Improved synchronized sensing for structural health monitoring using wireless smart sensor networks*

Jian Li, Kirill A. Mechitov, Robin E. Kim, and Billie F. Spencer, Jr.

Abstract— **High cost of traditional wired structural health monitoring (SHM) systems due to onerous and expensive cabling work prohibits the number of sensors and hence limits the application and success of SHM on civil engineering structures. Wireless smart sensor networks (WSSNs) have the advantages of being low cost, more flexible and robust data management, and can provide better understanding of structural behavior through dense deployment of sensors. However, implementation of wireless SHM systems has many challenges. One of them is synchronized sensing. This issue arises in WSSNs due to the fact that each smart sensor in the network is run by its own local clock, which is not necessarily synchronized with the clocks of other sensors. Moreover, even though the clocks can be accurately synchronized by exchanging time information through beacon messages, the measured data may still be poorly synchronized due to random delays from both software and hardware sources. Various algorithms have been proposed to achieve both synchronized clocks and sensing. However, these protocols still lack the desired performance for SHM applications due to extended data collection time, temperature variation, requirement for prompt response, etc. In this paper, an improved synchronized sensing approach with two different implementations is proposed to meet various application requirements under different application scenarios. The issue of nonlinear clock drift is discussed in detail and addressed in the approach. Experimental results show that the proposed time synchronization approach .is able to compensate the temperature effect on clock drift and provide efficient and accurately synchronized sensing (<50 μs error) service for SHM even for long sensing durations.**

I. INTRODUCTION

Structural Health Monitoring is a technology that encompasses a broad range of methods and applications with the goal of informing the current condition of structures in order to assist structural maintenance. However, high cost of traditional wired SHM systems due to onerous and expensive cabling work prohibits the number of sensors and hence limits the application and success of SHM on civil structures. Wireless smart sensors (WSS) are characterized by their capabilities of sensing, computation, data transmission and storage, all achieved by a single device. The removal of cables and the mass production of MEMS sensors reduce the cost dramatically. The wireless communication capability allows flexible network topology and hence enables a decentralized monitoring scheme, which adds robustness to the SHM system compared with the centralized approach in wired systems. Wireless smart sensors have the potential to change fundamentally the way civil infrastructure systems are monitored, controlled and maintained (Spencer, et al., 2004).

Despite the various advantages, implementation of wireless SHM systems still faces many challenges, such as constrained power availability, limited communication bandwidth and range, data loss, time synchronization (TS), and so on (Lynch and Loh, 2006; Nagayama and Spencer, 2007, 2010). Among them, the time synchronization issue arises in wireless smart sensor networks due to the fact that each smart sensor in the network is run by its own local clock which is not necessarily synchronized with the clocks of other sensors. Moreover, even though the clocks are perfectly synchronized, synchronization of the collected data is not guaranteed because (1) the sensors start sensing at different time due to the randomness of the processing time in the sensor board driver, (2) the sampling frequencies among the sensor nodes are different due to the low quality of crystals, (3) the sampling frequency for each individual sensor node can fluctuate over time due to jitter (Nagayama and Spencer, 2007). A direct consequence of imperfect time synchronization in the measured data is that the phase information among the data, which is considered as an important structural performance and damage indicator, is lost (Nagayama and Spencer, 2007; Krishnamurthy et al., 2008).

Time synchronization protocols for wireless smart sensor networks have been extensively studied in the past. Most of these protocols are designed for clock synchronization only. Beacon packets with time information are exchanged among sensor nodes to allow appropriate adjustment of their clocks according to the received beacon signals. Examples include Reference Broadcast Synchronization (RBS) (Elson et al., 2003), Time-sync Protocol for Sensor Networks (TPSN) (Ganeriwal et al., 2003), and Flooding Time Synchronization Protocol (FTSP) (Maroti et al., 2004). To realize synchronization in the measured data for SHM, in addition to utilizing clock synchronization protocols to achieve synchronized clocks, Nagayama and Spencer (2007) proposed

*Resrach supported by the National Science Foundation Grants CMS 06-00433, CNS 10-35773, CPS 10-35562, and CMMI 09-28886.

Jian Li is with the Department of Civil, Environmental, and Architectural Engineering, The University of Kansas, Lawrence, KS 66045 USA (Tel: 785-864-6850, e-mail: jianli@ku.edu).

Kirill A. Mechitov is with the Department of Computer Science at the University of Illinois at Urbana-Champaign, Urbana, IL 61801 USA (e-mail: mechitov@illinois.edu).

