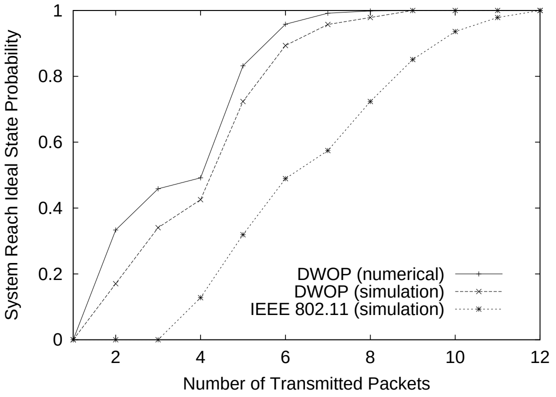


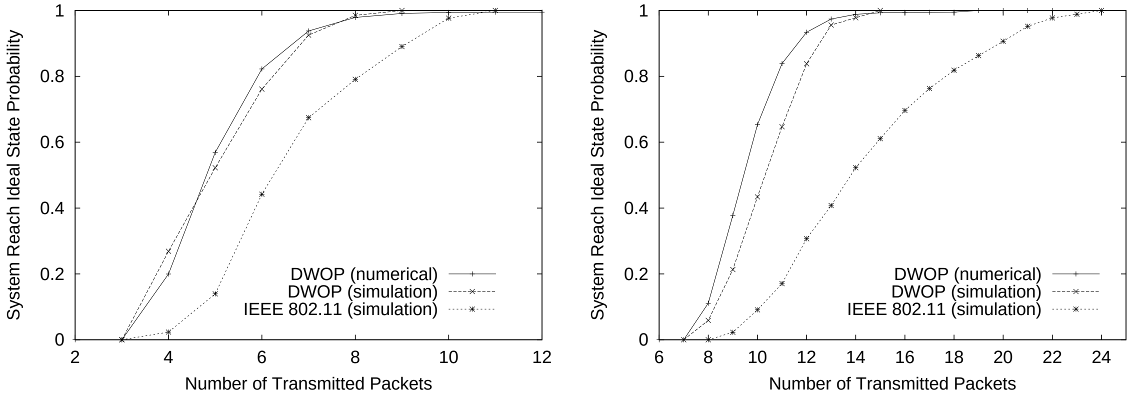
Figure 10: Probability Distribution of Perturbation Period Length

Let $P(n,k)$ denote the probability that the system returns to the steady state after n packets having been transmitted. According to above analysis, we have

$$P(n,k) = \begin{cases} 0.5Z_2^{n-1}, & n \leq k, \\ 0.5Z_2^{n-k-1} + 0.5Z_2^{n-1}, & n > k. \end{cases} \quad (9)$$

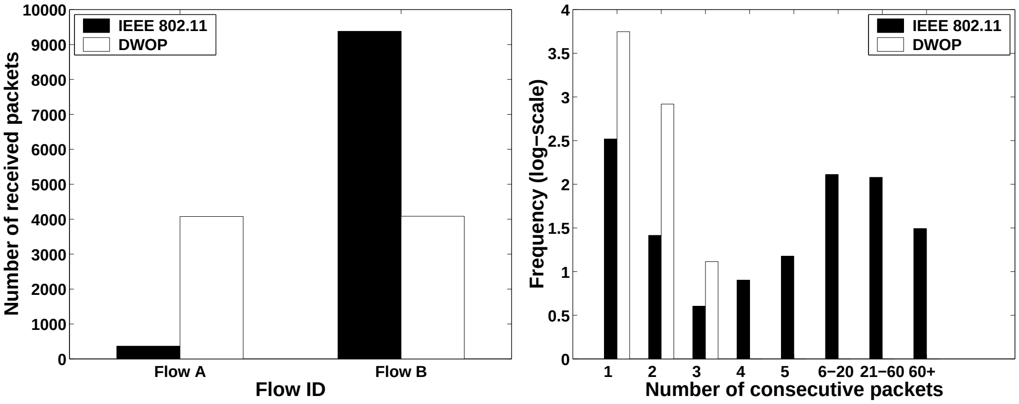
Finally, in Figure 10 we present numerical and simulation investigations on the duration required for the perturbed system to return to the steady state. We consider a system with 8 nodes (4 source nodes and 4 destination nodes) in the steady state and 2 nodes (one source node and one destination node) joining the system. We further consider that there are initially 4 packets in the original nodes with higher priority than the first packet of the new node.

