http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 8 (5): 776-780, 2009
ISSN 1812-5638

Range-Based Clock Synchronization Protocol for Wireless Sensor Networks

Zhetao Li, Renfa Li and Liangjiao Liu
School of Computers and Communications, Hunan University, Changsha 410082, China

Abstract: This study presents a novel range-based clock synchronization protocol by exploiting non-synchronized TDOA. It combines the method of Network Time Protocol (NTP) with round-trip TDOA. By applying round-trip TDOA repeatedly, all nodes in network efficiently synchronize to each other. Furthermore, clock synchronization and ranging can be performed simultaneously. Simulation results show it outperforms Timing-sync Protocol for Sensor Networks (TPSN) and Reference Broadcast Synchronization (RBS) in terms of the number of message exchanges and synchronization error.

Key words: Wireless sensor networks, clock synchronization, ultrasound ranging, time difference of arrival, parallel algorithm

INTRODUCTION

Clock synchronization is an important service in any distributed system, including Wireless Sensor Network (WSN) systems (PalChaudhuri *et al*., 2004). Most Applications of WSN require the accurate clock synchronization among sensor nodes. For example, TDMA (time-division-multiple-access) medium access scheduling and ranging based on Time of Arrival (TOA) require clock synchronization. Because of the lack of infrastructure in WSN, there is no global clock. Each deployed sensor nodes have theirs own internal clock and its own notion of time (Sivrikaya and Yener, 2004). Practically these clocks made of crystal oscillator can easily drift seconds per day (Sundararaman *et al*., 2005). Local clock of every node may not remain always synchronized, although they might be synchronized when they start. Hence, clock synchronization protocol is indispensable in the application of WSN.

Clock synchronization problem has been investigated thoroughly in Internet and wired networks. Network transport protocol (Mills, 1992) is in the Internet standard for clock synchronization. However, its implementation is too heavy-weight to be supported by sensor nodes. As for GPS (Bernhard *et al*., 1992), it is too expensive to attach on cheap node. Most of these existing protocols (Duda *et al*., 1987; Lamport, 1978) have focused on minimizing the synchronization error and achieving maximum accuracy. They were not designed with the intrinsic properties of WSN, such as limited resources of energy and computation, in mind (Greunen and Rabaey, 2003). Therefore, there has been an increasing research focus on designing synchronization algorithms specifically for WSN.

Thus far, a number of synchronization protocols for WSNs have been reported (Elson and Romer, 2003; Ganeriwal *et al*., 2003; Greunen and Rabaey, 2003; Maróti *et al*., 2004). In general, existing protocols were categorized as Sender-Receiver Synchronization (SRS) and Receiver-Receiver Synchronization (RRS). A representative based on RRS is reference broadcast synchronization (Elson and Romer, 2003). Every node periodically broadcasts to its neighbor reference beacons without explicit timestamps. By reducing non-deterministic latency involved in message transmission, namely the send time and the access time, RBS can provide a high degree of synchronization accuracy. For n nodes and m reference broadcast packets, RBS has a complexity of $O(mn^2)$. In contrast, the time-sync protocol for sensor networks (Ganeriwal *et al*., 2003) based on SRS decreases the doubled overheads of RBS. The basic idea of TPSN is similar to NTP. Timing-sync protocol for sensor networks is based on hierarchical structure of the network and synchronizes the entire network by exchanging timing messages along every branch (edge) of the hierarchical tree (Noh *et al*., 2008).

On the other hand, clock synchronization is indispensable in the implementation of TOA and TDOA. In TOA, the range between the sender and the receiver was determined by the duration of time required for a

Corresponding Author: Zhetao Li, School of Computers and Communications, Hunan University, Changsha, 410082, China
Tel: 86 155 73261946

signal to travel between them (Son *et al*., 2003). This information is correct only if the sender and the receiver remain accurately synchronized to Universal Time Coordinated (UTC). However, in TDOA systems the receiver is allowed to have a lower-accuracy clock. The sender will transmit a radio (RF) and ultrasonic (US) signal at the same time. The receiver can measure the time difference between the arrivals and thus deduce the distance between the sender and the receiver (Mailaender, 2007). Although, the node's clock offset is not necessary to be synchronized with UTC, the clock skew is needed.

All the clock synchronization methods rely on some sort of message exchange between nodes, the same to range estimation. Therefore, we propose a novel hybrid scheme by exploiting the ability of message exchange between nodes. To the best of present knowledge, there is no earlier study that focuses on range-based synchronization. It is understandable because that existing research was based on the assumption that clock synchronization is a prerequisite for time-based ranging.

