Distributed Node Scheduling Algorithms for Multiple Group Communications in Wireless Multi-hop Networks

I-Hsien Liu[1], Chuan-Gang Liu[2], Kun-Hsuan Liu[1], Shun-Hsiung Yu[1],
Zhi-Yuan Su[3], and Jung-Shian Li[1](✉)

[1] Department of Electrical Engineering, Institute of Computer and Communication, National Cheng Kung University, Tainan, Taiwan, R.O.C.
{dannyliu, Khliu, sxyu}@hsnet.ee.ncku.edu.tw,
jsli@mail.ncku.edu.tw
[2] Department of Applied Informatics and Multimedia, Chia-Nan University of Pharmacy and Science, Tainan, Taiwan, R.O.C.
chgliu@mail.cnu.edu.tw
[3] Department of Information Management, Chia-Nan University of Pharmacy and Science, Tainan, Taiwan, R.O.C.
szj1974@gmail.com

Abstract. We study the scheduling problem in performing multiple multicast communications in wireless multi-hop networks, it is necessary to ensure that each multicast group can complete one transmission from the source to all the destination nodes without conflict in every frame. The present study proposes two distributed token-based STDMA node scheduling algorithms which not only satisfy this requirement, but also minimize the frame length. In the first algorithm, the multicast groups are scheduled on a group-by-group basis, whereas in the second algorithm, multiple groups are scheduled in each scheduling operation. The first algorithm has the advantages of computational simplicity and a straightforward implementation, while the second algorithm increases the percentage of reused time slots and reduces the number of token forwarding events. The simulation results show that both algorithms achieve a shorter frame length than existing methods.

Keywords: Group communications · Wireless multi-hop networks · Node scheduling

1 Introduction

Recent years have witnessed a surge in the popularity of distributed applications such as audio and video conferencing, media streaming, interactive gaming, and so on. The transmission efficiency of such applications is generally enhanced by adopting a multicast broadcasting approach. Many of these applications require strict Quality of Service (QoS) guarantees (e.g., a minimal delay and a fair share of the available bandwidth). However, traditional wireless Medium Access Control (MAC) protocols are generally unable to satisfy these requirements. Furthermore, in ad hoc networks, traditional protocols often result in a low throughput due to collisions in the upstream

J.-H. Lee and S. Pack (Eds.): QShine 2016, LNICST 199, pp. 120–128, 2017.
DOI: 10.1007/978-3-319-60717-7_12

direction. As a result, more efficient MAC protocols for multicast group communications are required.

Most existing MAC protocols are based on a Time Division Multiple Access (TDMA) scheme, in which the time domain is divided into contiguous fixed-length slots and these slots are allocated in such a way that only one node or link is active at any moment in time. However, TDMA schemes result in a poor bandwidth utilization in wireless multi-hop networks since they do not support spatial reuse. That is, different nodes cannot access the shared channel concurrently even if their assigned wavelengths are widely separated in the transmission spectrum. Accordingly, several Spatial TDMA (STDMA) schemes have been proposed for improving the capacity of ad hoc networks by permitting multiple nodes to transmit simultaneously provided that they are collision-free [2, 5, 16]. Such schemes ensure that each network node can transmit at least once in every frame. Moreover, most STDMA schemes also attempt to minimize the number of assigned slots; thereby increasing throughput. However, the problem of finding the minimal frame-length is NP-complete [12].

Accordingly, the present study proposes a distributed node scheduling algorithms for minimizing the frame length for multiple multicast communications in wireless multi-hop networks. In the former algorithm, the various multicast groups are scheduled on a group-by-group basis, whereas in the second algorithm, multiple groups are scheduled in the same frame. In both cases, the nodes select an appropriate number of time slots in accordance with their respective loads. Moreover, in each algorithm, the time slots are selected in such a way that every multicast group can complete the transmission of one packet from the source to all the destinations without conflict in every frame.

