

Research Article

Design and Implementation of a Mobile Multihop WSN for AGV System

Jiaqi Zhang, Molin Jia, Noriyoshi Yamauchi, and Takaaki Baba

Graduate School of Information, Production and Systems, Waseda University, 2-7 Hibikino, Wakamatu-ku, Kitakyushu-shi, Fukuoka 808-0135, Japan

Correspondence should be addressed to Jiaqi Zhang, jackyzhang@fuji.waseda.jp

Received 4 November 2010; Accepted 10 November 2010

Academic Editor: Dimitrios Katsaros

Copyright © 2010 Jiaqi Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A system data sharing protocol of mobile WSN named synchronous dynamic multihop data sharing protocol (S-DMDS) is presented for automated guided vehicle (AGV) system. It is a cross-layer protocol designed from route layer to MAC layer. By adopting a concept of system data sharing, it is possible to make each node exchange the data timely with all the other nodes. It is also a topology-agnostic protocol which has no knowledge of neighbors, routes, or next hops. From the results of the 16-nodes simulation, S-DMDS protocol is proved to be efficient exchange data timely between the devices of AGV system in mobile multihop situation. Moreover, it also shows that S-DMDS significantly outperforms NST-AODV with investing about 41.6% system sharing delay as well as 80% RAM consumption. At last, 5-node experiment indicates that S-DMDS can work well in real environment.

1. Introduction

In recent years, wireless sensor networks (WSNs) technology becomes an emerging field in a wide range of applications such as industrial environment [1]. In addition, with the development of AGV system, the application combining WSN technology with AGV system becomes more widespread. Benefits of wireless technologies are fairly obvious: WSN facilitates installation and maintenance of AGV system and eliminates expensive cables and save costs; in addition, AGV system can easily be reconfigured. One potential dominant technology, which seems to be really effective for the industrial application, is the IEEE802.11 standard [2–5]. Its main drawback still remains in cost, particularly if simple WSNs are considered. Other cheaper technologies such as IEEE802.15.4 [6] and Bluetooth [7] (IEEE802.15.1) are already important actors in the market. However, IEEE802.15.4 does not have a good performance when it is applied to a multi-hop mesh network, although it supplies for such a function. And the serious beacon collision occurs in the beacon mode and the hidden node problem still exists in no-beacon mode. Moreover, both IEEE802.15.4 and Bluetooth need at least one coordinate node to manager

the whole network. At the same time, solutions based on the work of the IEEE 802.15.4 group, like WirelessHART and ISA SP100, have been presented, but devoted to process control [8]. They are able to manage very large mesh network but overall “information rate” is low within 1 node/s or less.

Furthermore, routing in WSNs is an also very challenging task due to the inherent characteristics that distinguish these networks from other wireless networks like mobile ad hoc networks or cellular networks. Most of the network architectures assume that sensor nodes are stationary. However, some devices may change their position rapidly in AGV system leading to the dynamic network topology. Routing messages from or to moving nodes is more challenging since route stability becomes an important issue.

Some mobile route algorithms proposed by MANET [9, 10] can adapt dynamic topology. Ad hoc On-demand Distance Vector routing (AODV) is one of the most studied routing protocol among them. AODV is a reactive routing protocol. When a node requires a route, it initiates a route discovery procedure broadcasting Route Request (RREQ) messages. When a node receives a RREQ, if either it has a valid route entry to the demanded destination or it is the destination itself, it creates and sends a Route Reply

(RREP) message back to the originator node. Every node maintains route entries with forward and backward next hop information that expires after a specified time if the path becomes inactive (i.e., it is not used for data transmission). However, full version of AODV is mainly proposed for implementation of laptop or PDA platform. It costs so much hardware resources that simple WSN node cannot supply. Therefore some simple version of AODV [11–15] is proposed for low cost application. Not So Tiny-AODV (NST-AODV) [16] is a typical one. It is implemented into MICAz [17] platform which use a 2.4 GHz IEEE 802.15.4 radio interface with a 250 kbps data rates. Although it has been proved to be efficient for point to point transmission, it may not be so suitable for AGV system. Since AGV system requires each device should obtain the information of all the other devices in time. It needs a mobile multi-hop data sharing WSN shown as Figure 1, which is not a conventional transmission style that is from source node to destination node.

S-DMDS protocol is proposed to provide multi-hop data sharing communication between mobile AGV nodes with reasonable low cost. It adopts a data sharing concept that each device can get an interface to communicate with the whole system. Via this interface, any node can get the system state in time.

This paper is structured as follow. In Section 2, AGV system scenario is introduced. Then, the protocol principle including software structure, data update processing and time division multiple access (TDMA) mechanism is described in Section 3. After that, the implement hardware (WHM-4) is introduced in Section 4. A series of performance evaluations are provided in Section 5. Finally, conclusion of the paper is made in Section 6.

2. AGV System Scenario

Recently, AGV system is widely used in many automatic systems. Due to the increasing demand for mobility and capability of work in harsh environment, the use of wireless sensor network (WSN) is gaining its importance.

In the scenario shown as Figure 2, establishing a mobile multi-hop wireless sensor network, we plan to control the AGV vehicle to carry cargos and register cargo information automatically in order to achieve unattended management in a storehouse. In the system, there are several AGV vehicles used for carrying cargos; some ground equipments for stacking goods; a charger is used for the AGV power supply and a monitor is also provided to let manager watch and control the whole system. The total number of devices in the AGV system is 16-node.

All the devices in the AGV system should detect their own state (such as location information of AGV and devices on the ground) through sensors and share their state or command information to the others via wireless network. For example, carrying command from devices on the ground should be delivered to the suitable AGV to execute. And traffic control must be executed by AGV itself intelligently in the case of intersection, dead-end road, and so forth.

For the wireless sensor network, the conditions and features of AGV system scenario are summarized as follow.

- (1) considering AGV vehicles are moveable, the topology of the network changes rapidly;
- (2) the real time system data sharing is necessary;
- (3) data transmission between every two nodes of the network may be multi-hop;
- (4) high network stability is required;
- (5) since power of wireless sensor module is supplied by devices of AGV system, the power consumption problem is not so important in this scenario.

S-DMDS protocol is designed for the AGV system. In the following section, S-DMDS is described in detail.

3. S-DMDS Protocol

S-DMDS is a protocol of system data sharing. To realize system data sharing, each node is arranged to broadcast a data sharing frame (DSF) regularly as shown in Figure 3 which is an interface between each node and the whole system. Since the DSF consists of every node's information in the network, each node can realize data exchanging with all the other nodes by reading and writing the DSF. Hence, mobile multi-hop network can be easily realized on simple wireless sensor nodes.

DSF is a data sharing media to collect the sharing information from all of the nodes. As shown in Figure 4, the structure of DSF consists of two main parts: head and data. Frame head is used for declaring the length and class of the frame, and it also includes sending node id. The purpose of the frame head is to realize a synchronous mechanism which integrates the main idea of [18, 19].

