A Simple Decentralized Timeslot Synchronization Algorithm for Large-Scale Wireless IoT Networks

Gurusanthosh Pabbisetty
Wireless System Laboratory
Information and Communication Platform Laboratories
Corporate Research & Development Center
Toshiba Corporation
gurusanthosh.pabbisetty@toshiba.co.jp

Hiroki Mori
Wireless System Laboratory
Information and Communication Platform Laboratories
Corporate Research & Development Center
Toshiba Corporation
hiroki2.mori@toshiba.co.jp

***Abstract*—Rapid growth in use of wireless IoT sensors to realize smart lighting / homes / factories would add a precious asset to the existing infrastructure i.e. numerous wireless devices, which are used to transfer sensed parameters. Various applications (such as indoor localization) could be realized with this asset if it is possible to collect wireless parameters such as RSSI among all IoT sensors. An efficient way is to use token-based TDMA mechanisms where every sensor gets a transmission opportunity in the assigned timeslot. But, synchronizing timeslots to identify transmission instant in a large-scale sensor network is a challenging task. In this paper, we propose a simple decentralized timeslot synchronization algorithm that could realize any TDMA based RSSI accumulation protocol in an ultra-efficient way compared to conventional schemes and evaluated using an event based simulator developed in Matlab.**

Keywords—Decentralized algorithm, Large-scale IoT, RSSI data accumulation, Token-based TDMA

I. Introduction

It is expected that total number of Internet of Things (IoT) connections will reach 83 billion by 2024 [1], resulting in a predominant increase in use of IoT devices and plays a crucial role in realizing smart lighting / homes / factories / cities. It is trivial to embed a wireless device with technologies like Bluetooth® / Zigbee / 802.15.4 into IoT sensors to transmit sensed parameters to a nearby access point. On the other hand, pandemics like coronavirus changed the dynamics of society and brought new challenges such as maintaining social distance, formation of crowd clusters and driving mankind in the direction of realizing touchless world. Although it is a social obligation to maintain social distancing, smart infrastructure realized with numerous IoT devices such as smart lighting could also assist in alerting the surrounding users.

Leveraging existing infrastructure with numerous low bandwidth wireless devices could help in detecting location of people / robots with localization algorithms. Various types of localization algorithms are proposed based on Received Signal Strength Indicator (RSSI), Time of Arrival, Time Difference of Arrival, and Angle of Arrival [2]. Readily available parameter with low cost wireless devices in smart infrastructure is RSSI. Although the localization accuracy achieved with RSSI is coarse-grained, accuracy could be enhanced with numerous wireless IoT devices by collecting RSSI among multiple IoT devices in multiple low bandwidth channels. With existing infrastructure, RSSI based localization is plausible and initial step is to efficiently collect RSSI among all IoT devices in multiple low bandwidth channels. One of the RSSI accumulation protocols is multi-spin [3]. It is a multi-channel token based TDMA protocol that defines order of transmission for sensors and collectively switch multiple low-bandwidth channels in a round-robin way. Each sensor transmits its beacon in assigned timeslot and includes measured RSSI while receiving beacons from other surrounding sensors. Even though the order of transmission is known to sensors, identifying transmission instant of each sensor plays a crucial role in realizing multi-spin. In a small-scale network, a central controller could help in identifying transmission instants and realizes time slot synchronization of all sensors. Realization of central controller in a large-scale network is practically difficult and there is a need of decentralized time slot synchronization mechanisms.

The basic concept of decentralized schemes is to derive transmission instant from ongoing beacon transmissions by sensors in synchronization (or) initiate a new synchronization instant if there are no surrounding sensors in synchronization and a similar scheme is proposed in [4]. The issue in a large-scale IoT network is possibility of initiating multiple simultaneous synchronization instants by different sensors unknowingly, resulting in multiple groups of sensors. If multi-spin is used for RSSI accumulation, every group accumulates RSSI among sensors in that particular group, resulting in partial RSSI data accumulation. In this work, we propose a novel scheme for decentralized time slot synchronization to avoid multiple synchronization instants initiated by different sensors by mapping a unique criteria specific to each sensor to initiate a new synchronization instant. Rest of the paper is organized as follows. An overview of proposed algorithm is explained in Section II. In section III, developed simulator is summarized to compare proposed algorithm with conventional algorithm inspired from [4] along with evaluation results, followed by concluding remarks in section IV.

