2005 ACM Symposium on Applied Computing

An Adaptive TDMA Slot Assignment Protocol in Ad Hoc Sensor Networks

Akimitsu Kanzaki
Dept. of Multimedia Eng.,
Grad. Sch. of Information
Science and Technology,
Osaka Univ.
1-5 Yamadaoka, Suita,
Osaka 565-0871, Japan
kanzaki@ist.osaka-u.ac.jp

Takahiro Hara
Dept. of Multimedia Eng.,
Grad. Sch. of Information
Science and Technology,
Osaka Univ.
1-5 Yamadaoka, Suita,
Osaka 565-0871, Japan
hara@ist.osaka-u.ac.jp

Shojiro Nishio
Dept. of Multimedia Eng.,
Grad. Sch. of Information
Science and Technology,
Osaka Univ.
1-5 Yamadaoka, Suita,
Osaka 565-0871, Japan
nishio@ist.osaka-u.ac.jp

ABSTRACT

Due to its ability to provide collision-free packet transmission regardless of the traffic load, TDMA (Time Division Multiple Access) has been applied in ad hoc sensor networks. We have previously proposed a TDMA slot assignment protocol that utilizes the channel bandwidth effectively in ad hoc sensor networks. However, in this protocol, the channel bandwidth cannot be fully utilized because the frame length across the whole network tends to increase. In this paper, we extend this protocol to further improve channel utilization. The extended protocol prevents the excessive increase of unassigned slots by minimizing each node's frame length. Furthermore, we verify the effectiveness of our proposed protocols by simulation experiments. The results show that our extended protocol improves channel utilization dramatically in comparison with our previous protocol.

Categories and Subject Descriptors

C.2.2 [**Network Protocols**]

Keywords

ad hoc sensor networks, TDMA, slot assignment

1. INTRODUCTION

Recent advances in hardware and wireless communication technologies have led to an increasing interest in ad hoc networks that are temporarily constructed by only mobile hosts. Especially, ad hoc sensor networks have emerged as an important application to monitor physical and social environments. In ad hoc sensor networks, each node continuously monitors the ambient environment and communicates with other neighboring nodes. This results very heavy traffic load in the whole network. In order to handle such situation, TDMA (Time Division Multiple Access) is well suited for the solution of the traffic because of its ability to provide collision-free packet transmission regardless of the traffic load. Until now, many transmission scheduling protocols using TDMA applied to ad hoc networks have been proposed. However, most of them (e.g., [2, 3]) do not consider autonomous behaviors of mobile nodes. Thus they cannot update the slot assignment of each node due to arrival or exit of node. Although there are some conventional protocols[1, 5, 6] that consider the dynamic change of network topology, they show poor channel utlization because of the existence of many conflicting or unassigned slots.

In [4], we have proposed ASAP (Adaptive Slot Assignment Protocol), which is a TDMA slot assignment protocol to improve the channel utilization by considering the autonomous behavior of nodes. However, under ASAP, since the frame length in the whole network tends to increase, the channel bandwidth is not fully utilized.

In this paper, we propose E-ASAP (Extended ASAP) as an extention of ASAP. E-ASAP enables each node to minimize its frame length by adding more detailed information to an ASAP control packet. As a result, E-ASAP drastically improves the channel utilization.

The remainder of the paper is organized as follows. In section 2, we briefly explain ASAP. In section 3, we explain E-ASAP proposed in this paper. In section 4, we show the results of simulation experiments, and conclude this paper in section 5.

2. ASAP (ADAPTIVE SLOT ASSIGNMENT PROTOCOL)

In this section, we briefly explain ASAP[4]. ASAP sets the frame length for a *new node* based on frame lengths of nodes in its *contention area* and minimizes the number of unassigned slots to improve the channel utilization. Here, a new node is defined as a node that newly joins the network, and the contention area is defined for each node as the set of nodes that can cause collisions between sent packets, i.e., nodes less than two hops from the node.