Since S-DMDS protocol employs TDMA as media access method to ensure each transmission to be successful. So it requires that once receiving DSF, each node needs to calculate next broadcast time slot by comparing the sending node id with its own id. Then the node can be synchronized again and again after receiving DSF.

The following equations describe the detailed relationship:

$$\text{If my id} > \text{sending id}$$

$$\begin{aligned} &\text{Broadcast time} \\ &= \text{current time} + (\text{my id} - \text{sending id}) \times \text{time slot}, \end{aligned} \quad (1)$$

$$\text{If my id} < \text{sending id}$$

$$\begin{aligned} &\text{Broadcast time} \\ &= \text{current time} + (16 + \text{my id} - \text{sending id}) \times \text{time slot}. \end{aligned} \quad (2)$$

Data part of DSF is composed of all nodes' sharing data. Besides, each node's information also consists of two parts, one is the sequence number, and the other is the sharing information. The function of sequence number will be explained in the next section.

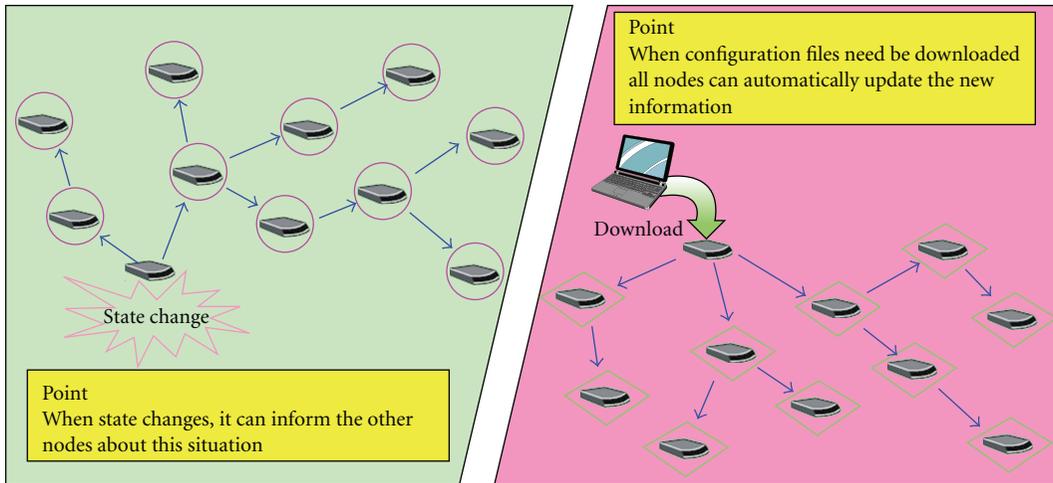


FIGURE 1: A multi-hop data sharing WSN.

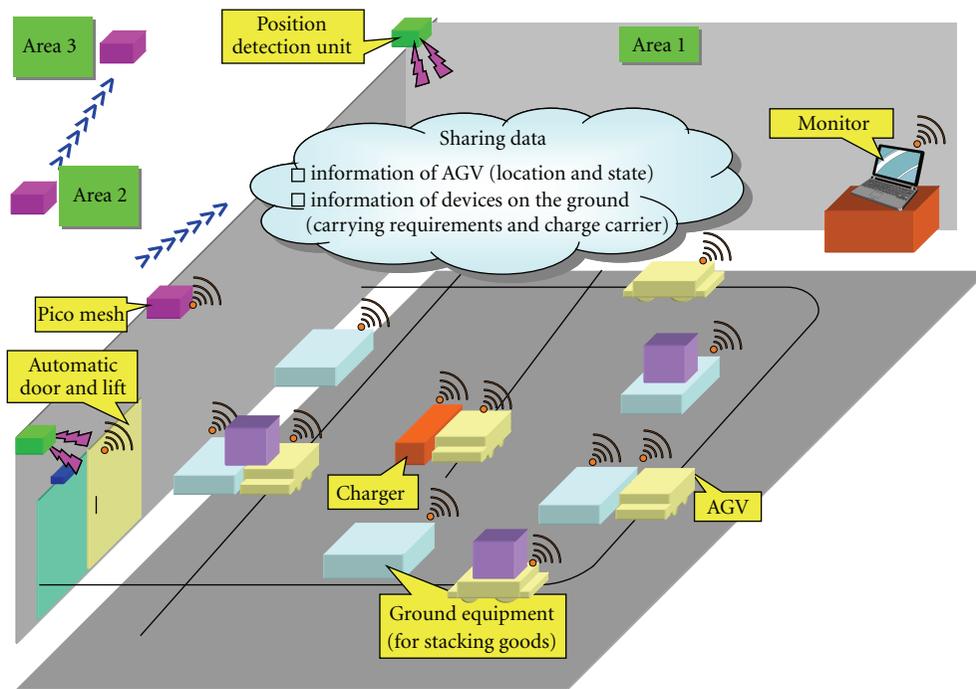


FIGURE 2: AGV system scenario.

When the network is working, every nodes broadcast data sharing frame (DSF) regularly in order to share their information to the whole network. When DSF is broadcasted in the network, any node receiving DSF can get other nodes information and store the information in its memory. Meanwhile, each node can also update its own corresponding unit in DSF when some information needs to be transmitted. Then in the next transmission time, it can make the other nodes know its new information by broadcasting DSF from its memory.

The modules' software structure is shown in Figure 5. After stating, each node will keep on waiting for receiving DSF except when it turns to broadcast time. If it receives

a new DSF, it will employ a data update process to update the information into its memory. If a node reaches broadcast time, it will enter the synchronous adjustment process to calculate its own broadcast time before broadcast.

In the following two parts, data update process and node synchronous process are explained, respectively.

3.1. Data Sharing Update Process. This process works when a new DSF has been received as shown in Figure 6. The value of Sequence Number (Seq No) indicates whether the node information is new or not. And whenever any node updates the information of its corresponding unit

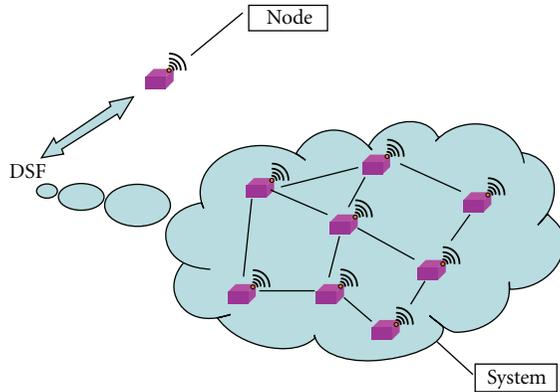


FIGURE 3: Data sharing mode.

in DSF, the corresponding Sequence Number will be increased by 1 to indicate that the information is the newer one.

Such purpose is to make sure that the receiving nodes can recognize whether the information of other nodes received is newer one by comparing Seq No.

3.2. Synchronous Mechanism. S-DMDS arrange each node to broadcast DSF at every broadcast cycle. In sending time sequence, broadcast cycle time has been divided into N time slots per broadcast cycle as shown in Figure 7. N is defined as the total number of the network. Each node could be allocated to its id's corresponding broadcast time slot to broadcast DSF after joining the sending time sequence. For example, node1 is arranged into 1st time slot to broadcast its DSF. Between every two broadcast time, it also adds a guard interval to make sure that previous node broadcast is finished before next node starts to broadcast. Besides, it can also make up the clock offsets problem.