2 Background and Related Work

In STDMA networks, data collisions (i.e., transmission failures) arise as a result of two different types of interference, namely "primary interference" and "secondary interference" [3, 8, 10, 12, 14, 15]. Primary interference occurs when a node is required to carry out more than one task in the same slot (e.g., to both receive and transmit data). Meanwhile, secondary interference occurs when the data receiving process of one node is interfered with by the data transmission process of a nearby node (i.e., the "hidden node problem" [9]).

In solving the data collision problem, many existing node scheduling algorithms use a chromatic approach, in which the network nodes are partitioned into different color classes, where those nodes with the same color are able to transmit simultaneously without collision [1, 5, 11, 13]. However, although such an approach ensures that every node can transmit at least once in a frame without conflict, many slots may be wasted since the scheduling process does not take the network load into account. For example, some nodes may be allocated a time slot, but may not actually have any data to transmit once the slot arrives. Accordingly, the scheduling algorithms proposed in this study take explicit account of the transmission flow within each multicast group and allow each node to select appropriate time slots in accordance with their particular transmission requirements.

3 Problem Definition, Network Model and Proposed Algorithms

3.1 Problem Definition and Network Model

The aim of the algorithms proposed in the present study is to improve the transmission efficiency of multiple multicast communications in wireless multi-hop ad hoc networks. Moreover, based on the assumption that each activated node can transmit only one packet per time slot, the proposed algorithms also attempt to guarantee that all of the multicast groups can complete the transmission of one packet from the source to all of the destination nodes in every frame. Finally, both algorithms seek to minimize the frame length. Note that in developing the two algorithms, it is assumed that a node can pick a slot to transmit for a particular group if, and only if, its upstream node in that group has already been scheduled. In addition, it is also assumed that the frame length is identical for all of the nodes in the network.

The wireless multi-hop network is modeled using a bidirectional graph G = (V, E), where V is the set of nodes, |V| is the number of nodes, and E is the set of edges between the nodes. If u,v ∈ V, there exists an edge(u, v). Furthermore ∈E indicates that node v can receive all the messages transmitted by u, and vice versa. Finally, it is assumed that all of the nodes in the network are homogeneous, i.e., the nodes all have the same transmission range.

In modeling the network, each node is assigned a unique ID and is assumed to be aware of the connectivity information of the network within its transmission range. That is, every node knows the IDs of all its one-hop neighbors. Since the present study considers a multiple multicast communications network in which every node in the network is a source, the network contains |V| source-based trees. In other words, every node participates in |V| multicast groups, and the multicast packets of each group are routed along a specific source-based tree. In addition, it is assumed that every node knows how to forward the multicast data which it receives. In other words, when a multicast packet arrives at a node, the node knows which of its links should be activated to forward the packet toward the other nodes in the source-based tree. For example, consider the source-based tree shown in Fig. 1, in which Node S is the

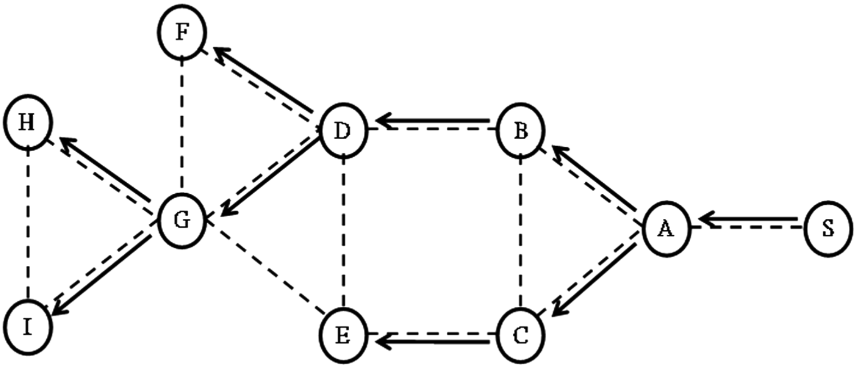


Fig. 1. Illustrative source-based tree

source. Node D (for example) lacks the connectivity information of the entire network, but knows that Nodes B, E, F and G are its one-hop neighbors. Thus, when Node S sends a multicast message to all its group members, Node D forwards the message to all its downstream nodes (i.e., Nodes F and G) on receipt of the message from Node B. Note that in this illustrative scenario, Node B is the parent of Node D, and Nodes F and G are the children of Node D.

