Local Clock Synchronization Without Transmission Delay Estimation of Control Messages in Wireless Sensor Networks

26

Ayako Arao and Hiroaki Higaki

Abstract

In wireless sensor networks, each wireless sensor node records events 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 conventional clock synchronization methods, wireless sensor nodes exchanges control messages with their local clock values and estimate their transmission delay. However, it is difficult to adjust their local clocks since transmission delay of control messages are difficult to estimate. 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

26.1 Introduction

A wireless sensor network consists of numerous 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 multihop 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 [7]. 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 [3]. 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]. Same as [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 GPS (Global Positioning System) 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. However, in wireless networks, due to collision avoidance methods such as CSAM/CA and RTS/CTS control in wireless 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

A. Arao · H. Higaki (✉)
Department of Robotics and Mechatronics, Tokyo Denki University, Tokyo, Japan
e-mail: arao@higlab.net; hig@higlab.net

S. Latifi (ed.), *Information Technology – New Generations*, Advances in Intelligent Systems and Computing 738,
https://doi.org/10.1007/978-3-319-77028-4_26

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.

26.2 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 [1]. 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. 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.

For synchronization of local clocks of wireless nodes in wireless ad-hoc networks, RBS [2], FTSP [4] and TSPN [6] 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 Fig. 26.1. 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.

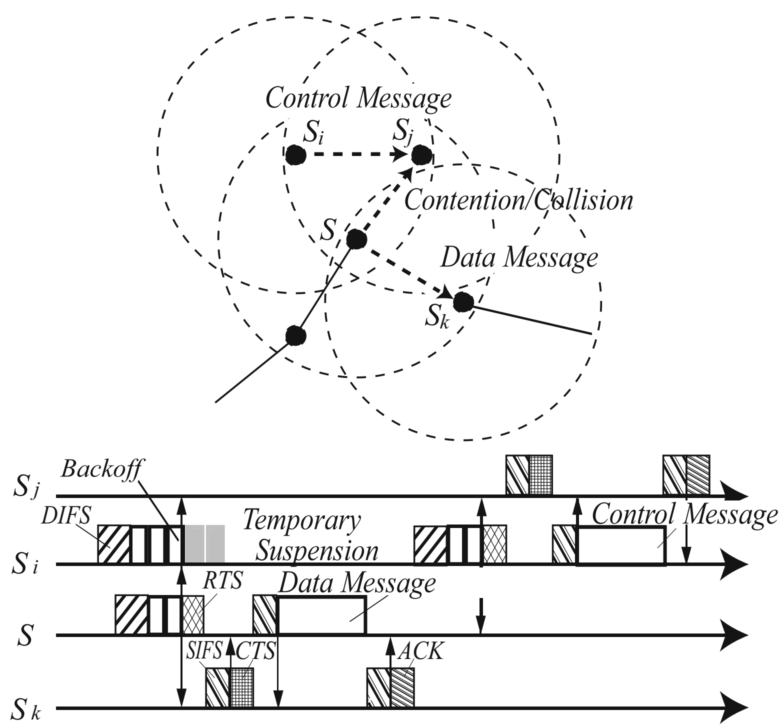

Fig. 26.1 Unpredictable transmission delay of control messages for clock synchronization in wireless ad-hoc networks

26.3 Proposal

26.3.1 Commonly Observed Events

Each wireless sensor node consists of a sensor module which detects events 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.[1] Hence, in accordance with the event observation records by a wireless sensor node S_i, a sequence $ESeq_i := |C_i(t_0), C_i(t_1), \ldots, C_i(t_{N_i})\rangle$ 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 Fig. 26.2 [5, 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 Fig. 26.3. 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*.

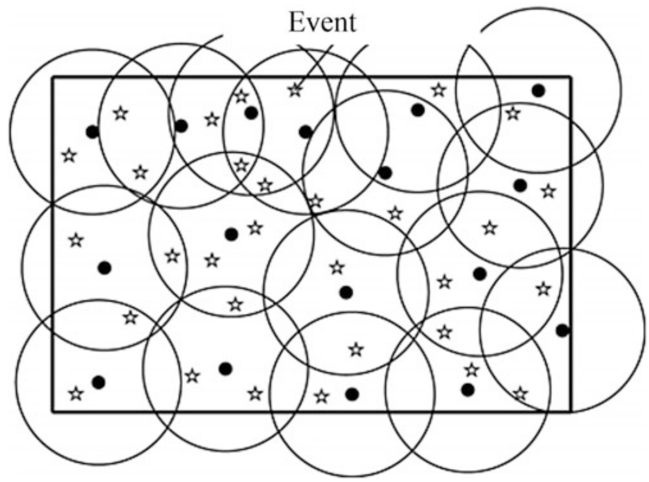

Fig. 26.2 Whole coverage of observation area by overlap observation areas of all sensor nodes