Robin Kim is with the Department of Civil and Environmental Engineering, University of Illinois at Urbana- Champaign, Urbana, IL 61801 USA (e-mail: kim491@illinois.edu).

Billie F. Spencer, Jr. is with the Department of Civil and Environmental Engineering, University of Illinois at Urbana- Champaign, Urbana, IL 61801 USA (e-mail: bfs@illinois.edu).

a resampling-based approach to further process the data and remove the errors introduced by the three factors mentioned above. However, these protocols may still lack the desired performance for SHM applications under certain demanding circumstances such as extended data collection time, temperature variation, requirement for prompt response in SHM applications, etc.

In the followed sections, time synchronization requirements for SHM applications are discussed first. A brief survey is then performed for general time synchronization protocols widely adopted in WSSNs for clock synchronization as well as SHM-specific time synchronization strategies and related applications. The nonlinear clock drift phenomenon is presented and its effect on the accuracy of data synchronization is investigated. Accordingly, a new time synchronization approach is proposed with two different implementations to meet various application requirements. Finally the performance of the proposed time synchronization approach is validated experimentally.

II. TIME SYNCHRONIZATION REQUIREMENTS FOR SHM APPLICATIONS

As in many other WSSN applications, synchronization of the network is highly desirable in SHM applications. However, due to the specific features in SHM, such as high sampling frequency and extended sensing duration, synchronization of data is not automatically guaranteed even with accurately synchronized clocks. In this section, we discuss the SHM-specific features and the imposed challenges in time synchronization.

2.1. Synchronized clocks vs. synchronized data

One of the distinct features of SHM is that data are typically collected at high frequency. Civil engineering structures vibrate at high frequency and a sampling rate which is at least twice of the frequency of interest is needed to capture meaningful dynamic responses. In addition, responses at high frequency are more sensitive to local damage and therefore the accuracy of time synchronization at high frequency is of great importance for damage detection. For example, a 1 ms synchronization error between two measured acceleration responses will result in 3.6 degree error in the phase angle at 10 Hz and 36 degree error at 100 Hz (Nagayama and Spencer, 2007). Such errors in phase information, depending on applications, can lead to false results in damage detection.

Due to the stringent requirement of synchronization accuracy in the measured data, accurate synchronization of the clocks in a WSSN is not always adequate in SHM. Synchronized clocks do not guarantee synchronized data because of three main factors related to both software and hardware issues as summarized in Nagayama and Spencer (2007): (1) due to the randomness of the processing time in the sensor board driver, the sensors do not start sensing at exactly the same time (2) the low cost wireless smart sensors are often equipped with low quality crystals; the actual sampling frequencies among the sensor nodes are therefore different (3) the sampling frequency for each individual sensor node can also fluctuate over time due to jitter.

2.2. Extended sensing duration

Compared with other monitoring applications such as environmental monitoring, in which a single or a few data points are collected during a sensing event, sensing for structural health monitoring is characterized by much more sampled data points and therefore requires longer sensing duration (minutes or even hours). One reason is that a large number of data points are needed to extract meaningful information of structural characteristics. For example, under a given frequency bandwidth, more data points provide higher resolution once the data is converted into the frequency domain and therefore higher accuracy of estimated modal frequencies. In addition, extended sensing duration is required to fully capture a transient event such as the forced vibration due to a train crossing a bridge.

A direct impact of extended sensing duration is that the effect of clock skew becomes significant. Clock skew is a phenomenon that two clocks drift away from each other due to differential clock speed. Even though the clocks were accurately synchronized when sensing started, they can drift away from each other during sensing and cause errors in data time stamps, which in turn lead to synchronization error in the sampled data. Nagayama and Spencer (2007) tested Imote2 nodes and estimated the maximum clock drift rate among the tested nodes to be around 50 μs per second, which can lead to 20 ms synchronization error after a 400 second measurement. Appropriate clock skew compensation is necessary to eliminate or reduce such an impact.

2.3. Temperature variation during sensing

Structural health monitoring systems are typically deployed in outdoor environment where temperature can change drastically during a short period of time. For example, sensors that are in the shade at the beginning of sensing and can soon come under direct sunlight, which can heat up the sensor quickly and introduce large temperature variation during sensing period. In addition, sensing for SHM occurs at high frequency and lasts for long period of time and therefore can generate a lot of heat, resulting in temperature change on the sensor board. Li et al. (2012) performed 10-minute sensing using Imote2s and SHM-A sensor boards. Temperature readings were collected during the process. The onboard temperatures of imote2s increased by almost 6˚C due to the heat generated by the CPU and ADC chip on the SHM-A board.