In this study, we argue that clock synchronization can be achieved by non-synchronized TDOA. We introduce Range-Based Clock Synchronization (RBCS) schemes by combining NTP with round-trip TDOA. The algorithm eliminates the need to synchronize all nodes before ranging. In practice, clock synchronization and ranging can be performed simultaneously. We also show that RBCS is a lightweight and robust synchronization with better accuracy than either RBS or TPSN by simulations.

RANGE-BASED CLOCK SYNCHRONIZATION

We use two-way TDOA between a pair of nodes in order to synchronize correctly. Figure 1 shows how node A and node B will synchronize. Node A sends two different types of signal simultaneously to node B who then sends reply message back to node A. T_1, T_5 and T_6 are measured by A's local clock while T_2, T_3 and T_4 represent the local clock's readings on B.

Node A initiates the synchronization by sending B two signals (in particular RF and ultrasound) simultaneously at time T_1. The value T_1 is also contained in these signals. Node B receives these signals at time T_2 and T_3. T_2 and T_3 are called the received timestamp and:

$$T_2 = T_1+\theta_{AB}+\delta_{ABrf}$$

$$T_3 = T_1+\theta_{AB}+\delta_{ABus}$$

where, θ_{AB} is the local clock offset, δ_{ABrf} and δ_{ABus} represent the propagation delay of radio and ultrasound signal, respectively.

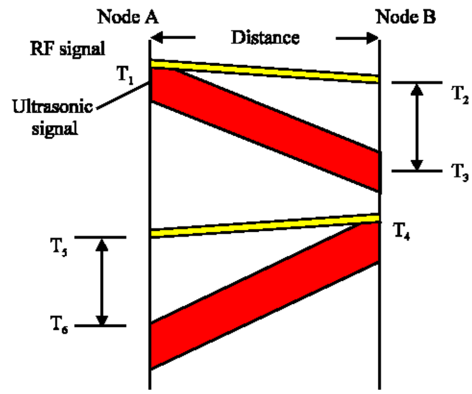

Fig. 1: Principle of RBCS

By using of TDOA, we deduce the distance between A and B with B's notion of timescale using this equation:

$$d_{AB} = (T_3 - T_2)\times\frac{c_{rf}c_{us}}{c_{rf} - c_{us}}$$

where, c_{rf} is the speed of RF signal, c_{us} is the speed of ultrasonic signal.

Ideally, if the receiver B is precisely synchronized to UTC, the measurement of TDOA is already enough to indicate the absolute distance between them. However, precise synchronization is hard to achieve. So, the measurement of non-synchronized TDOA is relative.

At time T_4, node B acknowledge A's synchronization by sending back two signals which includes T_1, T_2, T_3 and T_4. Node A receives them at time T_5 and T_6.

$$T_5 = T_4-\theta_{AB}+\delta_{ABrf}$$

$$T_6 = T_4-\theta_{AB}+\delta_{ABus}$$

By the same token, we deduce the distance between A and B with A's notion of timescale using equation like:

$$d_{BA} = (T_6 - T_5)\times\frac{c_{rf}c_{us}}{c_{rf} - c_{us}}$$

Thus, θ_{AB}, δ_{ABrf} and δ_{ABus} can be estimated with the following below:

$$\delta_{ABrf} = \frac{[(T_2 - T_1) + (T_5 - T_4)]}{2}$$

$$\delta_{ABus} = \frac{[(T_3 - T_1) + (T_6 - T_4)]}{2}$$

$$\theta_{AB} = \frac{[(T_2 - T_1) - (T_5 - T_4) + (T_3 - T_1) - (T_6 - T_4)]}{4}$$

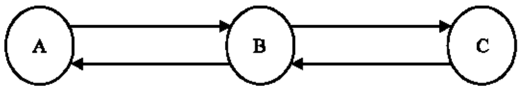

Fig. 2: An example of multi-hop synchronization

As a result, the clock on node A could be synchronized with node B's clock by adding offset θ_{AB}.

Instead of averaging the offset from multiple observations, we perform a very simple yet effective algorithm to correct skew. The clock skew between A and B k_{AB} can be obtained by:

$$k_{AB} = \frac{d_{AB}}{d_{BA}}$$

Theoretically speaking, RBCS can synchronize two clocks on node A and B accurately by the offset θ_{AB} and the skew k_{AB}.