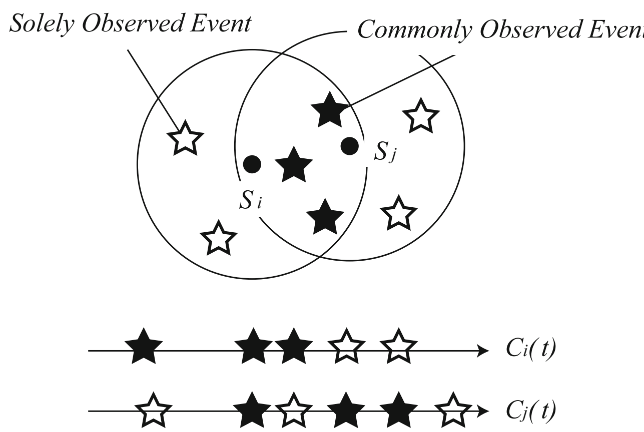

Fig. 26.3 Local clock values of observation time in S_i and S_j

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 occur-

[1] If various kinds of events are observed and identified by wireless sensor nodes, more precise estimation of commonly observed events is realized.

rence 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 Fig. 26.3. 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.

26.3.2 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 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 heuristical 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

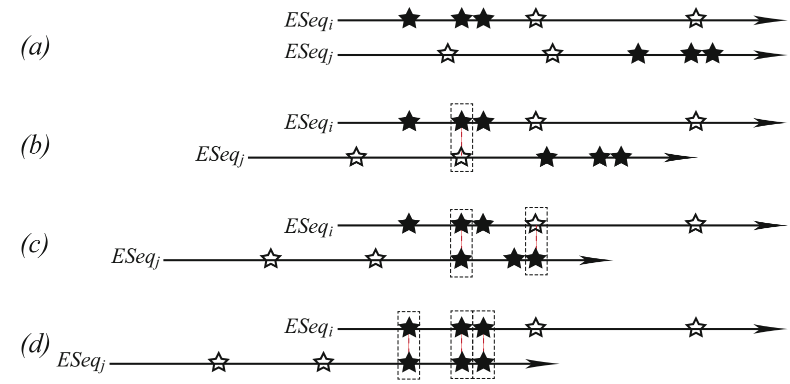

Fig. 26.4 Estimation of relative offset

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. □

For example, Fig. 26.4a shows two sequences of local clock values $ESeq_i$ and $ESeq_j$. Figure 26.4b,c and 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 one, two and three 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 Fig. 26.4c 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 $ESeq_j$ respectively, it is assumed that these local clock values represents 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 S_i and S_j, and the number of estimated commonly observed events where $C_j(t_{l'}^i) = C_i(t_{k'}^i) + O$ 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 an upper limit of relative offset between 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 $C_i(t_k^i) \in ESeq_i = |C_i(t_0^i), C_i(t_1^i), \ldots, C_i(t_{N_i}^i)\rangle$, search events $C_j(t_l^j) \in ESeq_j = |C_j(t_0^j), C_j(t_1^j), \ldots, C_j(t_{N_j}^j)\rangle$ 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 relate 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. □

26.3.3 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 26.5 shows the overview of our proposed method. Same as the method proposed in the previous subsection which supports only the cases with one 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

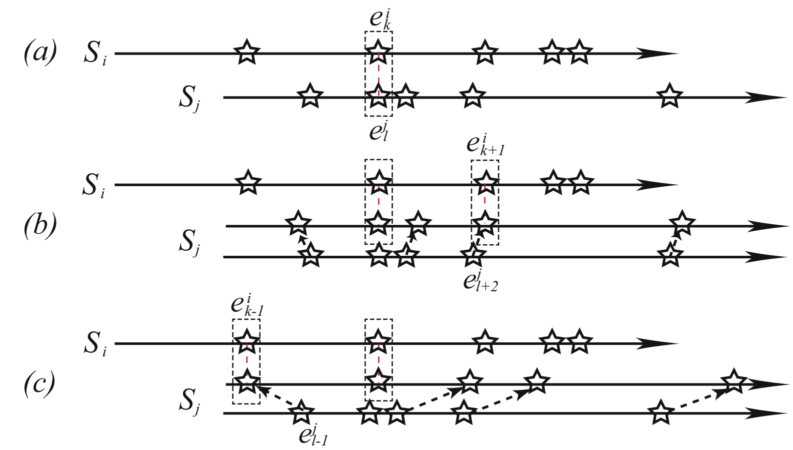

Fig. 26.5 Estimation of relative drift

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 three commonly observed events. Hence, enough observation period to record local clock values are required.