2.1 Frame Format

Fig. 1 shows the TDMA format in ASAP. The first slot in a frame is reserved for a new node to transmit control packets for requesting slot assignment information. Thus, no data packets are transmitted in this slot. The frame length of each node in ASAP is set as a power of two and can be changed dynamically depending on the slot assignment in its contention area. By setting the frame length as a power of two, packet collisions can be avoided between nodes with

SAC'05 March 13–17, 2005, Santa Fe, New Mexico, USA.

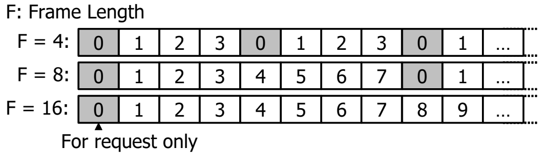


Figure 1: TDMA format in ASAP.

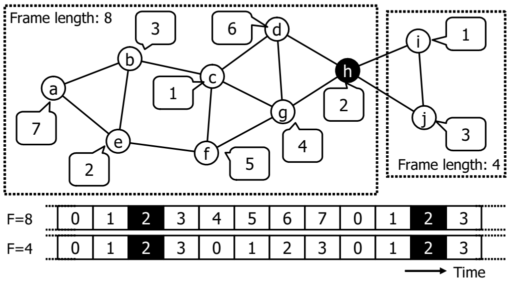


Figure 2: Consistency of slot assignment between nodes with different frame lengths.

different frame lengths. Fig. 2 shows an example of an ad hoc network formed by ten nodes. Here, the number shown in each balloon represents the slot assigned to a node in its frame length. Node h, which sets its own frame length as eight slots and connects to nodes i and j whose frame lengths are four slots, can transmit packets in slot 2 without collisions. This is because the frame of four slots can be easily translated to the frame of eight slots by doubling the frame and copying the four slots to the latter part.

2.2 Data Format

Each node maintains the following information: (a) the frame length and slot assignment of itself, (b) the slot assignment of its neighbors, and (c) the slot assignment of its hidden nodes (nodes of two hops away). Fig. 3 shows the information held by node h in Fig. 2. In Fig. 3, a number and a letter in the assignment information denote the slot number and the node to which the slot is assigned, respectively. Here, the information on its neighbors and hidden nodes are generated by recording their assignment information that each of them transmits in its assigned slot. In addition, the information of a node with shorter frame length is copied repeatedly every frame length in the assignment information. For example, in Fig. 3, node h copies the assigned slot of node i (slot 1) to slot 5 every 4 slots (the frame length of node i) in its information.

2.3 Packet Format

Each node has two modes, the *transfer mode* and the *control mode*, and behaves differently in different modes. In the transfer mode, nodes send *Data packets (DATs)*. A DAT contains the information on the frame length of the sender, the current slot number, and the maximum frame length among the sender and its neighbors. Of course, data is also contained in the DAT. On the other hand, in the control mode, nodes mainly send *Information packets (INFs)*. An INF contains the assignment information on the sender and its neighbors that is held by the sender.

2.4 Slot Assignment

To obtain the slot assignment information in the contention area, a new node collects INFs transmitted by its neighbors and sets its frame length as the maximum frame length among all nodes in its contention area. Then, the new node selects a slot assigned to itself by the following three procedures:

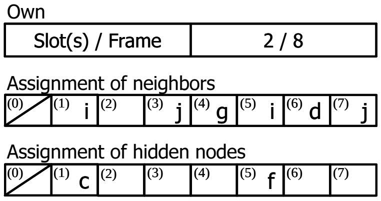


Figure 3: Information held by node h in Fig. 2.