II. Simple Decentralized Timeslot Synchronization Algorithm

Before dwelling into the details of proposed algorithm, a reference algorithm would help in a better understanding of proposed decentralized timeslot synchronization algorithm. Here, multi-spin [3] is the considered reference algorithm

ISBN 978-3-903176-35-5 © IFIP

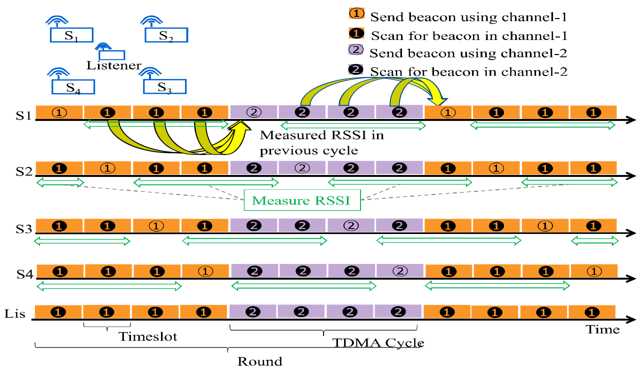

Fig. 1. Multi-spin with 4 sensors and 2 channels

A. Overview of Multi-spin [3]

Multi-spin [3] is a token based TDMA protocol that could be used to accumulate RSSI among sensors in different channels. Every sensor transmits beacon in the assigned timeslot and remaining sensors measure RSSI while receiving beacon. The transmitted beacon includes measured RSSI while receiving beacons from remaining sensors. A specific node named "Listener" is used to sniff the beacons and accumulate RSSI among all sensors.

An example of multi-spin is shown in Fig. 1. Network contains 4 sensors and a listener. It is assumed that all sensors are aware of timeslot duration, timeslot sequence, number of sensors in the network, and channel map to be used. Let us assume that the timeslot sequence used by S1, S2, S3, and S4 as 1, 2, 3, and 4 respectively, indicating that beacons triggered by sensors in a sequential round-robin manner as S1, S2, S3, S4, S1, and so on. Initially sensor S1 transmits an empty beacon without any RSSI values as no beacons are received by S1 at this time instant. Remaining sensors (S2, S3, S4) receives beacon and measures RSSI. In a similar way, remaining sensors transmit beacons including measured RSSI values in the assigned timeslot sequences. Listener sniffs the beacons and accumulates RSSI among sensors from received beacons. After 4 timeslots, all the sensors including listener switch channel to the next channel in channel map and repeats the same process.

B. Overview of Decentralized Synchronization Algorithm

In the previous sub-section, it is assumed that all sensors were synchronized to a reference time instant and could deduce the instant at which each sensor has to initiate beacon. There are 2 types of beacons namely: sync beacon and async beacon. A transmitted beacon is termed as "sync beacon" when the sensor is aware of reference time instant and a flag is used inside beacon contents to indicate the same. If a beacon is transmitted by a sensor that is not aware of reference time instant, it is termed as "async beacon".

