International Journal of Advances in Computer Networks and its Security – IJCNS
Volume 4 : Issue 1 *[ISSN 2250 – 3757]*
Publication Date : 09 January 2014

Election Root Node of Flooding Time Synchronization Protocol by Value of Neighbor Node

Ronnachai Sivasri
Department of Computer Engineering
Faculty of Engineering, Khon Kaen University
Khon Kaen,Thailand
Ronnachai.siva@gmail.com

Chatchai Khunboa
Department of Computer Engineering
Faculty of Engineering, Khon Kaen University
Khon Kaen,Thailand
chatchai@kku.ac.th

***Abstract*—Wireless sensor networks (WSNs) applications, that are similar to that of other distributed systems, and a time-synchronization protocol has been developed. The time synchronization of WSNs is required for coordination and data consistency. To improve accuracy for WSNs by controlling the root node is still a problem. The Flooding Time Synchronization Protocol (FTSP) is a one of protocol for synchronization. FTSP utilizes a simple election process based on minimum node IDs. In this paper, we propose (FTSP) a new election root node method by neighbor node (NB). We analyzed the time synchronization error by calculating the average of the pairwise differences of the reported global times by a new root election. The time average was reduced suggesting FTSP based on neighbor node is more advantages than lower ID and neighbor node method conferred significantly better accuracy than lower ID method**

***Keywords*— time synchronization; wireless sensor network; Flooding Time Synchronization Protocol**

I. Introduction

Wireless sensor networks (WSNs) applications, that are similar to that of other distributed systems, and a time-synchronization protocol has been developed. The time synchronization of WSNs is required for coordination and data consistency. To improve accuracy for WSNs by controlling the root node is still a problem.

The FTSP [1] is a good protocol for synchronization. It uses low bandwidth for commutation and robust against node and link failures. FTSP achieves its robustness by utilizing periodic flooding of synchronization messages and it flexible to update topology. FTSP uses the concepts of MAC layer time-stamp and skew compensation with linear regression for compensate the relevant error sources.

Location of root node is problem of wireless sensor network by FTSP. Location of root node is important for FTSP because it has effect to error on time synchronization. FTSP utilizes a simple election process based on unique node IDs, if the root node located at unsuitable for network such as edge of network, there must have a problem. Although FTSP is used to compensate for the relevance error sources by skew compensation with linear regression, the location of root node therefore is important for synchronization.

In this paper, we propose (FTSP) a new election root node method by neighbor node. We analyzed the time synchronization error by calculating the average of the pairwise differences of the reported global times by a new root election. The time average was reduced. This finding suggest that FTSP based on neighbor node is more advantages than lower ID and neighbor node method conferred significantly better accuracy than lower ID method

II. Related work

A. *The Flooding Time Synchronization Protocol*

The FTSP is an efficient protocol for synchronization. It uses low bandwidth for commutation and robust against node and link failures. FTSP achieves its robustness by utilizing periodic flooding of synchronize messages and flexible to update topology. FTSP uses the concepts of MAC layer time-stamping and skew compensation with linear regression to compensate the relevant error sources.

FTSP is synchronization with the local clock of collaborating nodes. The clock drift of nodes is necessary for accuracy between nodes and delays message in transmissions as summarizes in table 1, the FTSP using linear regression to compensate for clock drift.

In multi-hop, FTSP use reference points to operate synchronization. Sending and receiving periodic broadcast messages are generate reference points and the reference points are either transmitted by synchronization-root (root, for short) or any synchronized node in network. During network is initially synchronized, it then elect and dynamically reelect the root by simple election process based on unique node IDs.

B. *Leader Election Algorithms*

Leader election algorithm is a method in distributed computing for dynamic selecting a root node. In wireless sensor network, Leader election algorithms find many applications. Simple algorithms used, such as Bully algorithm, Ring algorithm and Voting algorithm. For example, FTSP method likes Bully algorithm but it is used lower IDs by unique IDs.