1. *Getting an unassigned slot (GU)*
   If some unassigned slots are found in the slot assignment information, the new node assigns one of them to itself.
2. *Releasing multiple assigned slots (RMA)*
   If no unassigned slot is found, the new node checks whether a node in the contention area is assigned multiple slots. If such a node is found, the new node releases one of these slots from its assignment information and assigns it to itself. If multiple slots are assigned to more than one node, the node with the largest number of assigned slots is chosen to release a slot.
3. *Doubling the frame (DF)*
   If no unassigned slot is found and no node has multiple assigned slots that can be assigned to the new node, the new node doubles the frame length of the slot assignment information and copies the assignment information to both the former half and the latter half of the doubled frame. As described above, the first slot in the frame is not assigned to any nodes; therefore, after doubling the frame length, the first slot in the latter half becomes an unassigned slot. The new node assigns this slot to itself.

2.5 Detection of Conflict and its Solution

In ASAP, a conflict of slot assignment occurs when a new node connects to equal or more than two nodes to which the same slots are assigned. When a new node detects the conflict, it firstly converts the slot assignment information of all nodes causing the conflict into that of the maximum frame length among those nodes. After that, the new node solves the conflict by the following three procedures:

1. *Deleting a conflicting slot*
   If there are some non-conflicting slots assigned to nodes causing the conflict, the conflicting slot is released from all of the nodes except the one with the smallest number of assigned slots. For example, in Fig. 4, since there are non-conflicting slots 7, 11, and 15, which are assigned to nodes a and b, conflicting slot 3 is released from node a, which is assigned more slots than node b.
2. *Dividing the assignment*
   If there are multiple conflicting slots assigned to nodes causing the conflict, these slots are divided up between the nodes. For example, Fig. 5 shows that conflicting slots 3 and 11 are divided between nodes a and b.
3. *Doubling the frame and dividing the assignment*
   If a conflict occurs among nodes to which only one slot is assigned, the conflict cannot be solved with the current frame length. In this case, the frame length of these nodes is doubled and multiple slots which are conflicting in the doubled frame are divided up between the nodes. If the conflict still cannot be solved

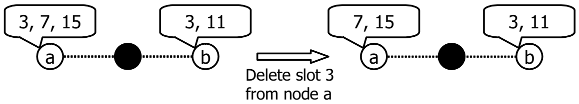


Figure 4: Deleting a conflicting slot.

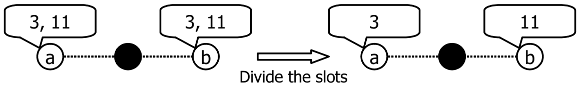


Figure 5: Dividing the assignment.

with the doubled frame length, the new node repeats the doubling of the frame and dividing of the slot assignment. For example, in Fig. 6, when slot 3 is conflicting between nodes a and b in the four-slot frame, the new node doubles the frame length (i.e., eight). After that, the new node divides up slots 3 and 7 between nodes a and b.

After solving the conflict, the new node selects a slot assigned to itself by the procedure described in section 2.4 and transmits the updated information to all its neighbors. The neighbors causing the conflict reconfigure their own assignment based on the updated information received from the new node.

However, in this situation, the new node may fail to collect the INFs due to their collisions. If such collisions occur, the new node sends an *Error detection packet (ERR)*, which consists of the information when the new node has detected collisions, to its all neighbors. The neighbors recerving the ERR recognize that they have caused collisions, and retransmit the INFs after waiting for randomly determined frames. This operation is repeated until the new node completes the collection of INFs from its all neighbors.

2.6 Releasing Slot Assignment

When a node exits from the network, it releases slots assigned to itself and stops transmitting DATs. Since a DAT contains information on the sender's frame length, neighbor nodes of the exited node can detect the exit when no packet from the exited node has been received during the time of the exited node's frame length. Then, they release the slots assigned to the exited node from their slot assignment information. In addition, they release the slots assigned to nodes that have left their contention areas due to the exit. After reconfiguring the slot assignment, the neighbors of the exited node send the updated information to their neighbors. The nodes that have received this information update the slot assignment information by releasing the slots assigned to the exited node. In this way, the operation of releasing slots assigned to the exited node is completed.

Here, if one of the following conditions is satisfied for each neighbor of the exited node, it halves its frame length and sends the updated information to all nodes in its contention area.