Note that in developing the proposed scheduling algorithms, it is assumed that the network topology remains unchanged during the scheduling process. Moreover, an assumption is made that all of the packets sent by the nodes are correctly received at their destinations. In other words, the transmission failure problem is not considered.

3.2 Group-by-Group Algorithm

Every multicast group in the multi-hop network is required to complete the transmission of one packet from the source to all of the destinations in every frame. In other words, every multicast group must be scheduled in each scheduling round. In the first algorithm proposed in the present study, designated as the Group-by-Group algorithm [6], the groups are scheduled on a group-by-group basis. Specifically, when a group is allocated to schedule, all of the nodes in the group select appropriate slots in which to perform transmission, and once all of these nodes have been scheduled, the scheduling process is repeated for the next group.

3.3 Greedy Algorithm

The Group-by-Group algorithm described in the previous section provides a straightforward means of achieving multiple multicast scheduling in ad hoc wireless networks. However, it has two drawbacks. First, the scheduling process is time consuming since each node can select only one time slot for transmission when receiving the token. Second, nodes within the same multicast group cannot choose the same slot for transmission, and thus the channel utilization efficiency is reduced. Accordingly, this section proposes a second algorithm, designated as the Greedy algorithm, which eliminates both drawbacks by allowing multiple groups to be scheduled in the same frame.

In the proposed algorithm, each node in the wireless network maintains a schedule queue containing a list of scheduling jobs to be performed by either itself or one of its one-hop neighbors. For each node, the jobs (J) are recorded using the data structure Jm n, where m is an index relating to the different multicast groups in the network and n is an index pertaining to the different slots in the time domain. Assume that a job Jm n is stored in the schedule queue of Node k. In practice, the existence of this job in the schedule queue has two implications: (1) the parent of Node k in Group m has already been scheduled (and thus Node k can pick a slot to transmit for Group m when it receives the token), and (2) the parent of Node k in group m has selected time slot n to perform its transmission. In other words, the schedule queue indicates to Node k which groups have already been scheduled by its upstream nodes and allows the node to pick an appropriate slot to perform its own transmission for each group. (Note that if Node k chooses to perform transmission for multicast Group m, the index of the selected slot must be greater than n.) Each schedule queue can store multiple jobs; where each job may refer to different multicast groups. Thus, in contrast to the Group-by-Group algorithm, the nodes in the network can schedule the transmissions of multiple groups in the same scheduling frame, As a result, the Greedy algorithm results in a better than the Group-by-Group algorithm. In addition to the scheduling queue, each node in the network also maintains a schedule table for itself and its one-hop neighbors. As in the Group-by-Group algorithm, the schedule table is used to record the state of each node in every time slot (i.e., Ti, Rj, DR or empty).

The scheduling process in the Greedy algorithm is again controlled using a token-based scheme. However, in contrast to the Group-by-Group algorithm, the token has the format {terminator, frame length}, where the "terminator" field is used to indicate which node wants to terminate the scheduling process. Note that the "terminator" field is generally set to 0, i.e., none of the nodes in the network wish to terminate the scheduling process. However, if the token arrives at a node, and the node finds that its schedule queue is completely empty (i.e., there are no outstanding jobs for either itself or any of its one-hop neighbors), the node replaces the current "terminator" field entry with its own node-ID since it believes that all of the multiple multicast groups in the network have been scheduled. The "frame length" field of the token indicates the current length of the frame, and has a value equal to the maximal time slot index amongst all of the time slots which have been selected thus far in the scheduling operation.

4 Simulation Results

In this section, the performance (i.e., frame length and number of token forwarding events) of the proposed algorithms is investigated by means of numerical simulations. In accordance with the assumptions described in Sect. 3.1, the network nodes all have the same transmission range. Furthermore, no changes in the network topology occur during the scheduling process. That is, the nodes are all static and the network is always connected. Moreover, every node in the network is a source node with data to transmit to all of the other nodes, and each group has a unique source-based tree (constructed using Dijkstra's algorithm). In performing the simulations, the performance of the two algorithms is evaluated given a network topology with a specific number of nodes and a specific average node degree. To ensure the reliability of the simulation results, 30 networks are randomly generated for each scenario considered, and the corresponding results for the frame length are then averaged.