Figure 26.6 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 $C_i(t_3^j)$, 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.

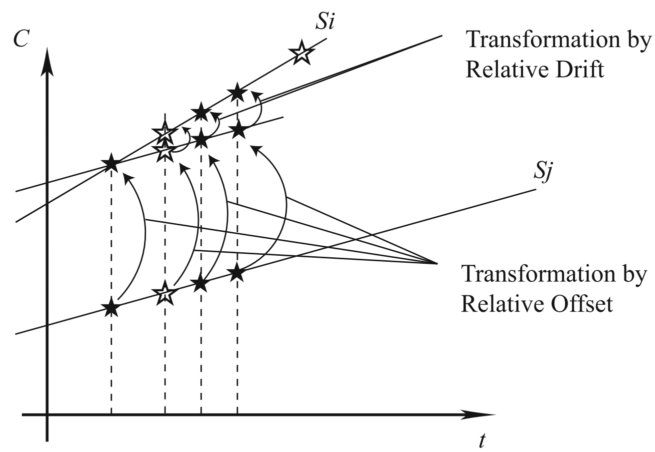

Fig. 26.6 Estimation of commonly observed events by offset and drift estimation (correct)

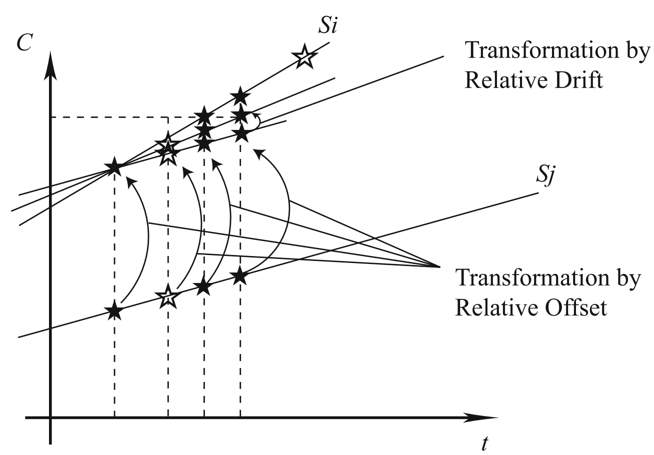

Fig. 26.7 Estimation of commonly observed events by offset and drift estimation (incorrect drift)

On the other hand, Figs. 26.7 and 26.8 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 Fig. 26.7, same as in Fig. 26.6, $C_i(t_1^i)$ 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

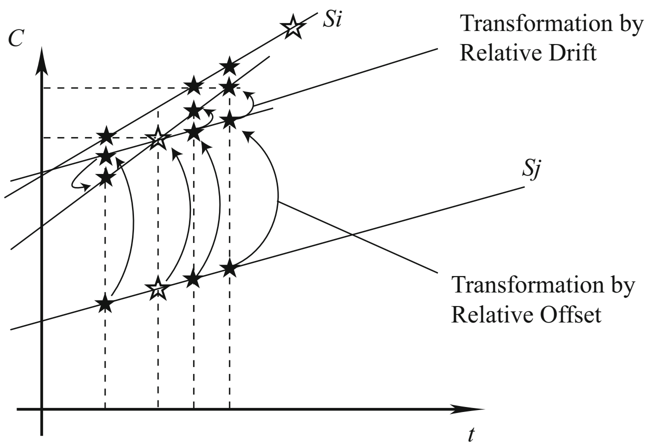

Fig. 26.8 Estimation of commonly observed events by offset and drift estimation (incorrect offset and drift)

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 $dt_j/dt_i = (C_j(t_4^j) - C_j(t_1^j))/(C_i(t_2^i) - C_j(t_1^i))$ 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 value $C_j(t_4^j)$ has the same C value (virtual 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 Fig. 26.7, 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.

Moreover, in Fig. 26.8, 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 Fig. 26.8, 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 Figs. 26.6, 26.7 and 26.8, 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.