However, it should be noticed that S-DMDS protocol allows only one time reference node to start broadcasting DSF at first. Otherwise, the synchronous system will be out of order. An example is given to explain the synchronous process in detail.

3.3. Synchronous Process. In the example, network topology is shown in Figure 8. Node1 is selected to be the time reference node. At first, the other nodes will keep on silence until receiving DSF. (Receiving DSF means the node joined the network group). After reference node1 starting to broadcast, node3, node4, and node15 can receive it and then calculate their broadcast time according to rule defined by (1) and (2). Because node3 has the nearest ID to node1 among them, it will broadcast DSF immediately. Thus, Node6 can receive the frame meanwhile it is also arranged to the sending temporal series. Following this rule, after several cycle all the nodes which can receive DSF are added to the system sending temporal series one by one. At last the synchronous process is over. Then every node can maintain the synchronous time table periodically once they receive DSF.

TABLE 1: Main parameters of WHM-4.

Main parameters of WHM-4	
Frequency	2405 MHz–2480 MHz
Bandwidth	75 MHz
Channel Step	5 MHz (16 channels maximum)
RF Baud Rate	250 kbps Maximum
Micro-controllers	TI-MSP430
Input	Digital Logic 4 Inputs
	Analog A/D 6 Inputs
Output	Open Collector Output 2 Outputs
I/F	RS232 3 pins
Power input	1.5 V/3 V External power supply
Board Size	38 mm × 55 mm × 20 mm

3.4. Prototype of S-DMDS Protocol. To show that by adopting S-DMDS how node can share their data to each other in one broadcast cycle time, a prototype is presented. As show in Figure 9, there are 4 nodes in this scenario. Node3 is a movable node as well as it cannot community with node1 directly.

At first stage, the scenario looks like as state 1, node1 is selected as a time reference node, and according to the sending temporal series, node1 will first start broadcasting, it sends DSF to its neighbor node2 and node4. Then node2 and node4 can get the information of node1.

After that, the scenario turns to state2, node2 starts to broadcast DSF which consists of the information of node1 and node2 to its neighbor node such as node1, node3 and node4. So node3 and node4 get the information of node1 and node2, meanwhile node1 and node4 can get the new information of node2. When the scenario turns to state3, node3's location changed. Therefore, Node3 ran out of node2's communication range, meanwhile, it ran into node4's communication range.

In this way, when Node3 broadcasts DSF, only node4 can receive the DSF which consists the information of node1, node2 and node3. At this time node4 has obtained all nodes' information. After node4 broadcasts DSF, all the nodes in this scenario can get the other nodes' information. Namely, the initial system data sharing is finished.

4. Implement Platform

A wireless sensor module named as WHM-4 developed by IPS of Waseda University is utilized for implementing S-DMDS protocol. The appearance of WHM-4 is shown in Figure 11. WHM-4 is a low-power consumption wireless module employing IEEE802.15.4 (2.4 GHz) radio interface, where every channel supports up to 250 kbps data rates. The power of RF module is 1.5 mA (3.3 V). The largest transmission range of WHM-4 is about 50 meters. The other main parameters of WHM-4 are shown in Table 1.

A TDMA schematic diagram is used to describe the AGV system with 16-node as shown in Figure 10. Then length of DSF (K) is defined as 340 bytes consisting of 320 bytes for storing the sharing data of 16-node and 20 bytes for control

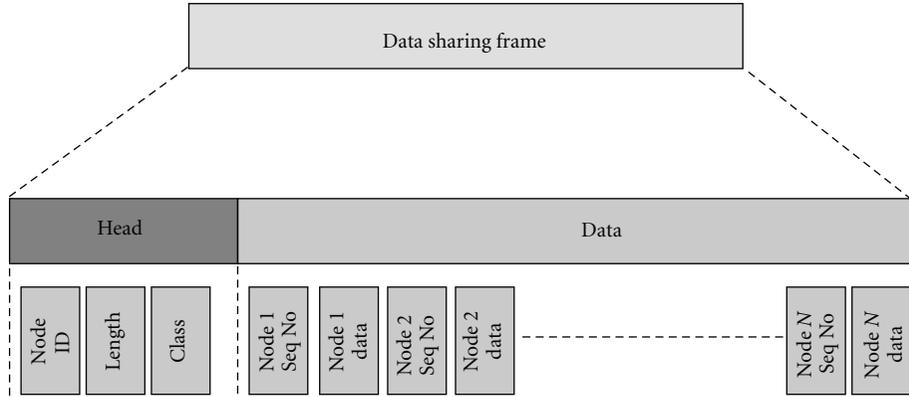


FIGURE 4: Structure of DSF.

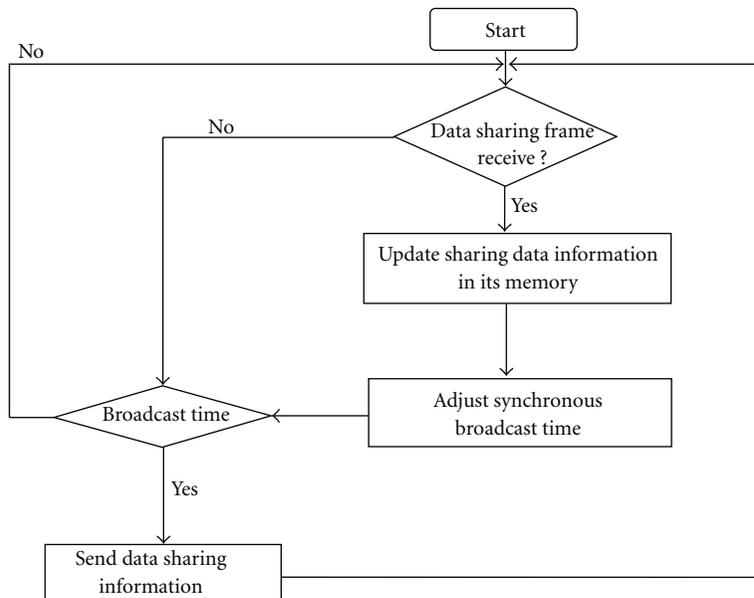


FIGURE 5: Flow chart of software structure in each WSN module.

information. Obviously, the length of each node sharing data is 20 bytes in DSF.

Therefore, each DSF transmission (Δt) takes about 10 ms and a guard interval (T_g) is kept into 10 ms for node to adjust broadcast time as well as avoiding clock offset. Each node broadcasts DSF every 320 ms (T_c) according to the sending sequence. Namely, T_c is the broadcast cycle.

5. Performance Evaluation

The evaluation scenarios are composed of 16 WHM-4 modules according to the requirement of AGV system. First, a throughput analysis is provided to evaluate the network capacity. And then a simulation of 16-node scenario is presented by using C++ simulation programs based on parameters of WHM-4 hardware. By doing such a simulation, the performance of S-DMDS in 16-node AGV system can be evaluated. At final part of this section, a five

nodes experiment is shown to prove S-DMDS protocol can work well in real environment.