Figure 2(a) and (b) compare the frame lengths obtained by the Group-by-Group and Greedy algorithms in networks comprising N = 20 and 40 nodes, respectively, with the equivalent results obtained using the DRAND algorithm [4] and a "non-spatial reuse" scheme. Note that the results presented for the non-spatial reuse scheme indicate the number of slots required to complete the scheduling process for all of the nodes given the constraint that each slot can only be allocated to one node. It can be seen that the frame length obtained using the non-spatial reuse scheme reduces with an increasing

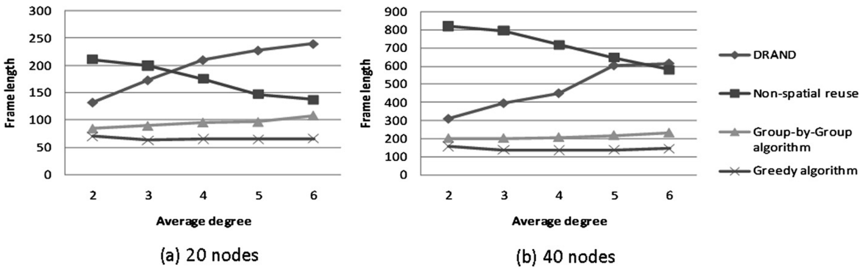


Fig. 2. Frame lengths obtained by various schemes given random network topologies with (a) N = 20 and (b) N = 40 nodes.

average node degree irrespective of the number of nodes in the network due to the nature of broadcast. The DRAND algorithm uses a chromatic approach to resolve the problem of collisions. However, it takes no account of the traffic flow when performing the scheduling process. Moreover, the DRAND algorithm also suffers from the exposed node problem. As a result, the algorithm yields a longer frame length than either of the two proposed algorithms. It is seen that for all values of N, the frame length obtained by the Group-by-Group algorithm is longer than that obtained by the Greedy algorithm. This result is to be expected since the Group-by-Group algorithm prevents the nodes from picking the same slot when scheduling the transmissions of the same group. In other words, the algorithm does not support spatial reuse within each group, and thus a greater number of time slots are required to complete transmission. The Greedy algorithm allows multiple nodes to pick the same slot to perform the transmissions of different groups provided that no collisions occur between them. As a result, the frame length is significantly reduced. In theory, an increasing node degree reduces the frame length in networks with a spatial reuse capability. However, as the node degree increases, the number of collisions also increases; thereby reducing the percentage of spatial reuse. As a result, the frame length obtained by the Greedy algorithm varies only very slightly as the average node degree is increased.

Figure 3 shows the variation of the frame length with the number of nodes as a function of the average node degree for the four considered schemes. In accordance

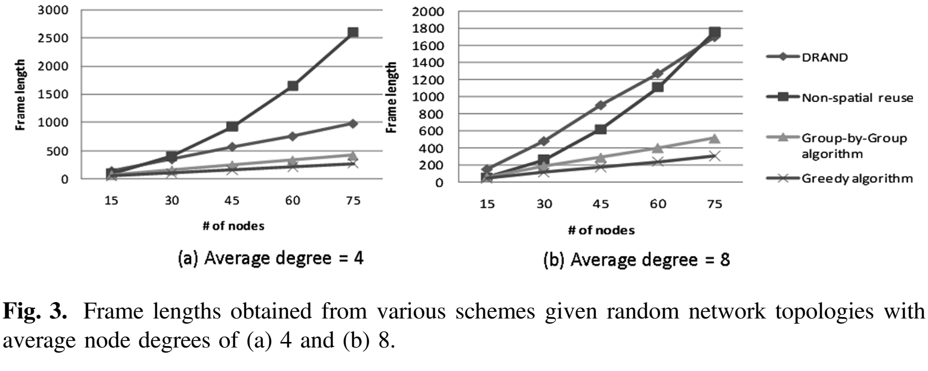
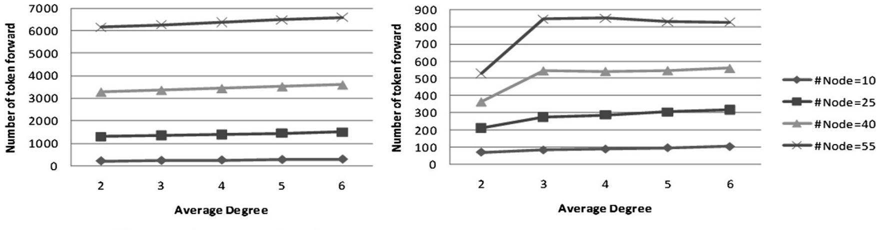