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 $Soff_{iv} := C_i(t^i_{N_i}) - C_j(t^j_0)$.
3. For every possible temporary relative drift $Sdri_{iv} := (C_i(t^i_{k'}) - C_i(t^i_k))/(C_j(t^j_{l'}) - C_j(t^j_l)) > 0$, 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 $C_i(t^i_k) \in ESeq_i = |C_i(t^i_0), C_i(t^i_1), \ldots, C_i(t^i_{N_i})\rangle$, search events $C_j(t^j_l) \in ESeq_j = |C_j(t^j_0), C_j(t^j_1), \ldots, C_j(t^j_{N_j})\rangle$ satisfying $(C_i(t^i_{k''}) - C_i(t^i_k))/(C_j(t^j_{l''}) - C_j(t^j_l)) = Sdri_{ij}$ and increments CO_{ij}.
6. If $CO_{ij} \geq MCO_{ij}$, $MCO_{ij} := CO_{ij}$, an estimated relate offset $Eoff_{ij} := Soff_{ij}$ and an estimated relative drift $Edri_{ij} := Sdri_{ij}$.
7. If $Soff_{ij} = C_j(t^j_{N_j}) - C_i(t^i_0)$, jump to step 10).
8. Search a relative offset update $Uoff_{ij} := \min(C_j(t^j_l) + Soff_{ij} - C_i(t^i_k))$ where $C_j(t^j_l) + Soff_{ij} - C_i(t^i_k) > 0$.
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. □

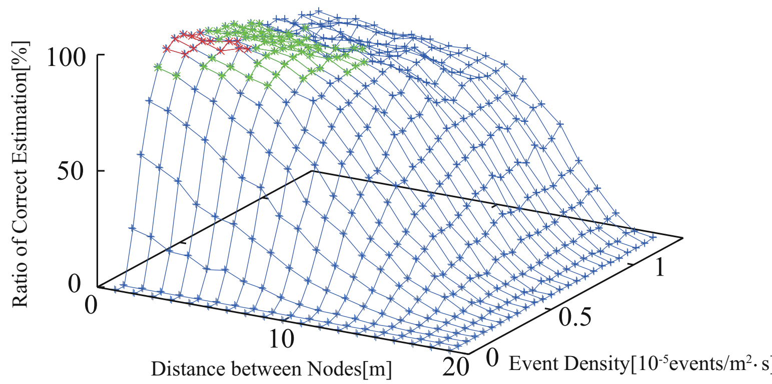

Fig. 26.9 Ratio of correct estimation of commonly observed events

Figure 26.5 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.

26.4 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 10 m observation ranges are located with their distance 0.5–19.5 m. Locations of events and intervals of two successive events are randomly determined according to the unique distribution and the exponential distribution,[2] respectively. 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 26.9 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. Less than $0.3 \times 10^{-5}/\text{m}^2\text{s}$ event occurrence density, too few commonly observed events occur. Hence, it is almost impossible to synchronize local clocks since our method requires more them three commonly observed events for the drift and offset estimation. On the other hard, the authors have been afraid that the correct estimation ratio decreases as the event density becomes higher since incorrect estimation of the commonly observed events might be caused. However, the simulation result shows that no such degradation is observed even with event density higher than $20.0 \times 10^{-5}/\text{m}^2 s$ (out of Fig. 26.9).

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.

[2]Events occur according to position arrivals.

26.5 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. F. Cristian, Progabilistic clock synchronization. Distrib. Comput. **3**(3), 146–158 (1989)
2. E. Jeremy, G. Lewis, E. Deborah, Fine-grained network time synchronization using reference broadcasts, in *Proceedings of the 5th Symposium on Operating Systems Design and Implementation* (2002), pp. 147–163
3. H. Kopetz, W. Ochsenreiter, Clock synchronization in distributed real-time systems. IEICE Trans. Commun. **C-36**(8), 933–940 (1987)

4. M. Miklos, K. Branislav, S. Kayla, The flooding time synchronization protcol, in *Proceedings of the 2AD International Conference on Embedded Networked Sensor Systems* (2004), pp. 39–49
5. Y. Qu, S.V. Georgakopoulos, A distributed area coverage algorithm for maintenance of randomly distributed sensors with adjustable sensing range, in *ICE Global Communications Conference* (2013), pp. 286–291
6. G. Saurabh, K. Ram, B.S. Mani, Timing-sync protocol for sensor networks, in *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems* (2003), pp. 138–149
7. L. Schick, W.L. Souza, A.F. Prado, Wireless body sensor network for monitoring and evaluating physical activity, in *Proceedings of the 14th ITNG* (2017)
8. A.S. Tanenbaum, M. Steen, *Distributed Systems Principles and Paradigms* (Prentice Hall, Upper Saddle River, 2002)
9. E. Tuba, M. Tuba, D. Simian, Wireless sensor network coverage problem using modified fireworks algorithm, in *Wireless Communications and Mobile Computing Conference* (2016), pp. 696–701
10. Y.-C. Wu, Q. Chaudhari, E. Serpedin, Clock synchronization of wireless sensor networks. ICE Signal Process. Mag. **28**(1), 124–138 (2011)