1. The frame lengths of all neighbors are smaller than that of itself.
2. The first slot in the latter half of the frame is an unassigned slot, and one of the following conditions is satisfied:
    - The slot assignment information of the former half and the latter half are completely the same.
    - For all slots in the former half of the frame and the corresponding slots in the latter half, a slot assigned to a particular node is also assigned to the same node in the other half or is not assigned to any node.

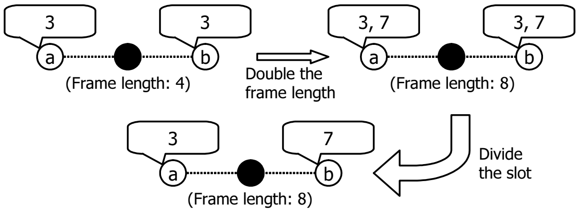


Figure 6: Doubling the frame and dividing the assignment.

Own

Slot(s) / Frame
2 / 8

Neighbors

Node	Slot(s) / Frame
d	6 / 8
g	4 / 8
i	1 / 4
j	3 / 4

Hidden nodes

Node	Slot(s) / Frame
c	1 / 8
f	5 / 8

Figure 7: Information held by node h in E-ASAP.

3. E-ASAP (EXTENDED ASAP)

In ASAP, since a new node decides the slot assignment for itself considering the slot assignment of nodes in its contention area, the channel utilization improves compared with the conventional protocols. However, the frame length in the whole network tends to increase because the frame length of a new node is set as the maximum frame length among nodes in its contention area. Moreover, each node has few opportunities to minify its frame length when a node exits from the network. In this section, we propose E-ASAP (Extended ASAP) as a slot assignment protocol, in which the frame length of each node is not affected by those of nodes in its contention area. In E-ASAP, each node assigns a slot to itself in the frame of the minimum length where conflicts do not occur. Thus, the channel utilization improves in comparison with ASAP.

3.1 Frame, Data, and Packet Format

Similar to ASAP, the frame length in E-ASAP is set as a power of two slots and the first slot in the frame is reserved for a new node to transmit control packets for requesting slot assignment information.

However, in E-ASAP, the information maintained by each node is different from that in ASAP. Fig. 7 shows the information held by node h in Fig. 2. As shown in Fig. 7, each node additionally maintains the frame lengths of the other nodes in its contention area.

In E-ASAP, each node behaves differently in the two modes: the transfer mode and the control mode. A DAT transmitted by a node in the transfer mode is the same as that in ASAP. However, an INF additionally contains the information on the frame length of the sender's neighbors as the information maintained by each node.

3.2 Slot Assignment

Similar to ASAP, a new node collects INFs transmitted by its neighbors to obtain slot assignment information in the contention area. Then, the new node firstly sets its frame length as four slots which is the minimum frame length in E-ASAP. After that, the new node searches a slot which it can assign to itself via the following three procedures:

1. *Getting an unassigned slot (GU)*
   If the first slot is not assigned to any neighbors, and some unassigned slots are found in the slot assignment

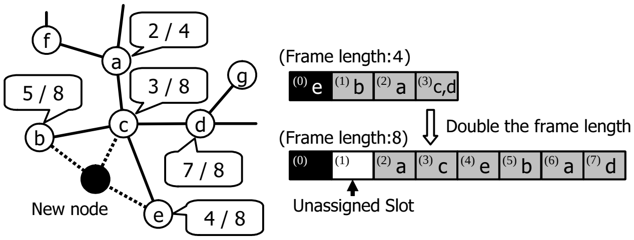


Figure 8: Assigning a slot in E-ASAP.

information, the new node assigns one of them, except for the first slot, to itself.

2. *Releasing multiple assigned slots (RMA)*
   If the first slot is not assigned to any neighbors and all the other slots are assigned to other nodes in its contention area, the new node checks whether one of the nodes is assigned multiple slots. If such a node is found, the new node releases one of these slots from its assignment information and assigns it to itself.
3. *Doubling the frame (DF)*
   If no slot is available in the current frame length, the new node doubles the frame and tries again to assign a slot by GU and RMA. This procedure is repeated until the new node finds a slot to assign.

