

# Estimation of Relative Offset and Drift for Synchronization of Local Clocks in Wireless Sensor Networks

Ayako Arao and Hiroaki Higaki  
Department of Robotics and Mechatronics,  
Tokyo Denki University, Japan  
Email: {arao,hig}@higlab.net

**Abstract**—In wireless sensor networks, each wireless sensor node records events that occurred in its observation area with their observation time. Each wireless sensor node possesses its own local clock whose drift and offset are generally different from the others. In addition, it is difficult for the wireless sensor nodes to adjust drifts and offsets of their local clocks since transmission delay of messages between neighbor wireless sensor nodes are difficult to estimate due to Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) and Request to Send/Clear to Send (RTS/CTS) control for collision avoidance. Thus, it is difficult to achieve orders and intervals among events observed by different wireless sensor nodes. Moreover, even if multiple wireless sensor nodes observe the same event, their recorded observation times might be different and two observation records for the same event by two wireless sensor nodes are not always recognized as the records for the same event. Based on an assumption that observation areas of neighbor wireless sensor nodes are overlapped, by using observation records of the commonly observed events by neighbor wireless sensor nodes, this paper proposes a novel method to estimate the relative drift and offset between local clocks of the neighbor wireless sensor nodes. Here, each sensor node only detects the occurrences of events and cannot achieve the locations where the events occur. Hence, commonly observed events between neighbor wireless sensor nodes are required to be detected. Our proposed method applies a heuristic that multiple observation records in neighbor wireless sensor nodes whose intervals are the same are estimated to be commonly observed events.

**Keywords**—Wireless Sensor Networks; Observation Time; Local Clock Synchronization; Relative Drift Estimation; Relative Offset Estimation.

## I. INTRODUCTION

A wireless sensor network consists of a great number of wireless sensor nodes with their sensor modules for achieving environmental data and wireless communication modules for transmission of data messages containing the environmental data to one of stationary sink nodes by using wireless multi-hop communication based on wireless ad-hoc communication. Each wireless sensor node possesses its local clock and the sensor node records observed events with the clock value at that time. Since the wireless sensor nodes work autonomously and their local clocks have individual differences, it is almost impossible for the local clocks in the wireless sensor nodes to be completely synchronized [4]. Especially due to individual differences in their crystal oscillators, incremented clock values in the same time duration are generally different one by one and networks with numerous number of nodes with their local clocks should be designed and managed on the assumption of the asynchronous local clocks [8]. Similar to [10], this paper assumes that a local clock value  $C_i(t)$  of a wireless sensor node  $S_i$  is represented with its offset  $O_i$  and drift  $dt_i/dt$  as  $C_i(t) = (dt_i/dt)t + O_i$ . Since each local clock of  $S_i$  has its

own offset and drift, it is expected that a clock value difference  $|C_i(t) - C_j(t)|$  between local clocks of  $S_i$  and  $S_j$  is required to be kept small by a certain clock synchronization procedure with a certain short interval. In addition, local clock values recorded when a wireless sensor node observes events are also required to be corrected according to the clock synchronization procedure.

In environments where Global Positioning System (GPS) or wave clocks are not available, relative offset and drift between two local clocks of wireless sensor nodes are required to be estimated. Various conventional methods for clock synchronization in wired networks have been proposed. Here, control messages carrying local clock values are exchanged among wired nodes and transmission delay for the messages are estimated for clock synchronization as in Figure 1. However, in wireless networks, due to collision avoidance methods such as CSMA/CA and RTS/CTS [1] control in wireless Local Area Network (LAN) protocols, dispersion of transmission delay of the control messages carrying local clock values is large and it becomes difficult to achieve precise synchronization of local clocks based on estimation of relative offset and drift between the local clocks of neighbor wireless sensor nodes. Hence, this paper proposes a novel clock synchronization method without control message transmissions with local clock values whose transmission delay is difficult to estimate. Our proposed method is based on the fact that observation areas of neighbor wireless sensor nodes are usually overlapped and events which occurs in the overlapped area are observed by the wireless sensor nodes simultaneously.

In Section 2, we review related works. In Section 3, we propose our clock synchronization method based on records of observed events in neighbor wireless sensor nodes. Its performance is evaluated in simulation experiments and the results are discussed in Section 4. Finally, we conclude in Section 5.

## II. RELATED WORKS

The problem of synchronization among local clocks in a network has been discussed and various synchronization methods have been proposed. The most fundamental approach to solve the problem is the algorithm discussed in [2]. Here, between two computers, local clock value request and reply control messages are exchanged where these control messages carry local clock values of sender computers (Figure 1). However, since the receiver computer cannot achieve its local clock values when the received control message is transmitted, the transmission delay of the received control message is required to be estimated. Therefore, the methods for clock synchronization by exchange of local clock values require more precise estimation of transmission delay of control messages.

Even with variation of transmission delay of control messages, it may be practically applicable for proposed methods to wired networks whose variation of transmission delay is not so large.

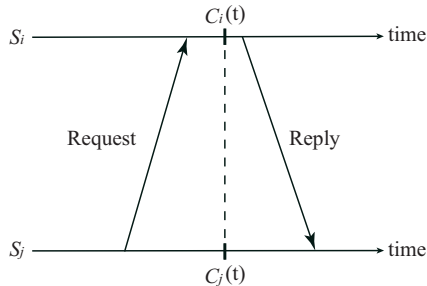


Figure 1. Clock Synchronization by Exchange of Control Messages with Local Clock Values.