Fig. 3. Frame lengths obtained from various schemes given random network topologies with average node degrees of (a) 4 and (b) 8.

with the network model considered in the present study (see Sect. 3.1), the number and size of the multicast groups in the ad hoc network both increase with an increasing number of nodes. Thus, for all of the considered schemes, the frame length increases as the scale of the network topology increases. In DRAND, each time slot is occupied by a different color. In other words, the transmission period of the nodes is equal to the total number of colors. Thus, although the number of colors in DRAND varies only slightly when fixing for different values of the average node degree, the frame length increases significantly as the number of nodes (i.e., multicast groups) increases. Notably, the two algorithms proposed in the present study both take the traffic flow into account when performing the scheduling process. As a result, they yield a shorter frame length than DRAND; particularly in network topologies with a larger number of nodes (multicast groups).

Figure 4(a) and (b) compare the performance of the two proposed algorithms in terms of the number of token forwarding events given networks with various scales and node degrees. It was shown in Theorem B in Sect. 3.2 that the token is forwarded 2N2-2N+4L times in the Group-by-Group algorithm. It is seen that the simulation results in Fig. 4(a) confirm this proof. Comparing the two figures, it is noted that the Greedy algorithm results in fewer token forwarding events than the Group-by-Group algorithm given a constant network scale and average node degree. This result is reasonable since the Greedy algorithm allows a node to pick more than one slot to transmit different groups in each scheduling operation, whereas the Group-by-Group algorithm permits each node to pick only one slot for one group in each frame. It is observed that for a constant value of N, the number of link forwarding events increases as the average node degree increases under the Greedy algorithm. At first glance, this finding seems counterintuitive since it is reasonable to expect that a higher node degree will reduce the total number of slots which need to be allocated to the nodes and will therefore reduce the number of forwarding events accordingly. However, in the second stage of the Greedy algorithm, the token is routed to all of the nodes via DFS in order to achieve a constant frame length for every node. In other words, the token must be forwarded (L × 2) times and thus the number of forwarding events actually increases as the average node degree increases.



(a) Group-by-Group algorithm

(b) Greedy algorithm

Fig. 4. Number of token forwarding events in (a) Group-by-Group algorithm and (b) Greedy algorithm given random network topologies with various scales and average node degrees.

5 Conclusion

This paper has proposed two token-based distributed node scheduling algorithms designated as the Group-by-Group algorithm and the Greedy algorithm, respectively, for improving the transmission efficiency of multiple multicast communications in multi-hop ad hoc wireless networks. Both algorithms feature the following properties: (1) distributed control; (2) local information exchange; (3) the complete transmission of one packet from the source to all the destination nodes in every frame for each multicast group in the network; (4) transmission traffic flow compliance; and (5) the elimination of both the hidden node problem and the exposed node problem. The simulations have shown that the Greedy algorithm results in a shorter frame length than the Group-by-Group algorithm due to its support of spatial reuse. Moreover, the Greedy algorithm reduces the number of token forwarding events and therefore spends less time to complete scheduling.