Multi-hop synchronization: Multi-hop synchronization is an extension of the pair-wise (single-hop) synchronization algorithm. Consider the example topology shown in Fig. 2. Node A and C can not hear each other, but each of them are heard by node B.

If the functions of synchronization between node A and B, B and C are represented by:

$$T_B = f_{BA}\ (T_A, \theta_{AB}, k_{AB})$$

$$T_C = f_{CB}\ (T_B, \theta_{BC}, k_{BC})$$

Then, the relationship of clock between node A and C can be written as:

$$T_C = f_{CB}\ (f_{BA}(T_A, \theta_{AB}, k_{AB}), \theta_{BC}, k_{BC})$$

After iterative calculation, we can obtain network-wide synchronization by multi-hop synchronization.

The RBCS protocol performs synchronization only when it is needed. Clocks run untethered at their own natural rates and the timestamps from different clocks are compared only when an event of interest occurs. This technique is similar to reactive routing. By synchronizing the nodes only when necessary, energy is conserved because the nodes can be switched to power-saving mode at all other times (Sundararaman *et al*., 2005). So, The RBCS is a kind of post-facto synchronization.

Internal and external synchronization: Although, internal synchronization is sufficient for many applications, others require absolute time as measured by an external reference such as UTC.

If the standard source of time such as UTC is not available, internal synchronization can be performed network-wide by the use multi-hop synchronization. In addition, the distance computed by TDOA is relative. If UTC is provided, we assume it as a global time reference. It initiates the synchronization by synchronizing with all immediate (single-hop) neighbors. Next, each neighbor of UTC synchronizes with their subsequent neighbor. This process continues until all nodes are reached (Greunen and Rabaey, 2003). The distance obtained by TDOA is absolute because all clocks synchronize to UTC.

SIMULATION RESULTS

Here, we verify our algorithm through simulations. In particular, we examine two performance parameters, i.e. the number of message exchanges and synchronization error. The simulations were conducted using NS2. It has been widely used in academic research. The nodes are placed uniformly at random within a 2-dimensional 50×50 m rectangular area. The radius of communication is 7 m. The simulation was executed for 18,000 sec or 5 h. Present setup includes multi-channel Medium Access Control (MAC) model. This MAC model minimizes collisions and thus collisions were considered negligible in the simulation.

Number of message exchanges: The number of message exchanges is an important performance indicator because it is closely related to the power consumption demand by the synchronization. Figure 3 shows the simulation result for the total number of message exchanges in various synchronization protocols. It is based on simulations of connected ad-hoc networks consisting of 300 nodes. In Fig. 3a, when the number of neighbor node is more than 6, the number of messages sent in RBCS is the least. This is because every node has to send two messages in RBCS. In Fig. 3b, there was no significant difference in the number of messages received between RBCS and TPSN. When the number of neighbor nodes is 10, the RBCS is expected to save about 75% of the amount of messages sent and 70% of the amount of message received compared with RBS. Considering the additional ability of ranging, RBCS has higher efficiency than TPSN and RBS in the use of message exchange.

Synchronization error: Figure 4 shows the synchronization error of the compared protocols. The TPSN achieves two times better performance than RBS by time-stamping the radio messages in the MAC layer of the radio stack and by relying on a two-way message exchange. The basic idea of RBCS is somewhat similar to NTP. In RBCS, two-way message exchange performed between pair-wise nodes was used to correct clock offset. Moreover, round-trip TDOA was used to correct clock