For synchronization of local clocks of wireless nodes in wireless ad-hoc networks, Reference-Broadcast Synchronization (RBS) [3], Flooding Time Synchronization Protocol (FTSP) [5] and Timing-sync Protocol for Sensor Networks (TSPN) [7] have been proposed. All these methods are based on the transmissions of control messages carrying local clock values as discussed before. Hence, for achieving highly precise synchronization among local clocks in wireless nodes, more precise estimation of transmission delay of control messages carrying local clock values are required. However, due to collision avoidance methods such as CSMA/CA and RTS/CTS control, it becomes much more difficult to estimate transmission delay of control messages for clock synchronization. The backoff timer for collision avoidance in CSMA/CA introduces unpredictable waiting time for data message transmissions and RTS/CTS control for avoiding collisions due to the hidden terminal problem requires much longer suspension of data message transmission procedure causing much higher unpredictability of total transmission delay as shown in Figure 2.

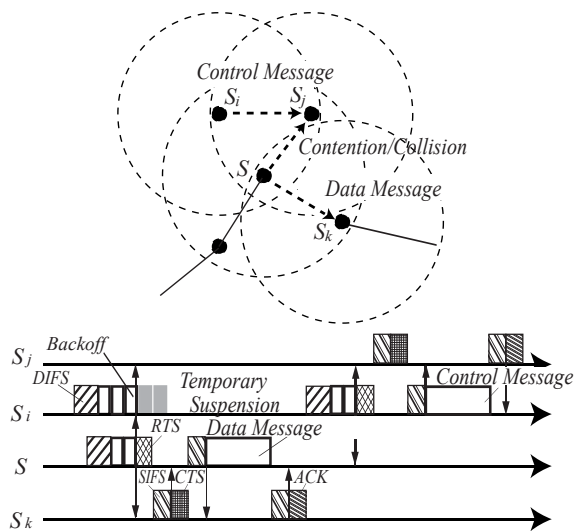


Figure 2. Unpredictable Transmission Delay of Control Messages for Clock Synchronization in Wireless Ad-Hoc Networks.

Especially in wireless sensor networks, high congestions of sensor data messages around the stationary wireless sink nodes are unavoidable so that prediction of transmission delay of control messages becomes difficult or almost impossible. In addition, burst traffic of data messages caused by some critical events also makes difficult to estimate transmission delay of control messages. In order to solve this problem, another approach without transmissions of control messages to which current local clock values are piggybacked are required to be considered.

### III. PROPOSAL

#### A. Commonly Observed Events

Each wireless sensor node consists of a sensor module which detects events that occurred within its observation area and a wireless communication module which transmits/receives wireless signals from/to its neighbor wireless nodes within its wireless signal transition area. A wireless sensor node  $S_i$  which detects an occurrence of an event within its observation area records kinds of the events with some additional related attributes including the clock value  $C_i(t)$  of its local clock at the instance  $t$  when  $S_i$  observes the event. For simplicity, this paper assumes that each event is detected by all the wireless sensor nodes whose observation areas include the location of the event at that instance, i.e. without any observation delay. In reality, each sensor device requires its specific response time for an event observation and the effect of the delay is discussed in our future work. In addition, all events are assumed to be the same kind. Hence, in accordance with the event observation records by a wireless sensor node  $S_i$ , a sequence  $ESeq_i := \langle C_i(t_0), C_i(t_1), \dots, C_i(t_{N_i}) \rangle$  of the clock values at the instances when  $S_i$  observes the events is induced. Here,  $C_i(t_j)$  is the value of the local clock of  $S_i$  at the instance  $t_j$  when  $S_i$  observes an occurrence of an event  $e_i(t_j)$  in its observation area. On the other hand, each wireless sensor node  $S_i$  communicates with its neighbor wireless sensor nodes within its wireless signal transmission area. Thus, it is possible for  $S_i$  to exchange its clock value sequence  $ESeq_i$  at occurrences of locally observed events with its neighbor wireless sensor nodes.

Generally, the observation area of a wireless sensor node is included in its wireless signal transmission area. In addition, in a wireless sensor network, an observation area where all the event occurred are surely observed and recorded by at least one wireless sensor node is required to be covered by observation areas of multiple wireless sensor nodes as shown in Figure 3 [6] [9]. Hence, observation areas of neighbor wireless sensor nodes usually overlap and the wireless sensor nodes whose observation area overlap can communicate directly by using wireless ad-hoc communication.

Suppose the case where observation areas of wireless sensor nodes  $S_i$  and  $S_j$  overlap as shown in Figure 4. As mentioned,  $S_i$  and  $S_j$  can communicate directly by wireless ad-hoc communication since they are included in their wireless transmission areas one another. Here, all the events occurred in the overlapped observation area are observed by both  $S_i$  and  $S_j$  and recorded with clock values of their own local clocks. These events are called *commonly observed events* of  $S_i$  and  $S_j$ . The other events, i.e. events observed by only one of  $S_i$  and  $S_j$ , are called *solely observed events*.