5.1. Throughput Analysis. The throughput of the network is defined as how much data is successfully delivered in unit time. Because each node sent same size package regularly and by adopting TDMA that each transmission is successful without any collision as shown in Figure 10, the throughput of network is defined as

$$\text{Throughput} = \frac{N \times K}{T_c} \tag{3}$$

In S-DMDS algorithm, $T_c = N * (\Delta t + T_g)$, and we define K as length of DSF ($K = 340$ bytes). Hence the network throughput becomes

$$\text{Throughput} = \frac{K}{\Delta t + T_g} \tag{4}$$

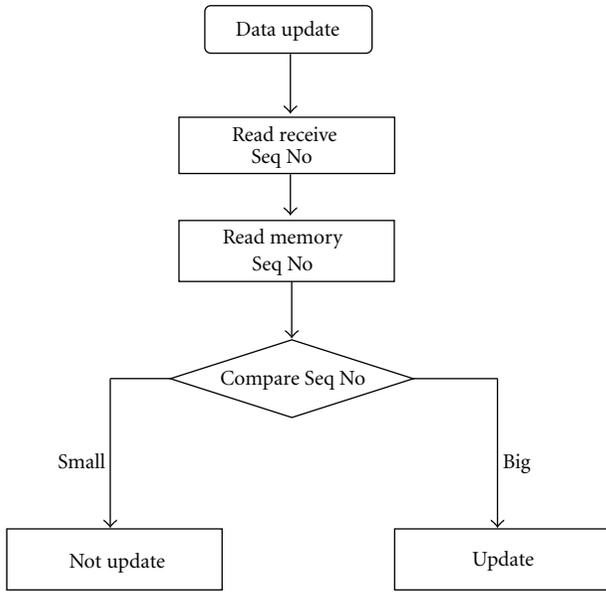


FIGURE 6: Data sharing update process flow chat.

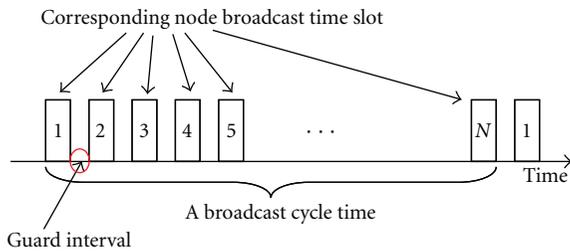


FIGURE 7: TDMA sending sequence.

Because Δt and T_g are determined by the speed of RF module and K is a constant, if the speed of RF module is fixed, the throughput becomes a constant value. In 16-node circumstance, a 2.4 G RF module of 125 kps transmission speed is adopted, so the network throughput is 136000 bits/sec.

5.2. Simulation of System Sharing Delay. To evaluate the system sharing delay in multi-hop situation, three different scale of network are established including the network of single-hop, 3-hop and 6-hop (6-hop is the largest network scale in 16-node circumstance). And a mobile situation is also included to indicate how S-DMDS performs.

Before doing the simulation, all nodes store only its own sharing data in its memory before broadcasting. System sharing delay (Tssd) means that how long all nodes can obtain system sharing data from all the other nodes. The network hops scale is assumed as minimum number of hops between two farthest nodes in network (e.g., from node1 to node 16, it takes 3-hops, so it means 3-hop networks in Figure 12).

5.2.1. Single-Hop Scale. For testing system sharing delay in single-hop network, node is allowed to communicate

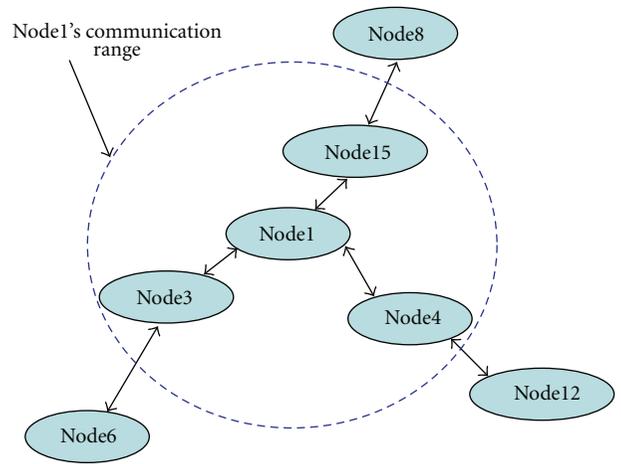


FIGURE 8: An example of synchronous process.

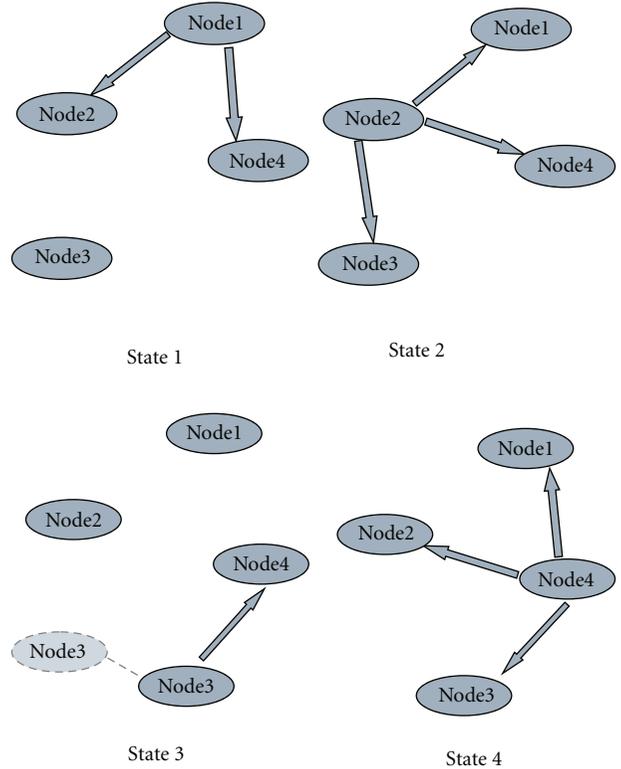
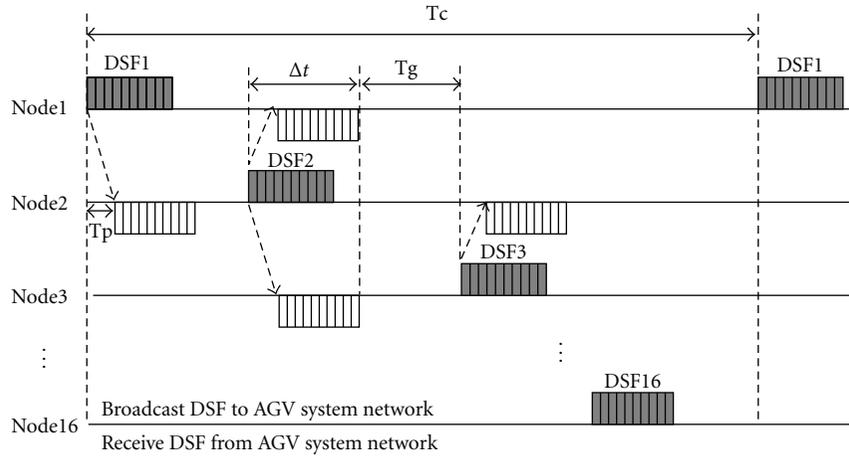


FIGURE 9: Data sharing process of the algorithm.

with all the others directly. And some mobile nodes are allowed to exchange their position to simulate mobile scenario as shown in Figure 13. The result shows that Tssd is about 320 ms (T_c) no matter whether nodes moved or not. In Figure 10, Tssd is the time at which the 16th nodes finish broadcasting DSE. Therefore, it can be summarized that in single-hop network, all nodes can share data to each other only within one T_c ($T_{ssd} = T_c$). It can fully satisfy the timely transmission required by AGV system.