Nonlinear clock drift is a direct consequence of temperature change during sensing. The clock of a wireless smart sensor is typically driven by a quartz crystal, whose resonant frequency is temperature dependent. Yang et al. (2012) showed clear correlation between temperature and clock skew and different pairs of clocks exhibit different clock skew change patterns with

respect to temperature. In Li et al. (2012), significant nonlinearity in clock drift was observed, induced by temperature change during sensing. Such nonlinear clock drift poses an additional challenge for synchronized sensing in SHM.

2.4 Need for rapid response to transient events

To conserve energy, smart sensors are often designed to spend most of their time in deep sleep mode and wake up periodically to listen for external commands. After waking up, if the network receives a command to collect data, it has to be resynchronized first before sensing. Therefore, a delay is introduced between the reception of the command and the start of sensing due to the need for resynchronization. Such a delay may cause the entire transient event such as earthquakes to be missed. Although the time needed for waking up the network also contributes to the delay, minimizing the delay due to time synchronization is critical towards capturing the entire transient structural response.

III. SURVEY OF TIME SYNCHRONIZATION IN WIRELESS SMART SENSOR NETWORKS

3.1. Clock synchronization protocols

Among the various clock synchronization protocols for WSSNs, three of them have been widely used including the Reference Broadcast Synchronization algorithm (RBS; Elson, 2003), the Time-sync Protocol for Sensor Networks (TPSN; Ganeriwal et al., 2003), and the Flooding Time Synchronization Protocol (FTSP; Maroti et al., 2004).

RBS is a receiver-receiver method. A reference node broadcasts reference messages which are time-stamped at the receivers upon reception. The receivers then exchange the recorded time with each other. This approach does not require timestamping at the sender, and so eliminates the uncertainties at the sender side. Linear regression is used to compensate for clock skew between the sensor nodes. However, it requires additional communication overhead associated with the exchange of messages between the receivers.

TPSN, on the other hand, is a sender-receiver approach. The reference messages are time-stamped at both the sender and receiver sides. Unlike the RBS which time-stamps the reference messages at the physical layer of the radio stack, TPSN time-stamps radio messages at the Medium Access Control (MAC) layer, by which the random delays associated with encoding, decoding, and interrupt handling can be eliminated. Therefore, TPSN can achieve higher accuracy than the RBS. However, TPSN does not compensate for clock skew and requires a fixed spanning tree of the network to be established before time synchronization.

FTSP is similar to TPSN in the sense that it is also a sender-receiver method and is based on MAC layer time-stamping. However, because flooding is used to disseminate reference messages, it does not need a spanning tree of network and is therefore more flexible than the TPSN and can accommodate dynamic topology change of the network. Linear regression is also used in FTSP to compensate for clock skew.

The above described methods can achieve very high accuracy in clock synchronization of wireless sensors. For example, experiments based on the Mica2 motes showed that FTSP achieved 1.5 μs TS accuracy in a single hop scenario and an average precision of 0.5 μs per hop in the multi-hop case (Maroti et al., 2004). However, as discussed in Section 2.3.1, accurately synchronized clocks do not guarantee synchronized data. Moreover, clock skew compensation in FTSP and RBS are based on the assumption that the sensor clocks drift away from each other in a linear fashion, which may not be the case in SHM applications as discussed in Section 2.3.

3.2 Time synchronization applications in SHM

Several WSSN-based SHM applications have attempted to achieve synchronization in wireless collection of vibration data. Kim et al., (2007) deployed a WSSN with 64 Mica motes on the Golden Gate Bridge in California to measure the structural response subject to ambient and extreme conditions. FTSP was adopted for synchronizing the clocks of the sensor nodes. Temporal jitter was reduced by eliminating unnecessary running of atomic sections in the software during sampling. Temporal jitter of approximately 10 μs was reported from experimental results. However, the actual synchronization error in the collected data was not reported. Wang et al. (2007) developed a WSSN prototype system for SHM. One beacon message from the central server was used to synchronize the clocks of the sensing units at the beginning of sensing. Clock drift was not compensated; therefore, even though the initial time synchronization error was 20 μs, it can accumulate up to 5 ms in a 6-minute period. Whelan et al., (2009) developed another prototype wireless SHM system using a similar time synchronization approach as Wang et al. (2007). The approach uses a single command to initiate sensing in the network and rely on accurate and stable crystal oscillators to achieve adequate time synchronization of the network for the sampling duration. Bocca et al. (2011) adopted a TS protocol called μ-sync (Mahmood and Jäntti, 2009) for wireless SHM, which is derived from FTSP. Therefore, MAC layer time-stamping and linear drift skew compensation are also utilized. The synchronization accuracy of data was not reported, but accuracy of 10 μs was observed in clock synchronization.