### [Commonly/Solely Observed Events]

An event which occurs at a certain instance  $t$  in an overlapped area of observation areas  $OA_i$  and  $OA_j$  of wireless sensor nodes  $S_i$  and  $S_j$  respectively and is observed and recorded with local clock values  $C_i(t)$  and  $C_j(t)$  into clock value sequences  $ESeq_i$  and  $Eseq_j$  by  $S_i$  and  $S_j$  respectively is called a commonly observed event of  $S_i$  and  $S_j$ . On the other hand, an event which occurs at a certain instance  $t$  in an area included by  $OA_i$  and excluded by  $OA_j$  and is observed and recorded with a clock value  $C_i(t)$  into only a clock value sequence  $ESeq_i$  by  $S_i$  is called a solely observed event of  $S_i$  against  $S_j$ . □

Each wireless sensor node  $S_i$  assumes to observe all the events occur within an observation area  $OA_i$  of  $S_i$ . As various widely available sensor modules,  $S_i$  only identifies the occurrence of the events and gets the clock values of its local clock at the instance of the occurrence of the events; however, it cannot identify the precise locations of the events in its observation area. Hence, it is impossible for  $S_i$  to identify whether an observed event is a commonly observed event with a neighbor wireless sensor node  $S_j$  or a solely observed event against  $S_j$ . Even though clock values at an instance when an event occurs are recorded by wireless sensor nodes which observe the event, since clock values  $C_i(t)$  and  $C_j(t)$  of wireless sensor nodes  $S_i$  and  $S_j$  at any instance  $t$  are generally different, it is impossible for a wireless sensor node to identify its commonly observed events with a specified neighbor wireless sensor nodes only by comparison of local clock values in their clock value sequences as shown in Figure 4. Since clock values  $C_i(t)$  and  $C_j(t)$  of  $S_i$  and  $S_j$  for a commonly observed event at an instance  $t$  are different and it is impossible to identify commonly observed events of  $S_i$  and  $S_j$  only by simply comparing the sequences of clock values.

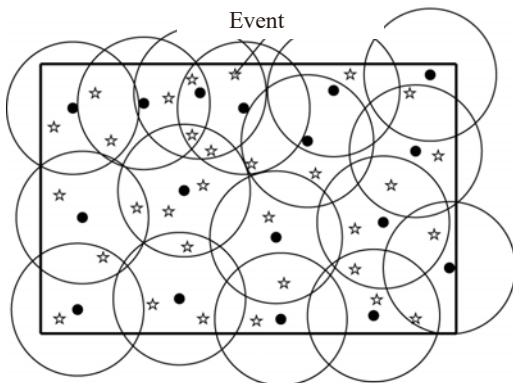


Figure 3. Whole Coverage of Observation Area by Overlap Observation Areas of All Sensor Nodes.

### B. Relative Offset Estimation

By using commonly observed events defined in the previous subsection, this paper proposes a method to estimate a relative drift  $dt_j/dt_i = (dt_j/dt)/(dt_i/dt)$  and a relative offset  $O_j - O_i$  under an assumption that local clock values  $C_i(t)$  and  $C_j(t)$  of wireless sensor nodes  $S_i$  and  $S_j$  are given as  $C_i(t) = (dt_i/dt)t + O_i$  and  $C_j(t) = (dt_j/dt)t + O_j$ , respectively. This subsection discusses a method to estimate only a relative offset where a relative drift is assumed to be

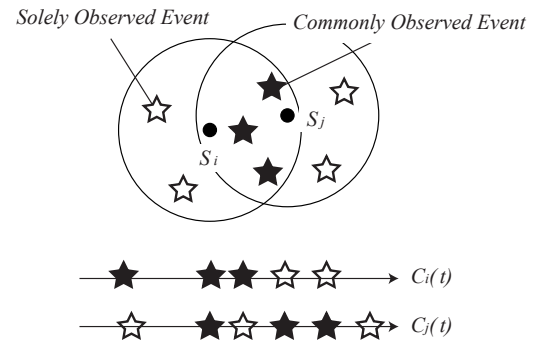


Figure 4. Local Clock Values of Observation Time in  $S_i$  and  $S_j$ .

1. The method to estimate both a relative drift and a relative offset is discussed in the next subsection.

In case that a relative drift of  $C_i(t)$  and  $C_j(t)$  is 1, i.e.  $dt_j/dt_i = 1$ ,  $C_j(t) - C_i(t) = O_j - O_i$ , i.e. a difference between clock values at any instance equals to their relative offset. Hence, if one of pairs of clock values of commonly observed events is identified, the difference between the clock values is their relative offset. However, it is difficult to identify a pair of clock values of a commonly observed event from local clock value sequences of neighbor wireless sensor node. This is because, as discussed in the previous section, even if wireless sensor nodes  $S_i$  and  $S_j$  observe the same event, i.e. their commonly observed event, at an instance  $t$ , their local clock values  $C_i(t)$  and  $C_j(t)$  at  $t$  are usually different, i.e.  $C_i(t) \neq C_j(t)$ . In addition, even if the instances  $t$  and  $t'$  of solely observed events observed by  $S_i$  and  $S_j$  respectively are different, i.e.  $t \neq t'$ , their local clock values  $C_i(t)$  and  $C_j(t')$  might be the same, i.e.  $C_i(t) = C_j(t')$ . Hence, the simple comparison between individual clock values  $C_i(t)$  and  $C_j(t')$  recorded in sequences  $ESeq_i$  and  $ESeq_j$  of local clock values of  $S_i$  and  $S_j$  does not result in correct estimation of the relative offset between their local clocks.