We make the following observations about the figure. First, note that information sharing allows the system to rapidly return to the steady state, as compared to waiting for each node to transmit in IEEE 802.11. Second, note that there is an inflection point in the distribution for information sharing occurring at 4 packet durations. This is corresponds to the number of packets in the original nodes with higher priority than the first packet of new nodes. The reason for this inflection is that when one of the original nodes captures channel, there will be 4 packets transmitted before the new source node transmits its first packet and the perturbed system cannot return to the steady state because the



(a) 4 Flows (b) 8 Flows

Figure 9: Probability Distribution of Transition Duration



(a) Throughput Comparison (b) Packet Order Distribution

Figure 11: Comparison for DWOP and IEEE 802.11 for Asymmetric Topology

original nodes do not have information about this node's HOL packet. Therefore, the probability for the system to return to the steady state after 4 packet transmissions depends only on the case in which node $m+1$ captures the channel first and continuously transmits 3 packets before one of the original nodes sends out its packet. Thus, the distribution duration for the system to return to the steady state increases slowly before reaching 4 packet transmission times quickly thereafter.

5. SIMULATION EXPERIMENTS

In this section we present simulation results to compare the proposed DWOP protocol with IEEE 802.11 with FIFO as the reference scheduler. The simulations were performed using the CMU Monarch wireless extensions to the *ns-2* simulator.[14]

We consider three topologies without channel errors and node mobility. The data packet size is set to 1000 bytes, while the data capacity of the wireless channel is 2 Mb/sec. For input traffic, we use constant-rate flows for each node with a slight jitter in inter-arrival times. To simulate heavy loads, we set the input rate for each flow to be high enough to individually saturate the channel; 5 random runs of 50 seconds each were performed for each test case. All other physical layer parameters were set to the default parameter values in *ns-2*. All flows are single hop, but all topologies consist of nodes which are out of radio range of at least one node in the network. Further, two sets of simulations are performed for each topology, with carrier sense threshold the same as the data threshold, and with the carrier sense threshold smaller than the data threshold (default values from CMU extensions were used in this case). The results for both values of carrier sense thresholds were found to be similar, and hence we primarily present them for the case where carrier sense threshold is equal to data threshold, to highlight the role of the graph-theoretic representation.

5.1 Asymmetric Information Topology

Here we present results for the topology shown in Figure 1(a). The simulation results are shown in Figure 11(a), which compares the throughput of IEEE 802.11 with DWOP. For IEEE 802.11, asymmetry of information helps Flow B to obtain 95% of total throughput whereas Flow A obtains only about 5% (the share is 70-30% with a smaller carrier-sense threshold). For DWOP, both flows have an equal share of throughput. For experiments with the carrier sense smaller than the data threshold, the total throughput is nearly identical to that of IEEE 802.11, whereas for the depicted case

---

[14] Available from http://www.isi.edu/nsnam/

with identical thresholds, the total throughput of DWOP is approximately two-thirds that of IEEE 802.11.

Figure 11(b) shows the distribution of the number of consecutive packets sent by any flow before it relinquishes the channel to the other flow.[15] For IEEE 802.11, the distribution is spread out with a single flow (Flow B) keeping the channel for a large number of consecutive packets (the maximum number of consecutive packets transmitted by Flow B is 129). For DWOP, a flow never transmits more than 3 consecutive packets. Thus we see that the DWOP approximates the ideal FIFO schedule significantly more closely than IEEE 802.11.

Figure 12 depicts the number of packets sent by each flow sampled at 1 second intervals. For DWOP, both flows have equal throughput at this time scale. On the other hand, with IEEE 802.11 Flow B starves out Flow A consistently over the entire duration of the simulation.