Fig. 8 shows an example of the above procedures. In the right side of Fig. 8, black, white, and gray rectangles denote the first slot, unassigned slots, and slots assigned to other nodes in the contention area, respectively. In the left side of Fig. 8, the right and left numbers in each balloon respectively denote the frame length and the slot assigned to the node. First, the new node tries to assign a slot to itself in the minimum frame length of four slots. However, the new node cannot assign any slots to itself since the first slot is assigned to neighbor node *e*. Thus, the new node doubles the frame length and tries to assign a slot in the frame length of eight slots. In this frame, the first slot is unassigned, and slot 1 is not assigned to any nodes. The new node assigns slot 1 to itself by GU.

After selecting a slot assigned to itself, the new node makes all nodes in its contention area update their slot assignment information by sending the updated information which consists of the whole information held by the new node. Moreover, if the new node selects a slot assigned to itself by RMA, the node which gives the slot to the new node sends the updated information to make all nodes in its contention area update their slot assignment information.

3.3 Detection of Conflict and its Solution

Similar to ASAP, a conflict of slot assignment occurs when a new node connects to multiple nodes with the same assigned slots. A new node can detect the conflict by collecting the INFs from its neighbors. In E-ASAP, such conflicts are solved by the same way as in ASAP.

3.4 Releasing Slot Assignment

Similar to ASAP, when a node exits from the network, it releases slots assigned to itself, and its neighbors detect the exit. After reconfiguring the slot assignment information, the neighbors of the exiting node temporarily release their own assigned slots and search for slots that can be assigned to themselves by GU, described in section 3.2. Then, if a neighbor finds a slot which it can assign to itself in the shorter frame than the current one, it updates its own assignment. Finally, each neighbor that updates its own assignment transmits the updated information to all other nodes in its contention area.

3.5 Discussion

In E-ASAP, each node can minimize its frame length due to the following three features.

1. Adding more information to control packets
   In E-ASAP, because each node additionally maintains the information on the frame length of other nodes in its contention area, the new node can determine the slot assigned to itself in more flexible way than that in ASAP. Therefore, the excessive increase of the frame length can be avoided.
2. Expanding the propagation range of control packets
   In ASAP, when the new node selects a slot assigned to itself, the updated information is propagated only to the contention area of the new node. Thus, nodes outside of the contention area of the new node may have incorrect assignment information. This may cause the deterioration of the channel utilization.
   On the other hand, in E-ASAP, because the node of which the slot assignment is changed by RMA sends the updated information to its contention area, all nodes can have correct assignment information. However, because the propagation range is expanded to 4 hop from the new node at maximum, the control traffic may become larger than that in ASAP.
3. Minifying the frame length when releasing slot assignment
   In E-ASAP, since each node maintains the information on the frame length of other nodes in its contention area, each node can have much opportunities to minify their frame length when its neighbor exits from the network. Therefore, the frame length in the whole network can be minimized.

4. PERFORMANCE EVALUATION

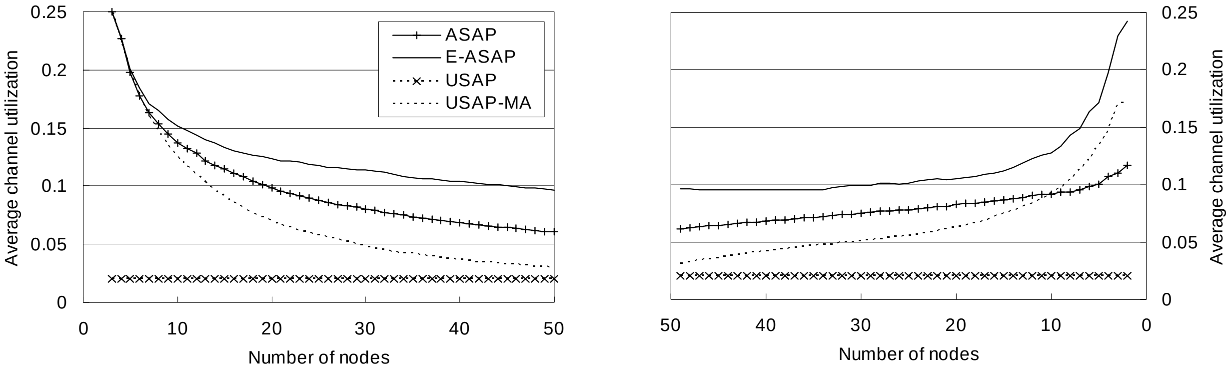
In this section, we show simulation results regarding performance evaluation of our proposed protocol. In the simulation experiments, we compare the performance of our proposed protocol with ASAP, USAP[5], and USAP-MA[6].