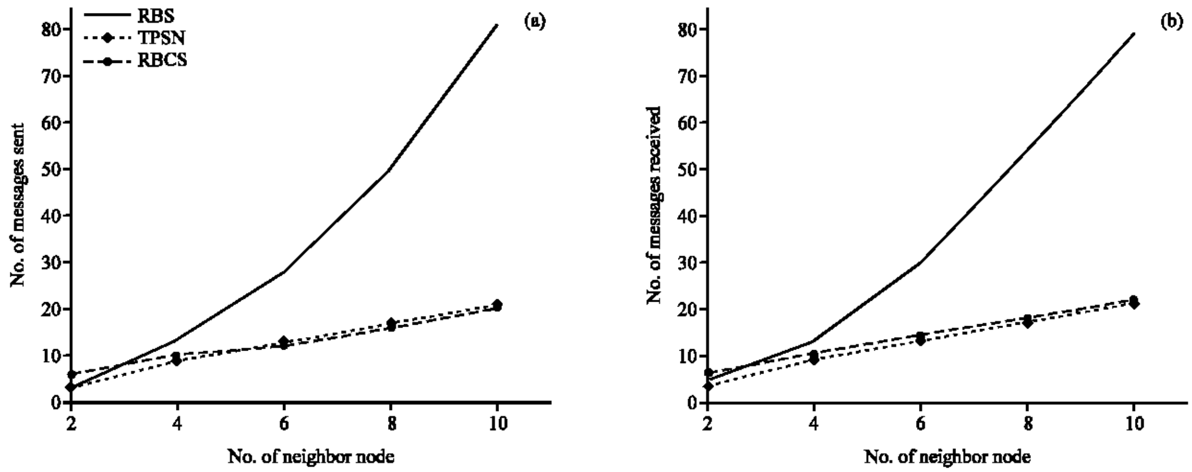

Fig. 3: The number of messages exchanges, (a) the No. of messages sent and (b) the No. of messages received

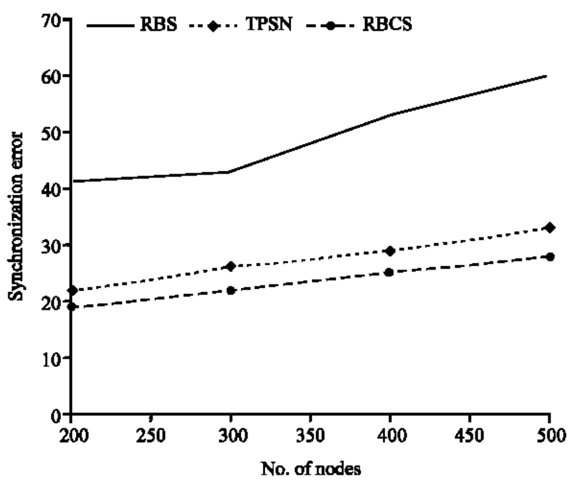

Fig. 4: Synchronization error as a function of nodes

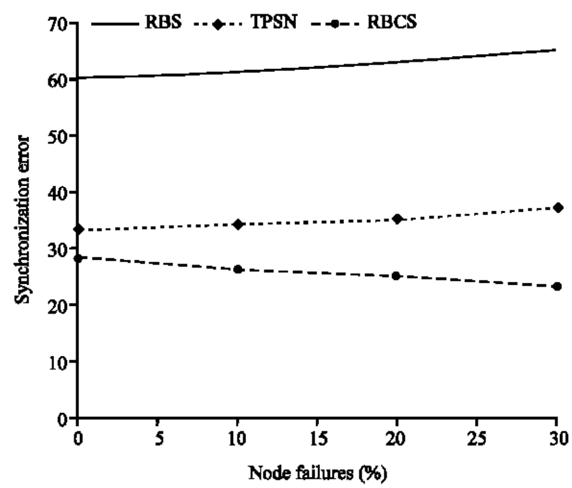

Fig. 5: Synchronization error with node failures

skew. Therefore, the precision synchronization of RBCS is better than TPSN. However, synchronization error increases with the number of nodes.

Robustness: Errors caused by failing hardware or drained batteries are the norm rather than the exception in WSN. Figure 5 shows synchronization error caused by different percentages of node failures. It is based on simulations of connected ad-hoc networks consisting of 500 nodes. Turning off a fraction of nodes simulated node failures in my simulations.

As the number of node failures increased, it appears that the synchronization error will decrease. Contrary to our expectations, since the density of node and the number of nodes lying in the intersection of the two neighborhoods decreases in RBS, the precision of synchronization will decrease. Timing-sync protocol for sensor networks is less robust since the failure or mistake at the root node may cause a catastrophic failure in the synchronization process. RBCS performs node synchronization in a distributed fashion and does not make use of an overlay spanning tree to direct the pair-wise synchronizations. As shown in Fig. 5, RBCS is more robust than either RBS or TPSN.

Scalability: To compare the scalability of RBS, TPSN and RBCS, we add the percentage of nodes in the simulated network from 10, 20, to 30. The initial number of nodes is 400. According to Fig. 6, synchronization error of compared protocols increases linearly with the percentage of increased nodes. The performance of RBCS is the best. As we pointed out earlier, RBCS have two different signals for pair-wise synchronization. Therefore, the whole synchronization process of RBCS is better than RBS.