Overview of proposed decentralized synchronization algorithm is shown in flow chart as in Fig. 2. For a simpler understanding, assumptions of multi-spin [3] are carried over here i.e. all the sensors are aware of total sensors in the network, timeslot duration, timeslot sequence number, channel map. Additionally, each sensor has a unique parameter named "max async beacons". This parameter is specifically used when a sensor is not aware of reference time instant. When all sensors in the network are not aware of reference time instant, there is a need to initiate reference time instant by any of sensors. This parameter "max async beacons" is used to decide the sensor that has to initiate reference time instant. Typically, if this parameter is set as a same number in all the sensors, there is a possibility of having two or more reference time instants, resulting in a partial RSSI data collection. Hence, we propose to have a unique value of this parameter to each sensor in the network, resulting in a single reference time instant even if the network is a large-scale wireless network. Max async beacon could be a function of sensor's timeslot sequence, sensor id or any function that makes this parameter as unique.

As shown in step-1 of Fig. 2., all sensors initialize above mentioned parameters to get reference time instant. In step-2, all the sensors initialize parameter "present async beacons" count to zero, indicating no async beacon is received. In step-3, all sensors generate a random time interval, followed by waiting for a beacon in the first channel of channel map (as in

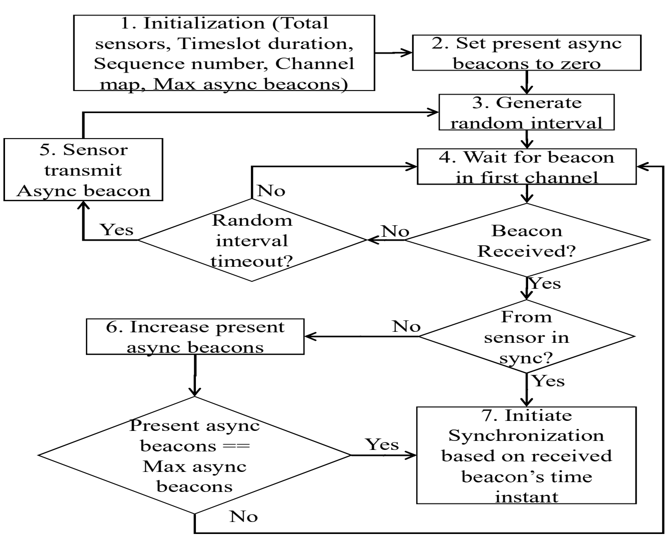

Fig. 2. Overview of proposed decentralized synchronization algorithm

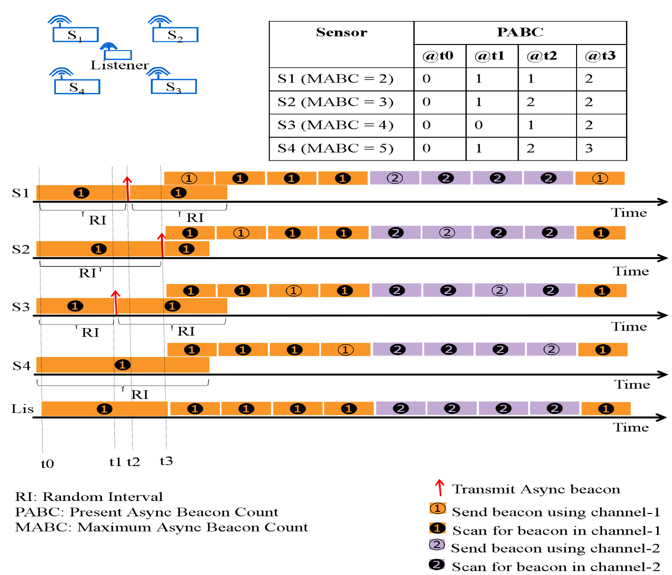

Fig. 3. Example of proposed distributed synchronization algorithm

step-4). While waiting for a beacon, if generated random interval is timed out without obtaining reference time instant, sensor transmits async beacon (as in step-5) and regenerates random time interval and awaits at first channel. While waiting for a beacon, if an async beacon is received, sensor would increase “present async beacons” parameter (as in step-6), and checks if the count has reached “max async beacons” of sensor. If the count is not reached, it awaits for next beacon. If the count is reached, sensor understands that there are no reference time instants in the network and initiates a reference time instant (as in step-7) by triggering sync beacon as per data accumulation protocol i.e. multi-spin. If a sync beacon is received, sensor understands that there is an existing reference time instant and deduce its transmission instant based on received sync beacon (as in step-7).