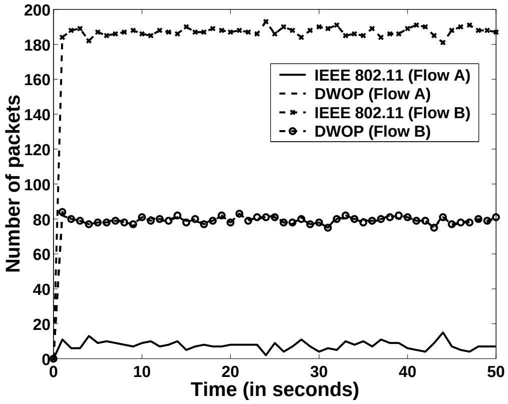


Figure 12: Bandwidth Share for Asymmetric Topology

5.2 Perceived Collision Topology

In this section, we consider the topology considered in Figure 1(b), to study the impact of perceived collisions on the throughput share of Flow B and the extent of resulting deviation from reference FIFO.

Figure 13(a) compares the throughput of IEEE 802.11 with DWOP. For IEEE 802.11, Flow B obtains a smaller throughput share whereas Flows A and C approximately divide the rest of the share equally. The reasons for unequal share were discussed in Section 2.4. However DWOP allows nearly equal share of the net throughput for all three flows, although with a net throughput of three-fourths (86% with a smaller carrier sense threshold) that of IEEE 802.11. Figure 13(b) shows the distribution of consecutive packets sent by a flow. As shown, IEEE 802.11 violates FIFO with a single flow keeping the channel for a maximum of 77 packets, while DWOP closely approximates FIFO with no flow keeping the channel for more than 4 consecutive packets, thereby confirming the efficacy of proposed stale entry detection mechanism.

5.3 10 Node Topology

In this section, we present the results for a more complex 10-node topology shown in Figure 14; in this topology, both Flows B and C can have stale entries. Figure 13(c) compares the throughputs obtained by IEEE 802.11 and DWOP. Observe that Flow C which has the maximum strong information, obtains the lowest throughput in IEEE 802.11. Flows B, D and E practically divide the bandwidth among themselves due to asymmetric information and spatial reuse. However, DWOP results in a near equal share of the net throughput being allocated to all the flows, albeit at a total throughput loss. The total throughput of DWOP is three-fifths (two-third with lower carrier sense threshold) that of IEEE 802.11.

Figure 13(d) depicts the distribution of consecutive packets sent by a flow. Observe that IEEE 802.11 violates FIFO ordering with a single flow keeping the channel for a maximum of 8 packets. However, this plot coupled with 14(a) shows that it is in fact Flow B, D and E that keep relinquishing the channel to each other. In contrast, DWOP closely approximates FIFO with no flow keeping the channel for more than 2 consecutive packets.

6. RELATED WORK

Distributed scheduling and media access to achieve fair bandwidth allocation in *ad hoc* wireless networks has been an intensive research topic in recent years, e.g., [10-13,15-17]. By exploiting the broadcast nature of the wireless medium, all of these schemes use some form of information sharing to allow distributed nodes to cooperate with each other to achieve a desired global behavior. For example, with passive information sharing (i.e., using measured information about channel idle times, collisions, etc.), the authors of [13] devise a distributed dynamic p-persistent MAC protocol designed to achieve proportional fairness. Using active information sharing (i.e., piggybacking), the authors of [17] devise a scheme to emulate Self-Clocked Fair Queueing (see also [5]) by piggybacking local virtual times and adjusting IEEE 802.11 backoff policies accordingly. Finally, the authors in [11] introduce three localized fair queueing models within the framework of the CSMA/CA paradigm to let distributed nodes to emulate Start-Time Fair Queueing (see also [6]) and achieve global weighted fairness in *ad hoc.*