In USAP, the frame length is fixed, and a node assigns an unassigned slot to itself. In USAP-MA, the frame length is set as a power of 2, and can be changed dynamically. Similar to USAP, a node assigns an unassigned slot in the frame to itself. If there is no unassigned slot in the frame, a node creates unassigned slots by enlarging the frame length, and assigns one of them to itself.

4.1 Simulation Environment

Since our protocols target ad hoc sensor networks, we assume that each node in the network does not move. A new node appears at a random position within the area where the new node can connect to at least one node. Here, the time elapsed to reconfigure the assignment information of each node is much shorter than the average interval of network topology changes. Thus, it is assumed that after the new node has appeared or the node has exited, no other node appears or exits from the network until all nodes reconfigure the assignment information and transit back to the transfer mode. There are initially two nodes, and this number increases one by one until fifty nodes are present in the network. The number of nodes then decreases one by one in the reverse order.

In both ASAP and E-ASAP, the new node listens to the network channel for the time of 128 slots for collecting the



(a) When the number of nodes increases (b) When the number of nodes decreases

Figure 9: Average channel utilization.

slot assignment information of nodes in its contention area. When multiple slots are available for assignment to the new node, the new node assigns the slot with the smallest slot number among them to itself. For USAP, we set the frame length as 50 slots in order to assure that all nodes can be assigned slots. A new node assigns the unassigned slot with the smallest slot number among all unassigned slots.

In [6], the authors did not define neither when and how to change the frame length nor how to select a slot assigned to a new node in USAP-MA. Thus, we define these behaviors in the simplest way. A new node sets its frame length as the maximum frame length among all neighbors and assigns the unassigned slot with the smallest slot number among all unassigned slots. If there is no unassigned slot in the frame, the new node doubles its frame length and assigns one of the unassigned slots generated in the latter half of the doubled frame. When a node exits from the network, neighbors of the exited node release the slots assigned to the exited node. When all slots in the latter half of the frame are not assigned to any nodes in the contention area, the neighbor halves its frame length.

4.2 Evaluation Criteria

In the simulation experiments, we evaluate the following three measures:

1. Average channel utilization
   We define the fraction of the number of assigned slots to the frame length as the *channel utilization*. In addition, we define the average of the channel utilizations of all nodes across the whole network as the *average channel utilization*. The higher the average channel utilization, the more frequently nodes can transmit DATs, and thus, the more efficiently the channel bandwidth is utilized.
2. Control traffic
   In ASAP and E-ASAP, every packet (DAT and INF) contains information on the node ID, the frame length, and the slot numbers. For simplicity, we assume that the data size of each information is 1 byte and define the total amount of transmitted packets when joining and exiting of nodes, except for DATs, as the *control traffic*. Since the authors of [5] and [6] did not define the detailed behavior to exchange control packets, we do not evaluate control traffics in USAP and USAP-MA.
3. Setting delay
   We define the average number of required slots after a new node joins or a node exits until all nodes transit back to transfer mode as the *setting delay*. Due to the same reason of control traffic, we do not evaluate setting delays in USAP and USAP-MA.