Let us consider an example network of 4 sensors and 2 channels in channel map same as the one taken for explaining multi-spin. Before starting multi-spin, all the sensors have to find a reference time instant in a distributed way as per proposed algorithm. Let us assume the parameter Max Async Beacon Count (MABC) for sensors S1, S2, S3, and S4 as 2, 3, 4, and 5 respectively. Initially, Present Async Beacon Count (PABC) of all sensors is zero. All sensors generate a random interval. In this example, timeout of random interval for S3 is generated at time instant t1 and hence, an async beacon is generated by S3 followed by regenerating random interval. At t1, count of PABC is increased for all sensors except S3. Subsequently, time out event is generated for S1 at t2, followed by time out event generated at t3 for S2. At this time instant t3, PABC of S1 is equal to MABC of S1. Therefore, S1 initiates reference time instant and transmit sync beacon at t3. After receiving sync beacon at t3 by remaining sensors, they deduce the time instant to transmit sync beacon based on reference time instant as per multi-spin [3].

This scheme could be applied even if a new sensor is added to the on-going beacon transmissions. New sensor would wait for any type of beacon. As soon as the sync beacon is received, new sensor derives the reference time instant from the received sync beacon and starts transmitting sync beacons in its assigned timeslot.

III. Algorithm Evaluation

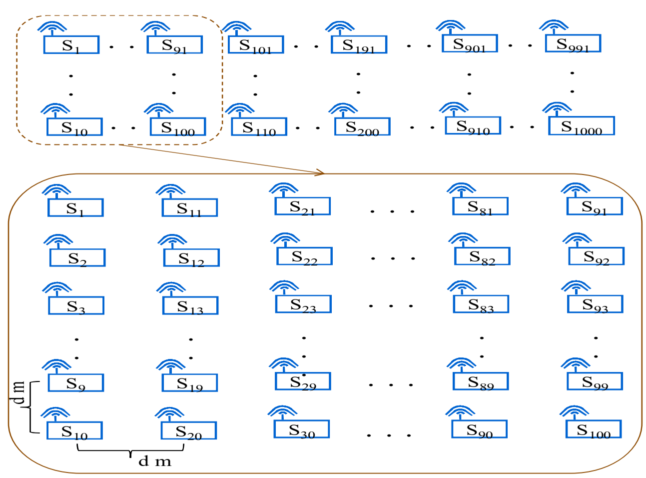

Fig. 4. Evaluation Topology

To evaluate the summarized algorithm, event based simulator is developed using Matlab. Aim of the simulator is to grasp number of different reference time instants generated in a large-scale wireless network and estimate the time required to transit all the sensors from asynchronization to synchronization i.e. all the sensors grasping the reference time instant. Accordingly, channel model among the nodes is considered as Free Space Path Loss (FSPL) [5] with frequency as 2450 MHz and path loss component as 2. Transmit power of all sensors is assumed as -10 dBm and in order to limit the coverage area of each sensor, -80 dBm is considered as threshold received power i.e. if the received power of sensors is less than -80 dBm, those sensors are considered as out of coverage area of transmitted sensor. The random variable assumed for random number generator is uniform random variable with a mean of number of sensors times timeslot duration in multi-spin [3] (Assumed timeslot duration is 5 milliseconds).

Evaluation topology is as shown in Fig. 4. Number of sensors are varied from 100 to 1000 with an increment of 100 sensors and distance between adjacent sensors is set as 5 m (typical distance between adjacent lighting sensors in smart lighting of a factory).