In order to solve this problem, this paper proposes a novel method to estimate the relative offset and drift between the local clocks of neighbor wireless sensor nodes by using multiple pairs of clock values recorded in the sequences of local clock values. As discussed, a clock value sequence  $ESeq_i$  of local clock values of a wireless sensor node  $S_i$  when it observes events in its observation area  $OA_i$  includes local clock values of commonly observed events with its neighbor wireless node  $S_j$ . Though local clock values of  $S_j$  for the same commonly observed events are surely included in a clock value sequence  $ESeq_j$  of local clock values of  $S_j$  when it observes them, it is impossible to detect the commonly observed events by simple comparison of local clock values in  $ESeq_i$  and  $ESeq_j$ . However, since the commonly observed events, i.e. events which occurs in the overlapped area of observation areas  $OA_i$  and  $OA_i$  of  $S_i$  and  $S_j$ , are observed at the same instance  $t$  by  $S_i$  and  $S_j$  even though  $C_i(t)$  and  $C_j(t)$  may be different, intervals between the same pair of commonly observed events in  $S_i$  and  $S_j$  are the same. That is, suppose that clock values of  $S_i$  and  $S_j$  when they observe two commonly observed events occur at instances  $t$  and  $t'$  are  $C_i(t)$ ,  $C_i(t')$ ,  $C_j(t)$ ,  $C_j(t')$ , respectively. Even if  $C_i(t) \neq C_j(t)$  and  $C_i(t') \neq C_j(t')$ ,

$C_i(t') - C_i(t) = C_j(t') - C_j(t)$  is surely satisfied.

Since both locations where events occur and intervals between successive events contain a certain randomness, i.e. a certain unpredictability, this paper introduces a heuristic based on a reversed proposition of the above one into estimation of commonly observed events. Thus, if there exist local clock values  $C_i(t_1)$  and  $C_i(t_2)$  in  $ESeq_i$  of  $S_i$  and  $C_j(t_3)$  and  $C_j(t_4)$  in  $ESeq_j$  of  $S_j$  and  $C_i(t_2) - C_i(t_1) = C_j(t_4) - C_j(t_3)$  is satisfied though  $C_i(t_1) \neq C_j(t_3)$  and  $C_i(t_2) \neq C_j(t_4)$ , it is highly possible for  $S_i$  and  $S_j$  to have been observed two same events, i.e. there are two commonly observed events occurred at  $t_1 = t_3$  and  $t_2 = t_4$  respectively in the overlapped area of their observation areas. Needless to say, it might be possible for solely observed events whose recorded clock values are  $C_i(t_1)$ ,  $C_i(t_2)$ ,  $C_j(t_3)$  and  $C_j(t_4)$  to satisfy  $C_i(t_2) - C_i(t_1) = C_j(t_4) - C_j(t_3)$  on accident. Hence, our heuristic method regards the possible relative offset that provides the maximum number of estimated commonly observed events which satisfies the above condition as an estimated relative offset.

#### [Estimation of Relative Offset]

Let  $ESeq_i$  and  $ESeq_j$  be sequences of local clock values  $C_i(t)$  and  $C_j(t)$  at instances when wireless sensor nodes  $S_i$  and  $S_j$  observe events. An estimated relative offset is what provides the maximum number of estimated commonly observed events where the transformed clock values with the estimated relative offset are the same. That is, with the estimated relative offset  $O$ , if the number of pairs of local clock values satisfying  $C_i(t) + O = C_j(t')$  where  $C_i(t) \in ESeq_i$  and  $C_j(t') \in ESeq_j$  is the maximum for all possible relative offsets,  $O$  is regarded as the estimated relative offset for  $S_i$  and  $S_j$ .  $\square$

For example, Figure 5(a) shows two sequences of local clock values  $ESeq_i$  and  $ESeq_j$ . Figures 5(b), 5(c) and 5(d) show the results of parallel translation of  $ESeq_j$  with possible relative offsets, i.e. where a pair of a local clock value  $C_i(t)$  and a transformed local clock value with a possible relative offset  $C_j(t') + O$  become the same value. There are 1, 2 and 3 estimated commonly observed events with the same transformed local clock values. If the maximum number of estimated commonly observed events is 3, the relative offset in Figure 5(c) is the estimation result in our method.

Now, we design an algorithm for estimation of a relative offset based on the heuristics. Here, for every pair of local clock values  $C_i(t_k^i)$  and  $C_j(t_l^j)$  in  $ESeq_i$  and  $ESeq_j$  of  $S_i$  and  $S_j$  respectively, it is assumed that these local clock values represent those at a certain commonly observed event, that is the difference  $O = C_j(t_l^j) - C_i(t_k^i)$  is regarded as the estimated relative offset of the local clocks of  $S_i$  and  $S_j$ , and the number of estimated commonly observed events where  $C_j(t_l^j) = C_i(t_k^i) + O$  is satisfied is counted. Here, the possible related offset is between the maximum  $C_i(t_{N_i}^i) - C_j(t_0^j)$  and the minimum  $C_i(t_0^i) - C_j(t_{N_j}^j)$  and the algorithm counts the estimated commonly observed events for every possible relative offset in this range. If there is a certain upper limit of relative offset between the local clocks of  $S_i$  and  $S_j$ , it is possible for the proposed algorithm to work with this limitation to reduce the time duration required for the proposed algorithm.