Leader election algorithm for broadcast networks was proposed to tolerate arbitrary process failures [9]. Every node will broadcast ID information to all other nodes. Then each node receive ID which has higher than root ID, it broadcasts its ID to all other nodes. If a node dose not receives any message for a time interval, it assumes itself to be the leader. Their algorithm is similar to election leader of FTSP so it cannot avoid bad location.

TABLE I. Summarizing delays message in transmission [1]

TIME	Magnitude	Distribution
Send and Receive	0-100 ms	nondeterministic, depends on the processor load
Access	10-500 ms	nondeterministic, depends on the channel contention
Transmission / Reception	10 – 20 ms	deterministic, depends on message length
Propagation	< 1µs for distances up to 300 meters	deterministic, depends on the distance between sender and receiver
Interrupt Handling	< 5µs in most cases, but can be as high as 30µs	nondeterministic, depends on interrupts being disabled
Encoding plus Decoding	100 – 200µs, < 2µs variance	deterministic, depends on radio chipset and settings
Byte Alignment	0 – 400µs	deterministic, can be calculated

Leader election algorithm for mobile Ad Hoc networks was proposed with the following assumptions about node and system architecture [10] such as each node has a value associated with it, unique and order node IDs, link are bidirectional and FIFO and each node has large buffer size whereas FTSP is broadcast message, low bandwidth and scalability is not suitable algorithm for FTSP..

Leader algorithm has been proposed for static network [11]. The algorithm use spanning trees to find a prospective leader at the root of spanning trees. However, it works when the topology is static network. Thus, this algorithm is not suitable for mobile network.

III. election root node

We created a table to keep values of neighbor node, node id, a state table and age of node and then calculated neighbor of any message. The data were reported to the network and by broadcast. Denote the value of *NEIGHBOR_NODE* by *N*. The pseudo-code describes method as shown in Fig. 1. When nodes received message, message of N were compared with their N. If nodes have lower N than message nodes will broadcast node id of massage and update rootID field. Therefore, each synchronization message contains four fields: the nodeID, the rootID, the rootNeighbor and the timestamp.

```
addTable form message.
calculateNeighbor form message.

if Neighbor of message is greater than myNeighbor of root.
        Update value and forward message.
If Neighbor of message equal to myNeighbor of root.
        If rootID of message is lower than my rootID.
                Update value and forward message.
```

Figure 1. The pseudo-code of forward root message.

IV. EXPERIMENT

We implemented FTSP on MICAZ platform and tested the protocol focusing on the election root node and location of root node by Avrora simulator [4] and MICAZ mote. In this study, we tested the protocol focusing on worst case of election root node by lower ID. Table II is show configuration for all experimental.

TABLE II. Configuration data for experimental

OS	TinyOS 2.1.1
Platform	Micaz
Max Entries	8
Beacon rate	10 s
Root timeout	100 s
Ignore root MSG	40 s
Entry valid limit	4 unit
Entry send limit	3 unit
Entry throwout limit	500 µs
Neighbor limit	30 unit

A. *Experimental error time synchronization of worst case.*

In worst case of Lowest ID on random topology and NB were tested a random node with 20, 40 and 60 nodes on a 30x60 and 30x90 square meters, each node has a transmission distance is 15 meters. After the system entered the steady state, randomly data were generated every 1 minute of the experiment 100 times.

The experiment was performed with network model that it has the highest average of the pairwise difference of report global time. We calculated a difference report of global time and average of the pairwise difference of the reported global time between election root node by lowest ID and value of NB. On a 30x60 square meter, election root node by lowest ID and NB has the average time synchronization error was 4.77µs and 3.37µs, respectively. On a 30x90 square meter, election root node by lowest ID and NB has the average time synchronization error is 6.57µs and 3.98µs, respectively (Table 3).