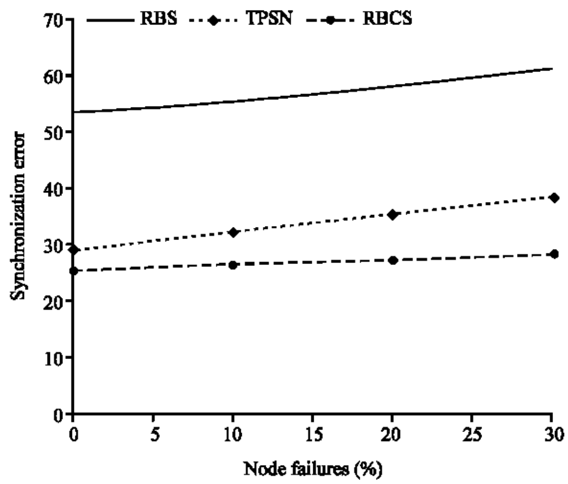

Fig. 6: Synchronization error with increased nodes

CONCLUSION AND FUTURE WORK

Clock synchronization and ranging are supporting technology in wireless sensor networks. This study proposes range-based clock synchronization protocol by analyzing the internal relationship between time and distance, which can eliminate the restriction of clock synchronization for ranging. It can achieve parallel processing for ranging and clock synchronization. The simulations show that the algorithm has superiority over RBS and TPSN as regards the number of messages exchange and synchronization accuracy.

The RBCS algorithm expands the scope of ranging application. We are applying it to other fields. Ranging error has direct impact on synchronization error, so the subsequent work is to further research the relationship between ranging error and synchronization error.

ACKNOWLEDGMENTS

This research is supported by National Natural Science Foundation of China with Grant No. 60673061 and The National Key Technology R and D Program of China with Grant No. 2007BAK23B03.

REFERENCES

Bernhard, H.W., L. Herbert and C. James, 1992. Global Positioning System Theory and Practice. Wien, Springer, New York.

Duda, A., G. Harrus, Y. Haddad and G. Bernard, 1987. Estimating global time in distributed systems. Proceedings of the 7th International Conference on Distributed Computing Systems, Berlin, Germany, Sept. 1987, IEEE Computer Society Press, pp: 299-306.

Elson, J. and K. Romer, 2003. Wireless sensor networks: A new regime for clock synchronization. ACM SIGCOMM Comput. Commun. Rev., 33: 149-154.

Ganeriwal, S., R. Kumar and M.B. Srivastava, 2003. Timing-sync protocol for sensor networks. Proceedings of the 1st International Conference on Embedded Networked Sensor Systems, Nov. 5-7, New York, USA., pp: 138-149.

Greunen, J.V. and J. Rabaey, 2003. Lightweight time synchronization for sensor networks. Proceedings of the 2nd ACM International Workshop on Wireless Sensor Networks and Applications, 2003, ACM, New York, USA., pp: 11-19.

Lamport, L., 1978. Time, clocks and the ordering of events in a distributed system. Commun. ACM., 21: 558-565.

Mailaender, L., 2007. Comparing geo-location bounds for TOA, TDOA and round-trip toa. Proceedings of the IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications, Athens, Sept. 3-7, IEEE Computer Society Press, pp: 1-5.

Maróti, M., B. Kusy, G. Simon and A. Ledeczi, 2004. The flooding time synchronization protocol. Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems, Baltimore, MD, USA., Nov. 03-05, ACM, New York, USA., pp: 39-49.

Mills, D., 1992. Network time protocol (version 3): Specification, implementation and analysis. DARPA Network working group report RFC 1305. http://www.faqs.org/ftp/rfc/rfc1305.pdf.

Noh, K.L., E. Serpedin and K. Qaraqe, 2008. A new approach for time synchronization in wireless sensor networks: Pairwise broadcast synchronization. IEEE Trans. Wireless Commun., 7: 3318-3322.

PalChaudhuri, S., A.K. Saha and D.B. Johnson, 2004. Adaptive clock synchronization in sensor networks. Proceedings of the 3rd International Symposium on Information Processing in Sensor Networks, Berkeley, California, USA., Apr. 26-27, ACM, New York, USA., pp: 340-348.

Sivrikaya, F. and B. Yener, 2004. Clock synchronization in sensor networks: A survey. IEEE Network Mag. Special Issue Ad Hoc Network. Data Commun. Topol. Contr., 18: 45-50.

Son, H.C., J.G. Lee and G.I. Jee, 2003. Mobile station location using hybrid GPS and a wireless network. Proceedings of the 57th IEEE Vehicular Technology Conference, Apr. 22-25, IEEE Computer Society Press, pp: 2716-2720.

Sundararaman, B., U. Buy and A.D. Kshemkalyani, 2005. Clock synchronization for wireless sensor networks: A survey. Ad hoc Networks, 3: 281-323.