T_c : a whole system broadcast cycle
 Δt : a DSF transmission time
 T_g : guard interval
 N : total number of nodes
 T_p : processing time

FIGURE 10: TDMA Schematic Diagrams of 16-node Scenario.



FIGURE 11: Appearance of WHM-4.

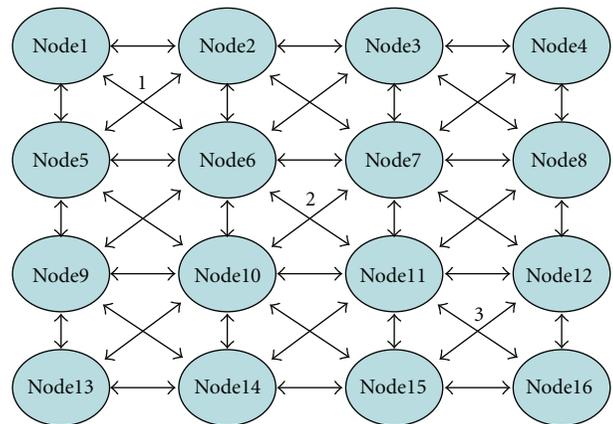


FIGURE 12: Topology A of 3-hop network.

5.2.2. *Three-Hop Scale.* To evaluate the Tssd performance in 3-hop networks, Topology A of 16-node network is established as shown in Figure 12. (e.g., node1 can only communicate with node2, node5 and node6). In order to simulate the mobile situation, some nodes' positions exchanged as shown in Figure 13, then the network topology turns to Topology B.

First, the simulations proved that after mobile nodes swapped their positions like Figure 13, S-DMDS protocol can adapt the topology change as usual without any additional cost. Thus, S-DMDS is proved to be efficient to deal with mobile situation.

From the result summarized in Figure 14, two cases represented by Topologies A and B indicate how long each node can obtain the sharing data from all the other nodes in

3-hops network. The maximum peaks of two cases represent the Tssd for the two networks, respectively. Those results suggest that the S-DMDS can provide very low system sharing delay of the network. In Topology A, it is only about 780 ms. A more interesting phenomena is that under the Topology B, the Tssd further decreases to around 640 ms. Meanwhile, sharing delay for each node becomes more balanced. Therefore, it means that system sharing delay in Topology B is better than that in the Topology A. After trying many times simulations, it is proved that under the Topology A the network takes the longest sharing delay in 3-hops scenario. The main reason causes by the following points:

The TDMA sending sequence arranges node to broadcast DSF from Node1 to Node16. Therefore, it is more efficient for the node of small ID like node1 to share its data with large ID node. (In Topology A, sharing data of node1 can be shared

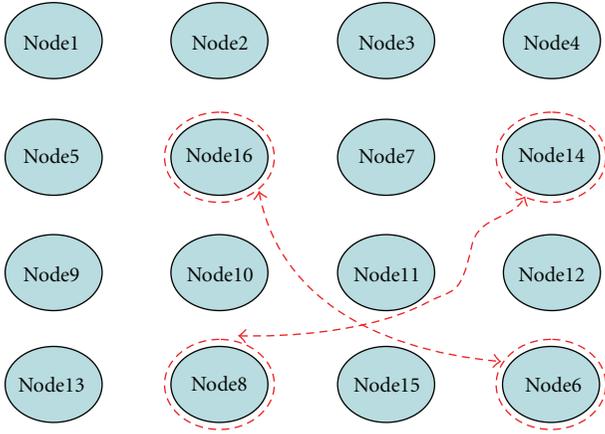


FIGURE 13: Mobile scenario (after exchanging positions of node 6 and node16 as well as the positions of node8 and node14, the network topology turns to Topology B).

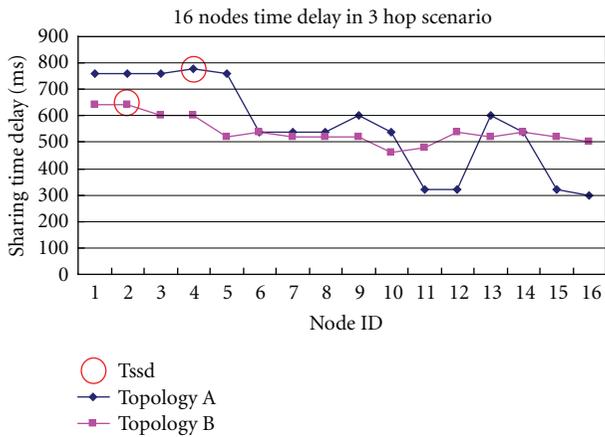


FIGURE 14: Sharing time delay in 3-hop scenario.

with all the other nodes in one T_c). In contrast, the sharing data of large ID node takes more time to be transmitted to small ID node. Besides, under Topology A, the nodes of large ID are arranged to gather together so that it leads to a delay overlap.

Hence, 3-hop scale simulation indicates that the largest Tssd is 780 ms.

5.2.3. Six Hops Scale. Transmission range has been set so that each node can only communicate with its neighbor node as shown in Figure 15 (e.g., node1 can only communicate with node2 and node 5), thus, network scale becomes 6-hops. Obviously, this scenario is the largest scale of 16 node network. Like 3-hop network, two scenarios are represented by Topologies A and B, respectively, in 6-hops scale. From the result described in Figure 16, the maximum Tssd is around 1.6 s.

With the network changing from 3-hop to 6-hop, the Tssd of both topologies increases. However, the increment under Topology A is much larger than that under Topology B.

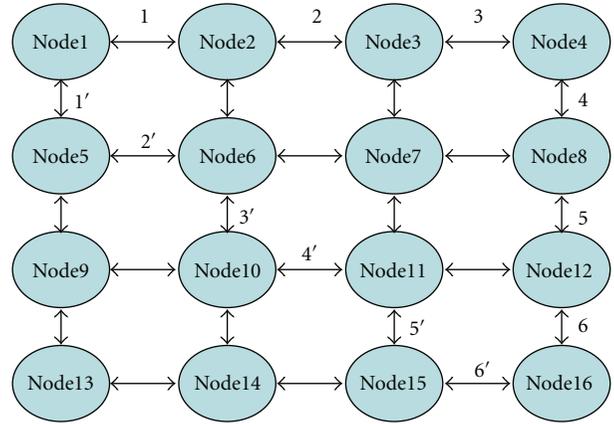


FIGURE 15: Topology A of 6-hop network.