TABLE III. Mean of error time by worst case

Nodes	Mean of error time			
	30x60 m^2		*30x90 m^2*	
	FTSP	*NB*	*FTSP*	*NB*
20	3.61	3.41	6.62	3.75
40	6.2	3.36	7.33	4.18
60	3.5	3.35	5.78	4.03

B. Experimental error time synchronization of random

Lowest ID and NB was tested with 60 nodes on 30x60 square meters each node has a transmission distance is 15 meters. After the system entered the steady state, randomly data were generated every 1 minute of the experiment 100 times for 20 times.

The election root node by lowest ID and NB has the average time synchronization error was 3.9µs and 3.45µs, respectively. Moreover, NB can reduce error time 11.54%.

V. Conclusion and Discussion

This paper is presented an effect of root node location for time synchronization and also able to show different of method of election root node by neighbor node.

The election root node by value of neighbor node was found to reduce the average error. The percentages of reducing error time were observed, 30-45% can be found when the distribution of nodes is markedly clustered whereas 10-20% when the distribution of the node that is not clustered form.

However, election root node by value of neighbor node can avoid worst case of election root node by lowest id (edge of network, non-uniform distribution system

Acknowledgment

We thank Office of The National Broadcasting and Telecommunications Commission for supporting the equipment and facility.

References

[1] M. Maroti, B. Kusy, G. Simon and A. Ledeczi, "The flooding time synchronization protocol". Proc. the 2nd International Conference on Embedded Networked Sensor Systems, pages 39–49, Nov. 2004.

[2] D. Cox, E. Jovanov, and A. Milenkovic, "Time synchronization for ZigBee networks," in *Proceedings of the Thirty-Seventh Southeastern Symposium on System Theory, 2005. SSST '05*, 2005, pp. 135– 138.

[3] Dan, A, "Localization with enhanced location accuracy using RSSI in WSN.",Advanced Networks and Telecommunication Systems (ANTS), 2011 IEEE 5th International Conference on, Pages 1 – 6, 18-21 Dec. 2011.

[4] "Avrora - The AVR Simulation and Analysis Framework." [Online]. Available: http://compilers.cs.ucla.edu/avrora/index.html. [Accessed: 13-Sep-2012].

[5] Hyochang Ahn, "Simulation of a RSSI-Based Indoor Localization System Using Wireless Sensor Network". Ubiquitous Information Technologies and Applications (CUTE), 2010 Proceedings of the 5th International Conference on, Pages 1 - 4, 16-18 Dec. 2010.

[6] F. Sivrikaya and B. Yener, "Time synchronization in sensor networks: a survey", Network, IEEE In Network, IEEE, Vol. 18, No. 4. (2004), pp. 45-50.

[7] P Levis, S. Madden, J. Polastre, R. Szewczyk, K. Whitehouse, A. Woo, D. Gay, J. Hill, M. Welsh, E. Brewer and D. Culler, "TinyOS: An Operating System for Sensor Networks", Ambient Intelligence, 2005, pp. 115-148

[8] "TinyOS Home Page." [Online]. Available: http://webs.cs.berkeley.edu/tos/. [Accessed: 13-Sep-2012].

[9] J. Brunekreef, J. Katoen, R. Koymans and S. Mauw. Design and Analysis of Leader Election Protocols in Broadcast Networks. In Distributed Computing, vol. 9 no. 4, pages 157-171, 1996.

[10] S. Vasudevan, J. Kurose, and D. Towsley, "Design and analysis of a leader election algorithm for mobile ad hoc networks," in *Proceedings of the 12th IEEE International Conference on Network Protocols, 2004. ICNP 2004*, 2004, pp. 350 – 360.

[11] R. Gallager, P. Humblet and P. Spira. A "Distributed Algorithm for Minimum Weight Spanning Trees.", In *ACM Transactions on Programming Languages and Systems*, vol.4, no.1, pages 66-77, January 1983.