In this evaluation, two algorithms are compared. One is the proposed algorithm with unique maximum async beacon count for each sensor (5 times timeslot sequence number) and other is the conventional algorithm inspired from [4] in which maximum async beacon count for all sensors is same as one. Considered performance metrics are percentage of sensors entered synchronization and time taken to transit from asynchronization to synchronization. If only one reference time instant is initiated, it implies that all sensors are synchronized to a single reference time instant. In other words, 100% of sensors are in synchronization. If two or more reference time instants are initiated, network is categorized into group of sensors based on reference time instant. In this scenario, percentage of sensors entered synchronization is mapped to reference time instant with maximum number of sensors. Each sample of output is generated after executing with 100000 different types of random seeds, followed by taking mean of metrics.

Percentage of sensors in synchronization vs number of sensors is shown in Fig.5 (left y-axis). Even though number of wireless sensors are increased, all are synchronized to a

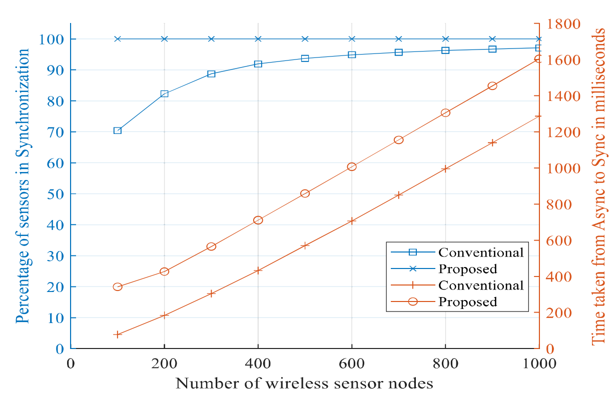

Fig. 5. Percentage of nodes in synchronization and time taken for all nodes to transit from async to sync vs number of sensors

ISBN 978-3-903176-35-5 © IFIP

#Sensors	Algorithm	#Reference time instants			
		1	2	3	4
100	Conventional	15%	50%	34%	1%
100	Proposed	100%	0%	0%	0%
1000	Conventional	5%	85%	10%	0%
1000	Proposed	100%	0%	0%	0%

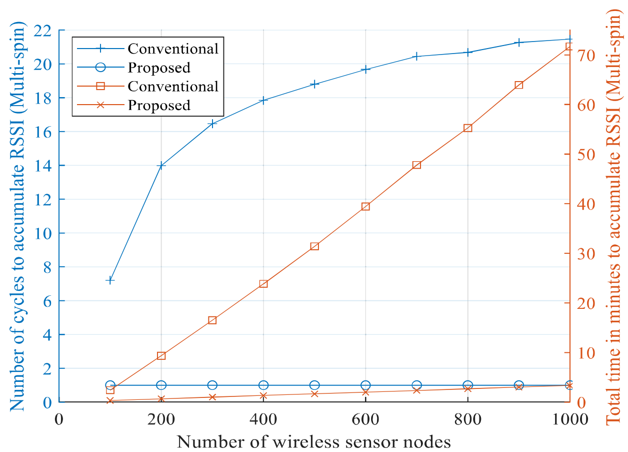

Fig. 6. Number of cycles and total time to accumulate RSSI using multi-spin vs number of sensors

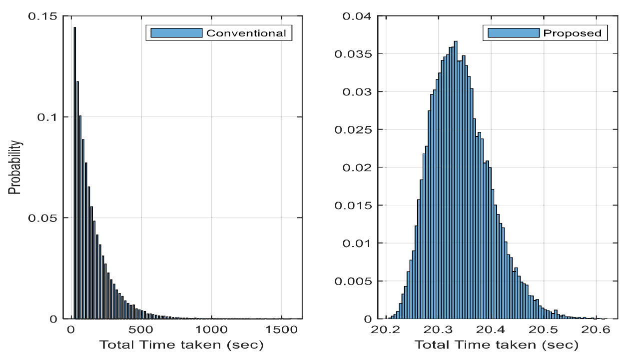

Fig. 7. Distribution of total time incurred for multi-spin with 100 sensors

single reference time instant with proposed algorithm. In conventional algorithm, multiple reference time instants are generated and the reference time instant with maximum number of synchronized sensor nodes is shown in Fig.5 (left y-axis). Number of generated reference time instants are shown in Table 1 for number of wireless sensors as 100 and 1000. With 100000 different types of random seeds, it is observed that generated reference time instants with conventional algorithm is not one in 85% and 95% of cases with number of sensors as 100 and 1000 respectively.