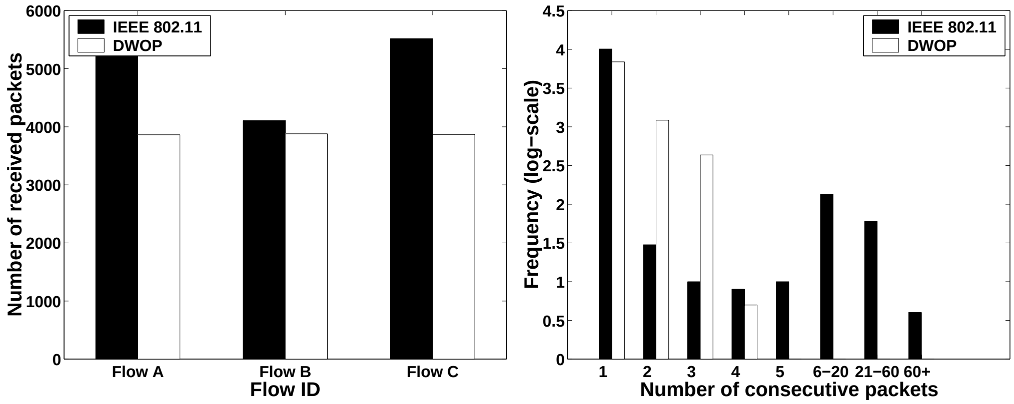
In contrast, our objective is to provide the reference scheduler service order at the MAC layer rather than per-flow or per-node fairness. This objective is shared with [2] as well as a degenerate case of the distributed priority scheduler in [8]. In contrast to [2], we consider complex topologies in which complete information is not available, and provide a graph-theoretic framework and protocol to address these topologies. In contrast to [8], we target a more deterministic behavior so that near-exact desired service ordering is achieved in most cases, whereas [8] focuses on meeting delay and rate targets. Thus, in [8], the precise service ordering is not a focus, so long as the quality-of-service targets are satisfied.

7. CONCLUSIONS

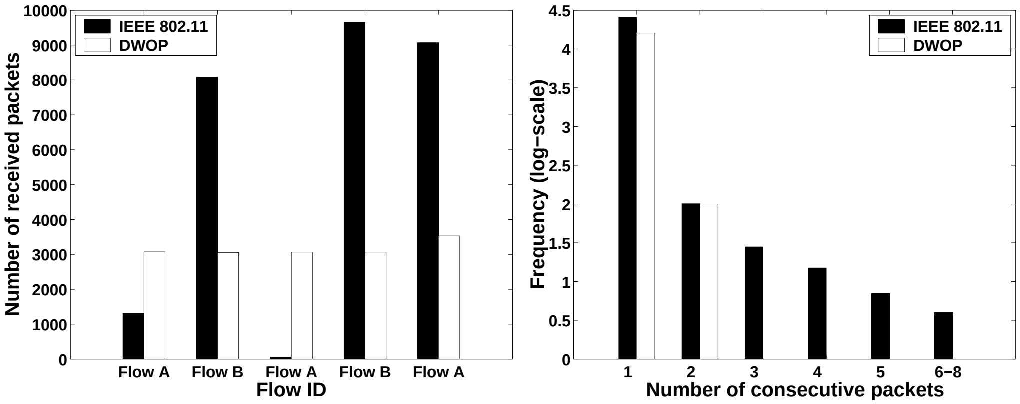
The goal of this work is to design a distributed media access and scheduling algorithm to achieve desired service order in wireless *ad hoc* networks. Choosing FIFO as an example target scheduler, we showed that the IEEE 802.11

---

[15] For FIFO with constant-rate arrival patterns and identical rates, we expect the service to alternate among flows. Thus, in an ideal FIFO system no flow should keep the channel for more than one packet.



(a) Throughput Comparison (b) Packet Order Distribution



(c) Throughput Comparison (d) Packet Order Distribution

Figure 13: Comparison of DWOP and IEEE 802.11 for Perceived Collision and 10-node Topology

protocol diverges significantly from FIFO order, even starving nodes in many cases, due to asymmetric information sharing and "perceived collisions". We showed via simulations and theoretical analysis that DWOP exploits information sharing to achieve nearly perfect service order, even in complex topologies with incomplete information, and in dynamic scenarios beginning with no information.

8. REFERENCES

[1] L. Bao and J. J. Garcia-Luna-Aceves. A new approach to channel access scheduling for ad hoc networks. In *Proc. ACM MOBICOM*, 2001.