#### [Relative Offset Estimation Algorithm]

- 1) Initialize the maximum number of estimated commonly observed events of wireless sensor nodes  $S_i$  and  $S_j$  as 0 by  $MCO_{iv} := 0$ .
- 2) A temporary relative offset and the number of estimated commonly observed events are initialized as  $Soff_{iv} := C_i(t_{N_i}^i) - C_j(t_0^j)$  and  $CO_{ij} := 0$ .
- 3) For each local clock value  $C_i(t_k^i) \in ESeq_i = |C_i(t_0^i), C_i(t_1^i), \dots, C_i(t_{N_i}^i)|$ , search events  $C_j(t_l^j) \in ESeq_j = |C_j(t_0^j), C_j(t_1^j), \dots, C_j(t_{N_j}^j)|$  satisfying  $C_i(t_k^i) = C_j(t_l^j) + Soff_{ij}$  and increments  $CO_{ij}$ .
- 4) If  $CO_{ij} \geq MCO_{ij}$ ,  $MCO_{ij} := CO_{ij}$  and an estimated relative offset  $Eoff_{ij} := Soff_{ij}$ .
- 5) If  $Soff_{ij} = C_j(t_{N_j}^j) - C_i(t_0^i)$ , jump to step 8).
- 6) Search a relative offset update  $Uoff_{ij} := \min(C_j(t_l^j) + Soff_{ij} - C_i(t_k^i))$  where  $C_j(t_l^j) + Soff_{ij} - C_i(t_k^i) > 0$ .
- 7)  $Soff_{ij} := Soff_{ij} - Uoff_{ij}$  and  $CO_{ij} := 0$ . Then, jump to step 3).
- 8) Return  $Eoff_{ij}$  as the required estimated relative offset and the algorithm terminates.  $\square$

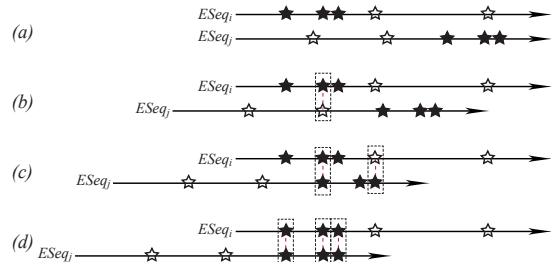


Figure 5. Estimation of Relative Offset.

#### C. Relative Drift Estimation

This subsection proposes an extended algorithm for estimation of both the relative offset and the relative drift for recorded local clock values in two neighbor wireless sensor nodes whose observation areas overlap. Figure 6 shows the overview of our proposed method. Same as the method proposed in the previous subsection which supports only the cases with 1 relative drift, the number of estimated commonly observed events between local clock value sequences  $ESeq_i$  and  $ESeq_j$  for every possible relative offset  $C_i(t_k^i) - C_j(t_l^j)$ . In addition, for estimation of the relative drift, another pair of local clock values  $C_i(t_{k'}^i) \in ESeq_i$  and  $C_j(t_{l'}^j) \in ESeq_j$  ( $k \neq k'$  and  $l \neq l'$ ) is needed. Here, an estimated relative drift is  $(C_i(t_{k'}^i) - C_i(t_k^i)) / (C_j(t_{l'}^j) - C_j(t_l^j))$ . After applying the transformation of local clock values with the estimated relative offset and the estimated relative drift, the number of estimated commonly observed events whose local clock values are the same is evaluated. Same as the previous subsection, according to a heuristic that the correct pair of relative offset and relative drift provides the maximum number of estimated commonly observed events, our proposed method estimate them. In order to apply our proposed method, for neighbor wireless sensor

nodes to estimate relative offset and drifts to transform the local clock values for synchronization, there should be more than 3 commonly observed events. Hence, enough observation period to record local clock values are required.

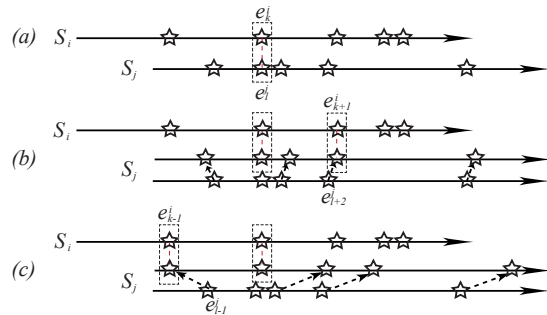


Figure 6. Estimation of Relative Drift..