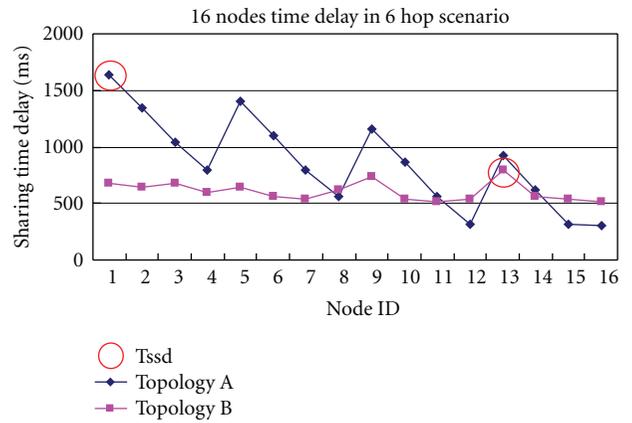


FIGURE 16: Sharing time delay in 6-hop scenario.

In summary, the influence caused by node position swapping in 6-hop scenario is larger than that in 3-hops scenarios.

5.2.4. Sharing Delay Analysis in Mobile Multi-Hop Scenarios. Figure 17 describes the Tssd in different hop scenario under Topologies A and B, respectively. Tssd rises rapidly following the increase of network hops. Moreover, Tssd under Topology A can be significantly decreased owing to the mobile situation realized under Topology B.

5.3. Comparison Simulation with NST-AODV. To explain how efficient S-DMDS can realize system data sharing, a comparison simulation with NST-AODV is provided. NST-AODV is a simple version of AODV based on IEEE802.15.4 for low cost device to realize mobile multi-hop wireless network.

5.3.1. A Discussion of Algorithm Complexity. Table 2 provides a general comparison of the two protocols. Since NST-AODV is proposed for point to point transmission, it needs to establish and maintain dynamic route table for mobile scenario as well as adopt many route control packets to coordinate each transmission.

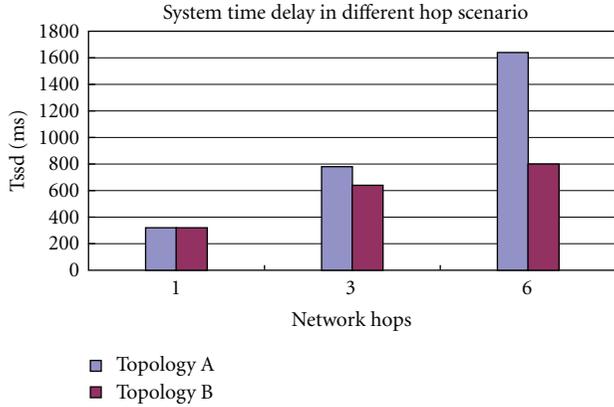


FIGURE 17: Relationship between network hops and system sharing delay in deferent topology.

TABLE 2: Comparison between S-DMDS with NST-AODV.

	NST-AODV	S-DMDS
Purpose	Point to point transmission	System data sharing
Dynamic routing table	Discovery and maintenance	No need
Media access	CSMA/CA	TDMA
Route control packet	RERR, RREQ, RREP	No need
RAM consumption	692 + 131 bytes	660 bytes

However, S-DMDS does not need such a complex process, because it is a topology-agnostic protocol. Therefore, NST-AODV takes more RAM consumption than S-DMDS. Besides, S-DMDS adopt TDMA as the media access method which is more stable than CSMA/CA, because AGV real time system is sensitive to packet loss.

5.3.2. Simulation in Static Scenario. NST-AODV is implemented into MICAz mote which has a same physical layer as WHM-4. [16] also gives a detailed performance evaluation of NST-AODV: to discovery route, it costs average 10 ms per hop; after route has been found, it also takes roughly 10ms per hop to execute an end to end transmission; when an active route changes, it takes around 50 ms to find a new route; and so forth.

In the simulation, both NST-AODV and S-DMDS are applied to finish system data sharing of 127 bytes in the 3-hops network of 16-node shown as Figure 12. Moreover, in NST-AODV, an additional delay will be caused (about 200 ms per node) once there are more than two nodes attempt to start discovery route or broadcast data at same time. To avoid data collision, node is assumed to share data one by one in simulation scenario.

The simulation result shown in Table 3 indicates that S-DMDS takes roughly 28.5% total sharing delay of NST-AODV in initial sharing situation and 38% total delay of NST-AODV in general situation. In initial situation, NST-AODV needs each node take 10 ms per hop to establish a

TABLE 3: Simulation result of system sharing delay in state scenario

	NST-AODV (Tssd_N)	S-DMDS (Tssd_S)	Ratio (Tssd_S/Tssd_N)
initial system sharing delay	9120 ms	2600 ms	28.5%
general system sharing delay	4560 ms	2060 ms	45.2%

TABLE 4: Simulation result of system sharing delay in mobile scenario.

	NST-AODV (Tssd_N)	S-DMDS (Tssd_S)	Ratio (Tssd_S/Tssd_N)
initial system sharing delay	9320 ms	2480 ms	26.6%
general system sharing delay	4760 ms	1980 ms	41.6%

route table of 15-node. And then each node spends another 10 ms per hop to share its data to the other 15-node. 16-node take about 9120 ms to share data. In same situation, S-DMDS only need about 780 ms for 16-node to share data to all the others at first round.

At this moment, S-DMDS can only share 20 bytes data because of DSF definition, so it needs approximate another 6 Tc to continue sharing the rest of data.

In general sharing situation, since NST-AODV can skip route discovery operation, it can reduce the system sharing delay down to 4560 ms. And S-DMDS only needs slightly less than 7 Tc (about 2060 ms) to finish system data sharing. Overall, S-DMDS is proved to perform much better than NST-AODV in terms of system data sharing.

5.3.3. Simulation in Mobile Scenario. In order to evaluate the performance in mobile scenario, node positions are allowed to be exchanged as shown in Figure 13. Table 4 shows the result of system sharing delay in mobile scenario.

After 4 nodes changes position, NST-AODV needs about 200 ms to recover the route. In contrary, as previous discussion in Section 5.2.2, the system sharing delay of S-DMDS will decrease after topology changes to the Topology B shown as Figure 13. Moreover, the change extent of S-DMDS is smaller than that of NST-AODV. Therefore, S-DMDS outperforms NST-AODV in mobile situation.

The simulation result proves that S-DMDS is much better than NST-AODV for system data sharing in 3-hop network of 16-node. Since topology changes rapidly in AGV system, S-DMDS takes around 41.6% system sharing delay of NST-AODV in most cases.