Time taken for all the sensors to enter synchronization from a synchronization vs number of sensors is shown in Fig.5 (right y-axis). In a typical scenario, as the number of sensors increase, time taken for all the sensors usually increases. As expected, time taken to enter synchronization for proposed algorithm is greater than conventional algorithm. But, in conventional algorithm, all sensors couldn't deduce a single time reference instant resulting in an additional post processing work to accumulate overall RSSI data among all nodes and all channels in the channel map. To grasp the overall time required to accumulate RSSI among all sensors in multiple channels, same protocol is repeated multiple times until the accumulation is completed. In this scenario, it is assumed that number of channels in channel map as 40 and timeslot duration as 5 milliseconds. To accumulate RSSI data in 40 channels, number of times the protocol is repeated (number of cycles) vs number of wireless sensor nodes is shown in Fig.6 (left y-axis) and total time to accumulate RSSI among all nodes including time taken for all nodes to enter synchronization from asynchronization is shown in Fig.6 (right y-axis). As the number of reference time instants generated in proposed algorithm is only one, time taken to accumulate RSSI among all nodes in multiple channels is quite minimal. On the other hand, as multiple reference time instants are generated, RSSI data accumulation continues until a single reference time instant is generated with other random seeds. Accordingly, number of cycles incurred is quite higher, implying the total time to accumulate RSSI increasing in a rapid way. It is observed that even with 100 number of sensors, proposed algorithm is 5 times faster than conventional one and with 1000 number of sensors, proposed algorithm is around 20 times faster than conventional algorithm. Probability distribution of total time incurred to implement multi-spin [3] with 100 number of sensors is shown in Fig.7. Conventional has more variations than proposed algorithm.

IV. Conculsion

In this work, we proposed a simple decentralized timeslot synchronization algorithm and compared it with conventional algorithm inspired from [4]. Proposed algorithm could help in deducting a single reference time instant even in a large-scale wireless IoT sensor network. On the other hand, time taken to enter synchronization is a bit higher than conventional algorithm. But, there is a clear edge over conventional algorithm with the overall time taken to accumulate RSSI data (such as multi-spin [3]) among all sensors in all channels. One of our future research directions is to identify a way to accumulate RSSI data in a large scale wireless IoT network.

References

[1] Juniper Research Whitepaper, "IOT ~ The Internet of Transformation 2020," 2020. Available: https://www.juniperresearch.com/document-library/white-papers/iot-the-internet-of-transformation-2020, Accessed 27th Nov 2020.

[2] R. C. Shit, S. Sharma, D. Puthal, and A. Y. Zomaya, "Location of Things (LoT): A review and taxonomy of sensors localization in IoT infrastructure," IEEE Comm. Sur. & Tut., vol. 20, pp. 2028-2061, January 2018.

[3] M. Bocca, O. Kaltiokallio, and N. Patwari, "Radio tomographic imaging for ambient assisted living," in Proc. EvAAL, 2013. Available: https://my.ece.utah.edu/~npatwari/pubs/bocca-evaal-book-chapter.pdf, Accessed 27th Nov 2020

[4] K. D. Colling, P. R. Carpenter, L. Koleszar, P. E. Schmidt, "Distributed ad hoc network protocol using synchronous shared beacon signaling," U.S. Patent 8385322B2, 2008. Available: https://patents.google.com/patent/US8385322B2/en, Accessed 27th Nov 2020.

[5] M. Schwartz, Mobile Wireless Communications, New York, NY, USA: Cambridge University Press, 2004.