Although both algorithms successfully solve the multiple multicast scheduling problem in the considered network model, several important issues remain to be addressed. In the present study, it has been assumed that all of the network nodes remain static as the scheduling process is performed. In other words, the scheduling result is valid only for the particular network topology in existence at the moment the scheduling algorithm is executed. As a result, the scheduling process must be repeated each time a change in the network topology occurs. In practice, the topology of ad hoc networks tends to change frequently as new nodes enter the network or existing nodes depart. Consequently, the efficiency of the proposed algorithms is seriously degraded. Accordingly, in a future study, a more flexible node scheduling algorithm will be proposed to accommodate changes in the network topology.

Acknowledgement. This work was supported in part by the Ministry of Science and Technology, R.O.C. under Grants MOST 103-2221-E-006 -146 -MY3 and MOST 104-2221-E-041-007, Ministry of Education, R.O.C. under Grant 104C-77-016.

References

1. Wolf, B.J., Hammond, J.L., Noneaker, D.L., Russell, H.B.: A protocol for construction of broadcast transmission schedules in mobile ad hoc networks. IEEE Trans. Wireless Commun. **6**, 74–78 (2007)
2. Wang, G., Ansari, N.: Optimal broadcast scheduling in packet radio networks using mean field annealing. IEEE J. Select. Areas Commun. **15**, 250–260 (1997)
3. Chlamtac, I., Pinter, S.S.: Distributed nodes organization algorithm for channel access in a multihop dynamic radio network. IEEE Trans. Comput. **C-36**, 728–737 (1987)
4. Rhee, I., Warrier, A., Min, J., Xu, L.: DRAND: distributed randomized TDMA scheduling for wireless ad hoc networks. IEEE Trans. Mob. Comput. **8**, 1384–1396 (2009)
5. Hammond, J.L., Russell, H.B.: Properties of a transmission assignment algorithm for multiple-hop packet radio networks. IEEE Trans. Wireless Commun. **3**, 1048–1052 (2004)

6. Li, J.-S., Liu, I.-H., Liu, K.-X., Yu, S.-H.: Providing multiple-player online game service with an efficient multicast scheduling scheme in wireless ad hoc networks. In: Proceedings of MASS 2012, China (2012)
7. Li, J.-S., Liu, K.-H., Wu, C.-H.: Efficient group multicast node scheduling schemes in multi-hop wireless networks. Comput. Commun. **35**, 1247–1258 (2012)
8. Badia, L., Erta, A., Lenzini, L., Zorzi, M.: A general interference-aware framework for joint routing and link scheduling in wireless mesh networks. IEEE Netw. **22**, 32–38 (2008)
9. Joa-Ng, M., Lu, I.-T.: Spread spectrum medium access protocol with collision avoidance in mobile ad-hoc wireless network. In: Proceedings of IEEE INFOCOM 1999, USA (1999)
10. Djukic, P., Valaee, S.: Delay aware link scheduling for multi-hop TDMA wireless networks. IEEE/ACM Trans. Netw. **17**, 870–883 (2009)
11. Appani, P.K., Hammond, J.L., Noneaker, D.L., Russell, H.B.: An adaptive transmission-scheduling protocol for mobile ad hoc networks. Ad Hoc Netw. **5**, 254–271 (2007)
12. Ramaswami, R., Parhi, K.K.: Distributed scheduling of broadcasts in a radio network. In: Proceedings of INFOCOM 1989, Canada (1989)
13. Ramanathan, S., Lloyd, E.L.: Scheduling algorithms for multihop radio networks. IEEE/ACM Trans. Netw. **1**, 211–222 (1993)
14. Hikmet, M., Roop, P., Ranjitkar, P.: Fairness-based measures for safety-critical vehicular ad-hoc networks. In: Proceedings of IEEE 18th International Symposium on Real-Time Distributed Computing 2015, Auckland, New Zealand (2015)
15. Yu, H., He, Z., Niu, K.: STDMA for vehicle-to-vehicle communication in a highway scenario. In: Proceedings of MAPE 2013, Chengdu, China (2013)
16. Silva, L., Pedreiras, P., Alam, M., Ferreira, J.: STDMA-based scheduling algorithm for infrastructured vehicular networks. Stud. Syst. Decis. Control **52**, 81–105 (2016)