4.3 Simulation Results

Fig. 9, 10, and 11 show the simulation results. The horizontal axis on all graphs indicates the number of nodes in the network. The vertical axis indicates the average channel utilization in Fig. 9, the control traffic in Fig. 10, and the setting delay in Fig. 11. In addition, graphs (a) and (b) indicate the results when the number of nodes increases and decreases, respectively.

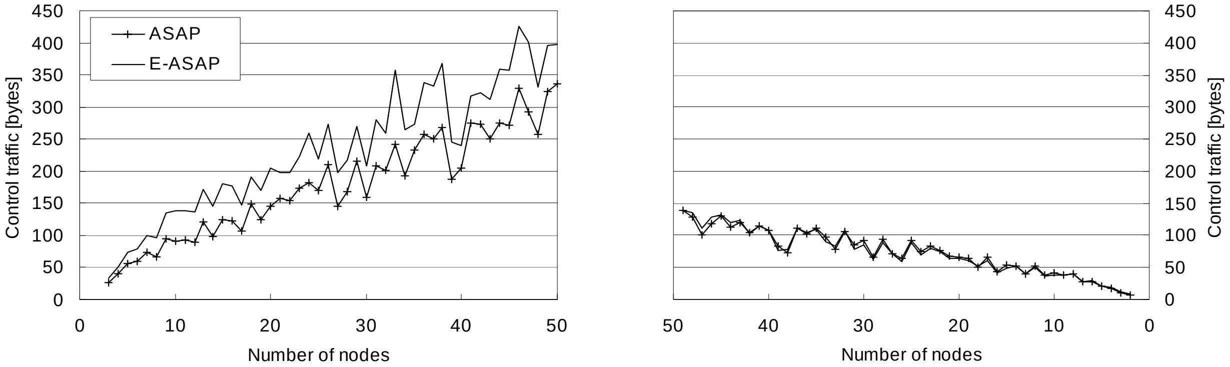
4.3.1 Average Channel Utilization

From Fig. 9, the average channel utilization in USAP is very low because the fixed frame length causes many unassigned slos. On the other hand, in USAP-MA, ASAP, and E-ASAP, which changes the frame length of nodes dynamically, the average channel utilizations are always higher than that in USAP. Especially, the average channel utilization in E-ASAP is much higher than that in other protocols. Furthermore, in E-ASAP, the channel utilization nearly recovers to the initial value (0.25) when the number of nodes decreases to 2. From this result, the channel utilization in E-ASAP is expected to remain high even when the number of nodes dynamically changes, i.e., nodes randomly appear and disapper from the network. While we do not show the details due to limited space, we have confirmed that E-ASAP archives the high channel utilization in such a situation by other simulation experiments.

4.3.2 Control Traffic

Fig. 10 shows that when the number of nodes increases, the control traffic caused by E-ASAP is approximately 10% more than caused by ASAP. This is due to the increase of information in control packets and the expansion of their propagation range. This result shows that there is a tradeoff between the average channel utilization and the control traffic. The maximum difference in control traffic between two protocols is about 100 bytes. Considering the available channel bandwidth, this difference may be small enough to be disregarded. However, when the node density is very high, the frame length of each node becomes very long, and the amount of information sent by each node becomes large. In such a situation, the difference in control traffic between two protocols increases significantly, and may affect the setting delay and the power consumption of each node.