[2] C. Barrack and K.-Y. Siu. A distributed scheduling algorithm for quality of service support in multiaccess networks. In *Proc. IEEE ICNP '99*, Oct. 1999.

[3] V. Bharghavan, S. Demers, S. Shenker, and L. Zhang. MACAW: A media access protocol for wireless LANs. In *Proc. ACM SIGCOMM '94*, 1994.

[4] E. Castillo. *Extreme Value Theory in Engineering.* Academic Press, 1988.

[5] S. Golestani. A self-clocked fair queueing scheme for broadband applications. In *Proc. IEEE INFOCOM*, 1994.

[6] P. Goyal, H. M. Vin, and H. Chen. Start-time fair queueing: A scheduling algorithm for integrated services packet switching network. *IEEE/ACM Trans. Netw.*, 5(5):690–704, Oct. 1997.

[7] IEEE. IEEE standard 802.11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications, 1997.

[8] V. Kanodia, C. Li, A. Sabharwal, B. Sadeghi, and E. Knightly. Distributed multi-hop scheduling and medium accsess with delay and throughput constraints. In *Proc. ACM MOBICOM*, 2001.

[9] J. Li, C. Blake, D. S. J. Decouto, H. I. Lee, and R. Morris. Capacity of ad hos wireless networks. In *Proc. ACM MOBICOM*, 2001.

[10] H. Luo and S. Lu. A topology independent fair queueing model in Ad Hoc wireless networks. In *Proc. IEEE ICNP*, Aug. 2000.

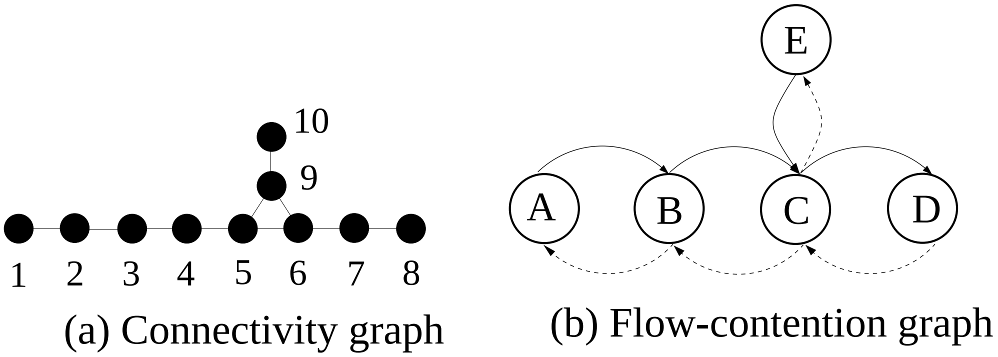
[11] H. Luo and S. Lu. A self-coordinating approach to distributed fair queueing in ad hoc wireless networks. In *Proc. IEEE Infocom*, 2001.

[12] H. Luo, S. Lu, and V. Bharghavan. A new model for packet scheduling in multihop wireless networks. In *Proc. ACM MOBICOM*, Aug. 2000.

[13] T. Nandagopal, T. Kim, X. Gao, and V. Bharghavan. Achieving MAC layer fairness in wireless packet networks. In *Proc. ACM MOBICOM*, Aug. 2000.

[14] B. O'Hara and A. Petrick. *IEEE 802.11 Handbook, A Designer's Companion.* IEEE Press, 1999.

[15] T. Ozugur, M. Naghshineh, P. Kermani, and J. Copeland. Fair media access for wireless LANs. In *Proc. IEEE GLOBECOM*, 1999.



(a) Connectivity graph (b) Flow-contention graph

Figure 14: A 10-node Topology

[16] N. Vaidya and P. Bahl. Fair scheduling in broadcast environments, Aug. 1999. Microsoft Research Tech. Rep. MSR-TR-99-61.

[17] N. Vaidya, P. Bahl, and S. Gupta. Distributed fair scheduling in a wireless LAN. In *Proc. ACM MOBICOM*, Aug. 2000.

[18] L. Zhang. *A New Architecture for Packet Switched Network Protocols.* PhD dissertation, Massachusetts Institute of Technology, July 1989.