5.4. The Experiments of 5 Nodes Based on WHM-4. To prove S-DMDS can work in reality, 5-node experiments are provided. The length of DSF is defined as 78 bytes, so each DSF transmission (Δt) takes slightly less than 5 ms and a guard interval (T_g) is set as 15 ms in 5-node experiments. The broadcast cycle (T_c) changes to be around 100 ms. To evaluate the performance of S-DMDS in real

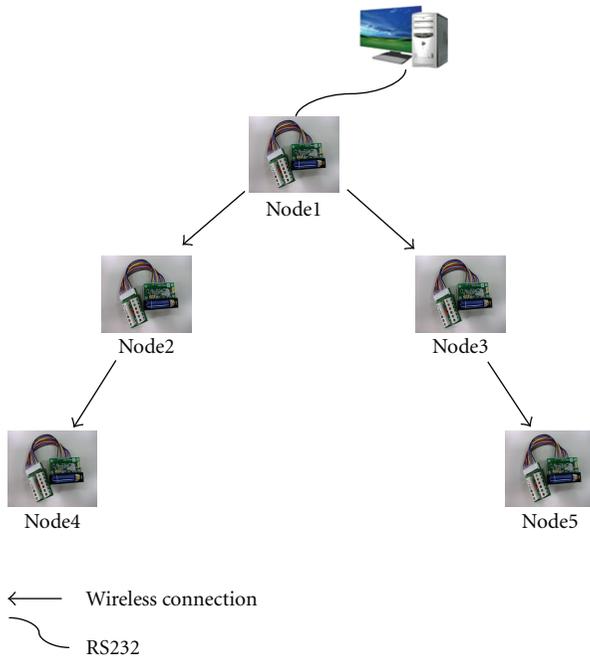


FIGURE 18: Experiment network structure 1.

implementation, two experiment scenarios are provided including 2 hops mobile transmission scenario and 4 hops system data sharing scenario.

5.4.1. *Two Hops Mobile Transmission Experiment.* Two-hop transmission with two mobile nodes experiment is shown in Figure 18. (other nodes take max 2 hops to receive node1's sharing data) Node1 is connected with the PC by RS232. By using wireless communication, node1 can send the new sharing data from the PC to the other nodes such as node2 and node3. Then node2 and node3 will transform the sharing data to node4 and node5, respectively.

To test performance of S-DMDS in case of mobile situation, the positions of node2 and node5 are exchanged to simulate the dynamic topology. So the network structure turns to the structure 2 shown in Figure 19, and the results of the above two experiments are summarized in Figure 20 to explain how much time each node takes to receive the sharing data from node 1.

By doing the experiments, first, S-DMDS is proved to be really implemented into WHM-4 to build a mobile multi-hop network. Secondly, a node such as node1 is very efficient to share its sharing data to all the other nodes in the network. In addition, when the locations of node2 and node5 changed, the network can work normally without any influence.

5.4.2. *System Data Sharing Experiment of Four Hops.* A four-hop network scenario is provided to evaluate the system sharing delay in real multi-hop situation shown as Figure 21.

Figure 22 shows the time delay of each node receiving the data from node1 and node 5. To share node1's data, it only needs 1 broadcast cycle, but to share node5's data it needs 3.2 broadcast cycle.

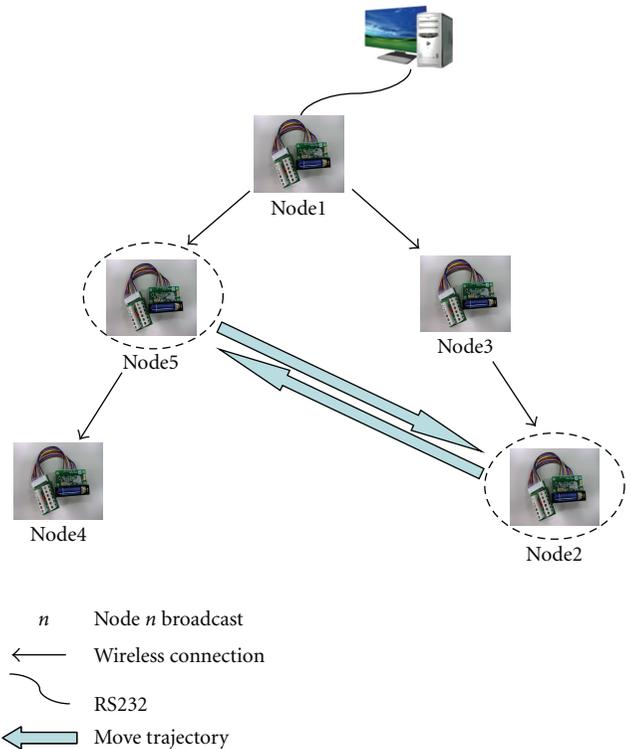


FIGURE 19: Experiment network structure 2.

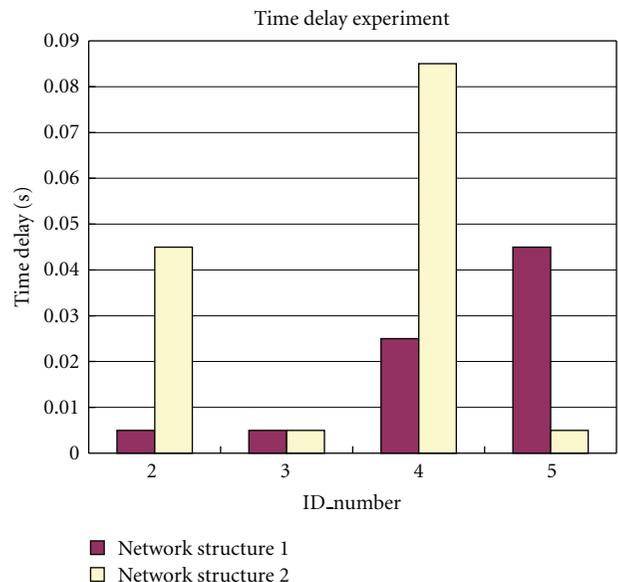


FIGURE 20: Experiment result of two-hop network.

Because after node1 broadcasted the data, node2, node3 and node4 start to broadcast one by one. So the data from node1 can be sent to node5 in one broadcast cycle. Before the middle nodes (such as node 2, node 3 and node 4) broadcast DSF, they also input their sharing data to the DSF, therefore, each node can get previous nodes' data. So node5 can obtain all the nodes' sharing data. And then the data from node5

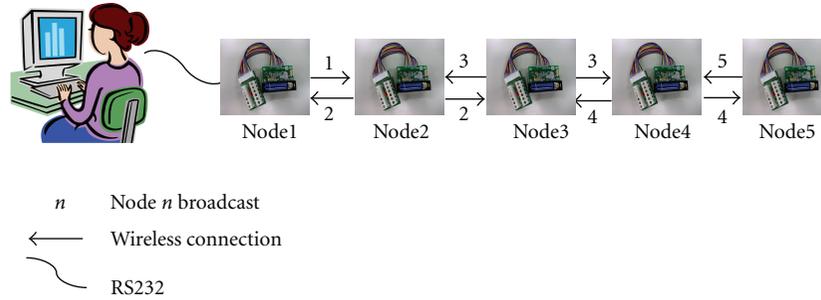


FIGURE 21: Four hops transmission experiment.