Figure 7 shows a case of correct estimation of commonly observed events with correct estimation of a relative drift  $dt_j/dt_i$  and a relative offset  $O_j - O_i$ . Here, pairs of local clock values  $C_i(t_1^i)$  and  $C_j(t_1^j)$ ,  $C_i(t_2^i)$  and  $C_i(t_3^i)$ , and  $C_i(t_3^i)$  and  $C_j(t_4^j)$  are those for commonly observed events, i.e.,  $t_1^i = t_1^j$ ,  $t_2^i = t_3^j$  and  $t_3^i = t_4^j$ , respectively, and the rest  $C_i(t_4^i)$  and  $C_j(t_2^j)$  are local clock values for solely observed events in  $S_i$  and  $S_j$ , respectively. By consideration that  $C_i(t_1^i)$  and  $C_j(t_1^j)$  are local clock values in  $S_i$  and  $S_j$  when a commonly observed events of  $S_i$  and  $S_j$  occurs, the relative offset is estimated as  $O_j - O_i = C_j(t_1^j) - C_i(t_1^i)$  and the line representing the local clock value in  $S_j$  is parallelly displaced as the points representing the local clock values  $C_i(t_1^i)$  and  $C_j(t_1^j)$  of the commonly observed event are overlapped. Then, by consideration that  $C_i(t_2^i)$  and  $C_j(t_3^j)$  are local clock values in  $S_i$  and  $S_j$  when a commonly observed events of  $S_i$  and  $S_j$  occurs, the relative drift is estimated as  $dt_j/dt_i = (C_j(t_3^j) - C_j(t_1^j))/(C_i(t_2^i) - C_i(t_1^i))$  and the line representing the local clock value in  $S_j$  is rotated around the point representing the local clock value  $C_i(t_1^i)$  as the points representing the local clock values  $C_i(t_2^i)$  and  $C_j(t_3^j)$  of the commonly observed event are overlapped. Now, the lines representing the local clock values of  $S_i$  and  $S_j$  are overlapped and all the commonly observed events including that for  $C_i(t_3^i)$  and  $C_j(t_4^j)$  are correctly estimated.

On the other hand, Figures 8 and 9 show the cases when estimation of relative drift and/or offset is incorrect and estimation of commonly observed events is also incorrect as a result. In Figure 8, same as in Figure 7,  $C_i(t_1^i)$  and  $C_j(t_1^j)$  are considered to be local clock values in  $S_i$  and  $S_j$  when a commonly observed events of  $S_i$  and  $S_j$  occurs, and the relative offset is correctly estimated as  $O_j - O_i = C_j(t_1^j) - C_i(t_1^i)$  and the line representing the local clock value in  $S_j$  is parallelly displaced as the points representing the local clock values  $C_i(t_1^i)$  and  $C_j(t_1^j)$  of the commonly observed event are overlapped. However, by incorrect consideration that  $C_i(t_2^i)$  and  $C_j(t_4^j)$  are local clock values in  $S_i$  and  $S_j$  when a commonly observed events of  $S_i$  and  $S_j$  occurs, the relative drift is incorrectly estimated as  $dt_j/dt_i = (C_j(t_4^j) - C_j(t_1^j))/(C_i(t_2^i) - C_j(t_1^j))$  and the line representing the local clock value in  $S_j$  is rotated around the point representing the local clock value  $C_i(t_1^i)$  as

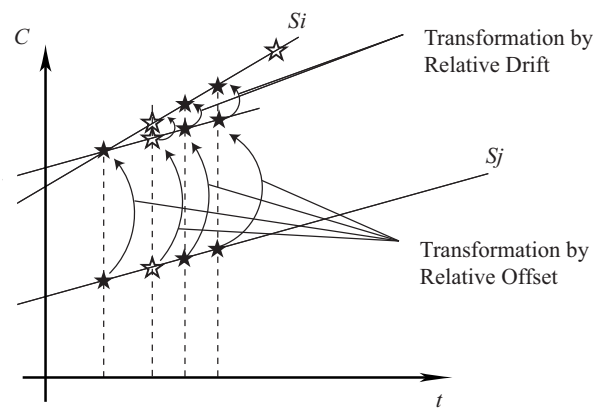


Figure 7. Estimation of Commonly Observed Events by Offset and Drift Estimation (Correct).

the points representing the local clock value  $C_j(t_4^j)$  has the same  $C$  value (vertical axis) as  $C_i(t_2^i)$ . Here, pairs of points on the two lines representing the local clock values in  $S_i$  and  $S_j$  with the same  $C$  value (vertical axis) correspond to a commonly observed event of  $S_i$  and  $S_j$ . However, in Figure 8, though pairs of  $C_i(t_2^i)$  and  $C_j(t_3^j)$ , and  $C_i(t_3^i)$  and  $C_j(t_4^j)$  are those of local clock values for commonly observed events, their  $C$  values are not the same, i.e., these pairs of local clock values are not estimated to be those for commonly observed events.

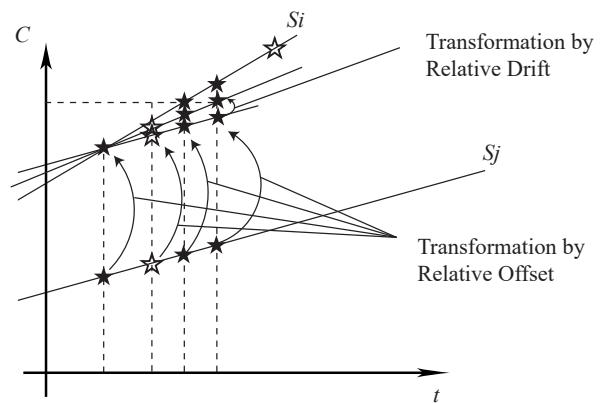


Figure 8. Estimation of Commonly Observed Events by Offset and Drift Estimation (Incorrect Drift).