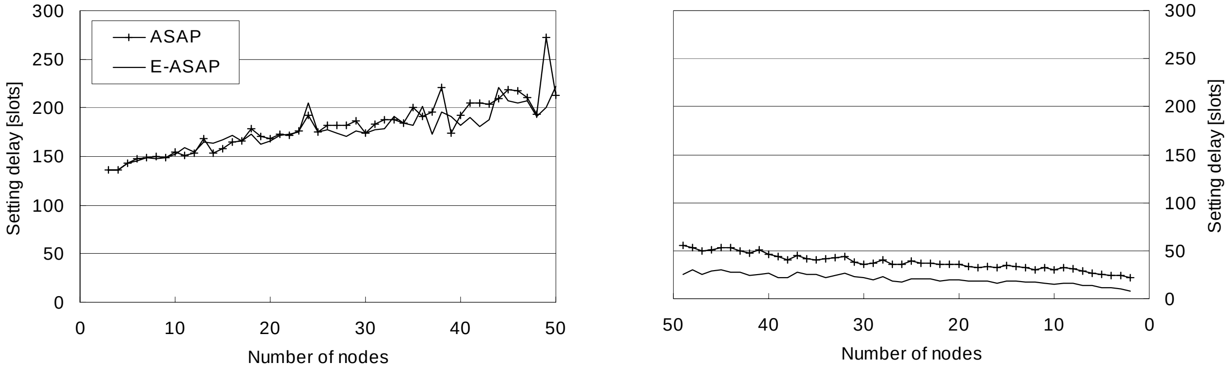
When the number of nodes decreases, the control traffics in the two protocols are almost identical. This is because that the propagation ranges of control packets are the same



(a) When the number of nodes increases

(b) When the number of nodes decreases

Figure 10: Control traffic.



(a) When the number of nodes increases

(b) When the number of nodes decreases

Figure 11: Setting delay.

in both protocols.

4.3.3 Setting Delay

Fig. 11 shows that when the number of nodes increases, the setting delays in both protocols are almost identical. On the other hand, when the number of nodes decreases, the setting delay in E-ASAP is shortened by almost half of that in ASAP. This occurs because each node has many opportunities to transmit packets under high channel utilization in E-ASAP. The difference in setting delays between two protocols is about 20 slots. However, when the node density is very high, the frame length of each node increases greatly. In such a situation, the difference in setting delays between the two protocols may become large and cannot be disregarded.

5. CONCLUSION

In this paper, we have proposed E-ASAP, which is a TDMA slot assignment protocol to improve the channel utilization. E-ASAP decreases the number of unassigned slots by adding more detailed information to control packets and expanding the propagation range of the control packets.

Our protocol can effectively assign slots to nodes when a node appears and dissapears from the network, but it cannot accommodate the movement of nodes since it targets ad hoc sensor networks. As part of our future work, we plan to consider a frame format to accommodate the movement of nodes. We will also extend our protocol to have a facility of adapting the change of network environment such as traffic load.

Acknowledgement

This research was partially supported by The 21st Century Center of Excellence Program "New Information Technologies for Building a Networked Symbiotic Environment" and Grant-in-Aid for Young Scientists (A)(1668005) of the Ministry of Education, Culture, Sports, Science and Technology, Japan, and by Tateishi Science and Technology Foundation and JSPS Research Fellowship for Young Scientists.

6. REFERENCES

[1] I. Chlamtac, A. Farago, and H. Zhang: "Time-spread multiple-access (TSMA) protocols for multihop mobile radio networks," *IEEE/ACM transactions on networking*, vol. 5, no. 6, pp. 804-812 (Dec. 1997).

[2] C. Florens and R. McEliece: "Scheduling algorithms for wireless ad-hoc sensor networks," in *Proc. IEEE Global Telecommunications Conference (GLOBECOM '02)*, vol. 1, pp. 6-10 (Nov. 2002).

[3] H. Lee, J. Yeo, S. Kim, and S. Lee: "Time slot assignment to minimize delay in ad-hoc networks," *IST Mobile Communications Summit 2001* (Sept. 2001).

[4] A. Kanzaki, T. Uemukai, T. Hara, and S. Nishio: "Dynamic TDMA slot assignment for ad hoc networks," in *Proc. International Conference on Advanced Information Networking and Appications (AINA 2003)*, pp 330-339 (Mar. 2003).

[5] C. D. Young: "USAP: a unifying dynamic distributed multichannel TDMA slot assignment protocol," in *Proc. IEEE MILCOM '96*, vol. 1, pp. 235-239 (Oct. 1996).

[6] C.D. Young: "USAP multiple access: dynamic resource allocation for mobile multihop multichannel wireless networking," in *Proc. IEEE MILCOM '99*, vol. 1, pp. 271-275 (Nov. 1999).