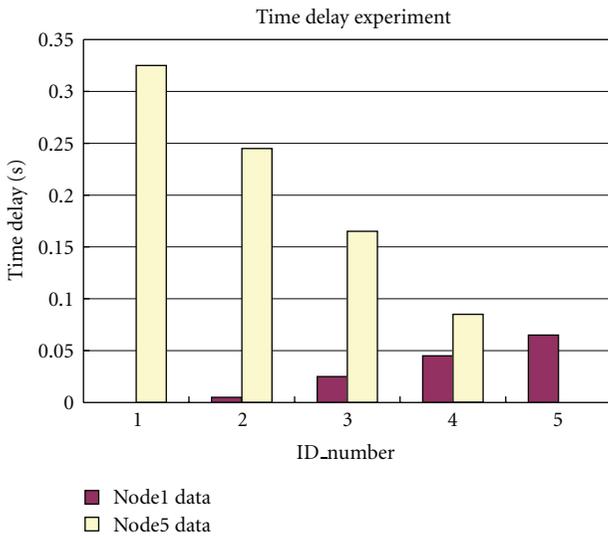


FIGURE 22: Experiment result of four-hop network.

starts to be delivered to node1. Because it is the last node in broadcast cycle, the data needs wait about 0.08 s to be sent to node4. By parity of reasoning, it needs roughly 0.32 s to deliver node5’s data to node1. At this time, all the nodes have kept the other nodes’ data, thus, the system data sharing has been realized.

6. Conclusion

To realize a mobile wireless sensor network for AGV system that provides each device exchange its state information timely with all the others, S-DMDS protocol is proposed. The proposed protocol is implemented by simple wireless module with low calculation overhead. It adopts a regularly broadcasted DSF as a data exchange media to realize data sharing of whole system faster than conventional approaches. Moreover, dynamic topology will not influence the performance of network. TDMA is employed so that S-DMDS can provide stable and efficient accessing without transmission collision.

S-DMDS protocol has been implemented into WHM-4 hardware module which will be equipped to the devices of AGV system. 16-node simulation results indicate that

maximum Tssd in 3-hop and 6-hop network is 780 ms and 1.6 s, respectively. Furthermore, Tssd can be significantly decreased in the mobile situation. Therefore, the Tssd of S-DMDS is low enough to satisfy AGV system in dynamic multi-hop situation. In addition, the results of comparison simulation show that compared with NTS-AODV, S-DMDS takes only about 41.6% system sharing delay as well as 80% RAM consumption. At last, 5-node experiment proved S-DMDS can work well in real environment.

In conclusion, S-DMDS protocol is an outstanding solution to compose mobile multi-hop WSN with simple wireless sensor module for AGV system.

Acknowledgments

The authors would like to sincerely thank Mr. T. Tsuji and Mr. T. Ottawa of Logic Product Corporation.

References

- [1] T. S. Rappaport, “Indoor radio communications for factories of the future,” *IEEE Communications Magazine*, vol. 27, no. 5, pp. 15–24, 1989.
- [2] A. Willig, M. Kubisch, C. Hoene, and A. Wolisz, “Measurements of a wireless link in an industrial environment using an IEEE 802.11-compliant physical layer,” *IEEE Transactions on Industrial Electronics*, vol. 49, no. 6, pp. 1265–1282, 2002.
- [3] P. Ferrari, A. Flammini, D. Marioli, and A. Taroni, “IEEE802.11 sensor networking,” *IEEE Transactions on Instrumentation and Measurement*, vol. 55, no. 2, pp. 615–619, 2006.
- [4] C. Koulamas, S. Koubias, and G. Papadopoulos, “Using cut-through forwarding to retain the real-time properties of profibus over hybrid wired/wireless architectures,” *IEEE Transactions on Industrial Electronics*, vol. 51, no. 6, pp. 1208–1217, 2004.
- [5] H. Ye, G. C. Walsh, and L. G. Bushnell, “Real-time mixed-traffic wireless networks,” *IEEE Transactions on Industrial Electronics*, vol. 48, no. 5, pp. 883–890, 2001.
- [6] “IEEE 802.15.4-2003 MAC and PHY specifications for low-rate wireless personal area networks,” 2003.
- [7] “BT SIG, Specification of the Bluetooth System,” 2004, <http://www.bluetooth.org/>.
- [8] T. Lennvall, S. Svensson, and F. Hekland, “A comparison of WirelessHART and ZigBee for industrial applications,” in *Proceedings of the 7th IEEE International Workshop on Factory Communication Systems (WFCS ’08)*, pp. 85–88, May 2008.

- [9] D. Johnson, D. Maltz, and J. Broch, "The dynamic source routing protocol for mobile ad hoc networks," IETF MANET Working Group, draft-ietfmanet-dsr-03.txt, November 1999.
- [10] C. E. Perkins and E. M. Royer, "Ad-hoc on demand distance vector routing," IETF MANET Working Group, draft-ietfmanet-aodv-05.txt, March 2000.
- [11] K. Kim, S. D. Park, G. Montenegro, and S. Yoo, "6LoWPAN ad hoc on demand distance vector routing (LOAD)," draft-daniel-6lowpan-load-adhocrouting-01, IETF Internet Draft (Work in progress), July 2005.
- [12] G. Montenegro, "AODV for IEEE 802.15.4 networks," draft-montenegro-lowpan-aodv-00, IETF Internet Draft (Work in progress), July 2005.
- [13] I. Chakeres and L. Klein-Berndt, "AODVjr, AODV simplified," *Mobile Computing and Communications Review*, vol. 6, no. 3, pp. 100–101, 2002.
- [14] "TinyAODV implementation, TinyOS source code repository," <http://cvs.sourceforge.net/viewcvs.py/tinyos/tinyos-x/contrib/hsn/>.
- [15] C. E. Perkins, E. Belding-Royer, and I. Chakeres, "Ad hoc on-demand distance vector (AODV) routing," draft-perkins-manet-adovbis-01, IETF Internet Draft (Work in progress), February 2004.
- [16] C. Gomez, P. Salvatella, O. Alonso, and J. Paradells, "Adapting AODV for IEEE 802.15.4 mesh sensor networks: theoretical discussion and performance evaluation in a real environment," in *Proceedings of the International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoW-MoM '06)*, pp. 159–167, June 2006.
- [17] "MOTE-KIT2400 Datasheet," <http://www.xbow.com/>.
- [18] J. Elson, L. Girod, and D. Estrin, "Fine-grained network time synchronization using reference broadcasts," in *Proceedings of the 5th symposium on Operating systems design and implementation (OSDI '02)*, December 2002.
- [19] M. Maroti, B. Kusy, G. Simon, and A. Ledeczi, "The flooding time synchronization protocol," in *Proceedings of the Second International Conference on Embedded Networked Sensor Systems (SenSys '04)*, pp. 39–49, November 2004.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