Moreover, in Figure 9, both relative offset and drift are incorrectly estimated. Here,  $C_i(t_1^i)$  and  $C_j(t_2^j)$  which is local clock value in  $S_j$  when its solely observed event occurs are considered to be local clock values in  $S_i$  and  $S_j$  when a commonly observed event of  $S_i$  and  $S_j$  occurs. A relative offset is incorrectly estimated as  $O_j - O_i = C_j(t_2^j) - C_i(t_1^i)$  and the line representing the local clock value in  $S_i$  is parallelly displaced as the points representing the local clock values  $C_i(t_1^i)$  and  $C_j(t_2^j)$  have the same  $C$  value (vertical axis). Then,  $C_i(t_2^i)$  and  $C_j(t_4^j)$  are considered to be local clock values of the commonly observed event of  $S_i$  and  $S_j$ , that is, the relative drift is also incorrectly estimated as  $De_j/dt_i = (C_j(t_4^j) - C_j(t_1^j))/(C_i(t_2^i) - C_i(t_1^i))$ , and the line

representing the local clock value in  $S_j$  is rotated around the point representing  $C_j(t_1^j)$  which has already displaced from the original position as the points representing the local clock value  $C_j(t_4^j)$  has the same  $C$  value (vertical axis) as  $C_i(t_2^i)$ . Here, pairs of points on the two lines representing the local clock values in  $S_i$  and  $S_j$  with the same  $C$  value (vertical axis) correspond to a commonly observed event of  $S_i$  and  $S_j$ . In Figure 9, no correct pairs of local clock values in  $S_i$  and  $S_j$  are estimated to be those of commonly observed events and two pairs of local clock values in  $S_i$  and  $S_j$  are incorrectly estimated to be those of commonly observed events.

As shown in these three examples in Figures 7, 8 and 9, the number of estimated commonly observed events with incorrect estimation of relative offset and drift is usually smaller than that with correct estimation of them. It may be possible for pairs of local clock values of different events to be estimated as those of commonly observed events since the transformed  $C$  values are coincidentally the same. However, since the probability of such coincidental cases is low, the proposed heuristic that the correct relative drift and offset provides the maximum number of estimated commonly observed events is almost always applicable.

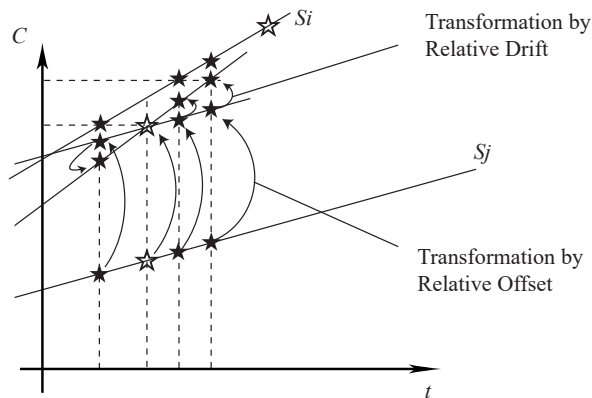


Figure 9. Estimation of Commonly Observed Events by Offset and Drift Estimation (Incorrect Offset and Drift).

#### [Relative Offset and Draft Estimation Algorithm]

- 1) Initialize the maximum number of estimated commonly observed events of wireless sensor nodes  $S_i$  and  $S_j$  as 0 by  $MCO_{iv} := 0$ .
- 2) A temporary relative offset is initialized as  $Soff_{iv} := C_i(t_{N_i}^i) - C_j(t_0^j)$ .
- 3) For every possible temporary relative drift  $Sdri_{iv} := (C_i(t_{k'}^i) - C_i(t_k^i)) / (C_j(t_{l'}^j) - C_j(t_l^j)) > 0$ , apply the following steps 4), 5) and 6).
- 4) The number of estimated commonly observed events is initialized as  $CO_{ij} := 0$ .
- 5) For each local clock value  $C_i(t_k^i) \in ESeq_i = |C_i(t_0^i), C_i(t_1^i), \dots, C_i(t_{N_i}^i)|$ , search events  $C_j(t_l^j) \in ESeq_j = |C_j(t_0^j), C_j(t_1^j), \dots, C_j(t_{N_j}^j)|$  satisfying  $(C_i(t_{k''}^i) - C_i(t_k^i)) / (C_j(t_{l''}^j) - C_j(t_l^j)) = Sdri_{ij}$  and increments  $CO_{ij}$ .

- 6) If  $CO_{ij} \geq MCO_{ij}$ ,  $MCO_{ij} := CO_{ij}$ , an estimated relative offset  $Eoff_{ij} := Soff_{ij}$  and an estimated relative drift  $Edri_{ij} := Sdri_{ij}$ .
- 7) If  $Soff_{ij} = C_j(t_{N_j}^j) - C_i(t_0^i)$ , jump to step 10).
- 8) Search a relative offset update  $Uoff_{ij} := \min(C_j(t_l^j) + Soff_{ij} - C_i(t_k^i))$  where  $C_j(t_l^j) + Soff_{ij} - C_i(t_k^i) > 0$ .
- 9)  $Soff_{ij} := Soff_{ij} - Uoff_{ij}$  and  $CO_{ij} := 0$ . Then, jump to step 3).
- 10) Return  $Eoff_{ij}$  and  $Edri_{ij}$  as the required estimated relative offset and the required estimated relative drift and the algorithm terminates.  $\square$

Figure 6 shows an example. According to the method proposed in the previous subsection, a pair of local clock values  $C_i(t_k^i)$  and  $C_j(t_l^j)$  is assumed to be for a possible commonly observed events. In addition, another pair of local clock values are also assumed to be for another possible commonly observed events and all the local clock values are transformed according to parallel translation. Then, the number of estimated commonly observed events with the same transformed local clock values are assigned is counted and the relative offset and drift that provide the maximum number of estimated commonly observed events is regarded as the correct ones.

#### IV. EVALUATION

Precision of our proposed method depends on the number of commonly observed events of neighbor wireless sensor nodes. Form this point of view, this section evaluates the performance of our proposed method by simulation experiments. Suppose two stationary wireless sensor nodes with 10m observation ranges are located with their distance 0.5–19.5m. Locations of events and intervals of two successive events are randomly determined according to unique distribution and exponential distribution, respectively. That is, events occur according to Position arrivals. With various event density, the ratio of correct estimation of commonly observed events, i.e. the ratio of correct estimation of relative offset and drift of their local clocks, is evaluated.

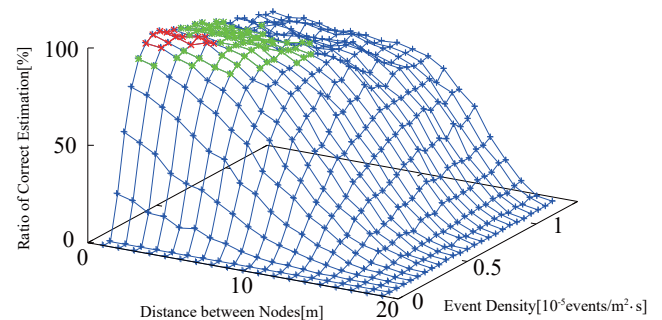


Figure 10. Ratio of Correct Estimation of Commonly Observed Events.

Figure 10 shows the simulation results. Red points represent correct estimation ratio higher than 99%, green points represent correct estimation ratio higher than 90%, and blue points represent others. Except for cases with extremely low event density and with extremely narrow overlapped observation

area, our proposed method provides high correct estimation ratio. The performance is independent of the wireless transmission traffic of sensor data messages, e.g. around stationary wireless sink nodes, which is the most important advantage against the conventional method in which precise estimation of transmission delay of control messages are required.

## V. CONCLUSION

This paper has proposed a novel clock synchronization method for wireless sensor networks. Different from the conventional methods by exchanging control messages with current local clock values and by estimation of transmission delay of the control messages, the proposed method estimates the relative offset and drift between two local clocks of neighbor wireless sensor nodes based on records of local clock values of event observations and estimation of commonly observed events of them. This paper has also designed estimation algorithms of relative offset and drift and evaluated their performance.

## REFERENCES

- [1] "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," Standard IEEE 802.11 (2016).
- [2] Cristian, F., "Probabilistic Clock Synchronization," *Distributed Computing*, Springer, vol. 3, no. 3, pp. 146–158 (1989).
- [3] Jeremy, E., Lewis, G. and Deborah, E., "Fine-Grained Network Time Synchronization using Reference Broadcasts," *Proceedings of the 5th Symposium on Operating Systems Design and Implementation*, pp. 147–163 (2002).
- [4] Kopetz, H. and Ochseneiter, W., "Clock Synchronization in Distributed Real-Time Systems," *ICE Transactions on Computers*, vol. C-36, no. 8, pp. 933–940 (1987).
- [5] Miklos, M., Branislav, K. and Kayla, S., "The Flooding Time Synchronization Protocol," *Proceedings of the 2AD International Conference on Embedded Networked Sensor Systems*, pp. 39–49 (2004).
- [6] Qu, Y., and Georgakopoulos, S.V., "A Distributed Area Coverage Algorithm for Maintenance of Randomly Distributed Sensors with Adjustable Sensing Range," *ICE Global Communications Conference*, pp. 286–291 (2013).
- [7] Saurabh, G., Ram, K. and Mani, B.S., "Timing-Sync Protocol for Sensor Networks," *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems*, pp. 138–149 (2003).
- [8] Tanenbaum, A.S. and Steen, M., "Distributed Systems Principles and Paradigms," *Prentice Hall* (2002).
- [9] Tuba, E., Tuba, M., and Simian, D., "Wireless Sensor Network Coverage Problem Using Modified Fireworks Algorithm," *Wireless Communications and Mobile Computing Conference*, pp. 696–701 (2016).
- [10] Wu, Y-C., Chaudhari, Q. and Serpedin, E., "Clock Synchronization of Wireless Sensor Networks," *ICE Signal Processing Magazine*, vol. 28, no. 1, pp. 124–138 (2011).