In the above SHM applications, the performance of the TS algorithms was reported in terms of clock synchronization, not data synchronization. Sazonov et al. (2010) proposed a hierarchical time synchronization architecture in which local cluster nodes are synchronized by beacon signals, whereas spatially distributed cluster nodes are synchronized by GPS time reference.

Synchronization accuracy was evaluated according to the phase of the sinusoidal signals sampled by the wireless sensors. Accuracy with an error less than 23 μs was achieved between the sampled sinusoidal data. However, the duration of the sinusoidal data was not reported, so it is unclear whether the same level of accuracy can be maintained if the sensing duration is larger such that temperature effect becomes significant.

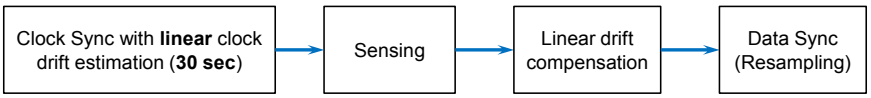


Figure 5. A two-stage TS strategy based on linear clock drift compensation (stage 1) and resampling (stage 2)

Nagayama and Spencer (2007) proposed a two-stage TS strategy which is able to achieve tight synchronization among sampled data. As illustrated in Fig. 5, in the first stage, FTSP is adapted to provide clock synchrony. Before sensing, a 30-second period is used to broadcast beacon messages for estimating the clock drift rates at the sensing units through linear regression. The linear clock drift rates are then used to correct the clocks so that the samples can be accurately time-stamped during sensing. After sensing is finished, in the second stage, the data is resampled to remove the sources of error due to the three factors discussed in Section 2.1 so as to achieve tight synchrony in the sampled data. Using Imote2 and SHM-A sensor board as the testing platform, experiment was performed by installing six Imote2 nodes on a three-dimensional truss structure under band-limited white noise (BLWN) excitation. TS accuracy was evaluated through the phase angle of the cross power spectral density (CPSD) between the sampled data. Approximately 30 μs accuracy was estimated from the experimental data. The limitation of this approach, however, is that the linear clock drift estimation in the first stage may not be able to accurately compensate the clock skew when sensing duration is large and therefore nonlinearity in clock skew becomes significant due to temperature effect. In addition, the 30-sec period for broadcasting beacon messages before sensing delays the start of sensing and is therefore undesirable in SHM as discussed in Section 2.4.

IV. Nonlinear Clock Drift in Wireless Smart Sensors

In general, the resonant frequency of clock crystal oscillators is sensitive to temperature change. When temperature is constant, the clock crystals will tick at constant frequencies; therefore, different clocks tend to drift away from each other at constant speeds over time. As a demonstration, a set of four Imote2 sensor nodes and a gateway node are programmed to exchange beacon messages to monitor the clock offsets between the gateway and the four leaf nodes over a 3500-second period. FTSP is utilized to enable MAC layer time-stamping when sending and receiving beacon packets. The gateway node is programmed to transmit beacon packets to the four receivers every 2 seconds. The beacon packet is time-stamped and the send time is attached to the beacon right before it is transmitted. Upon reception of the beacon packet, the receivers time-stamp the beacon and calculate the offset between the receive time and the send time. Once the transmission of all beacon packets is finished, the gateway node collects all offset data from the leaf nodes which are plotted in Fig. 6. No sensing was performed during the process and therefore the temperature of the leaf nodes was constant over the period. As a result, the clocks were ticking at constant speeds and were drifting away from each other in a linear fashion as shown in Fig. 6.

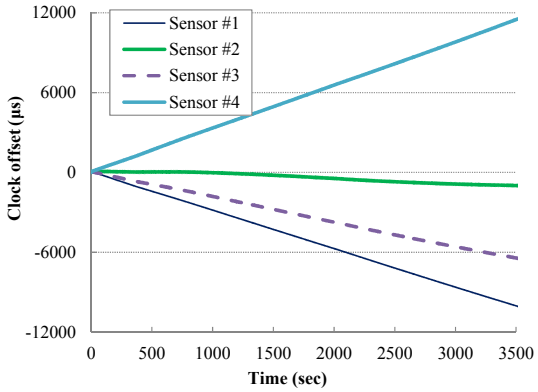


Figure 6. Linear clock drift under constant temperature

When temperature changes, the resonant frequency of clock crystal oscillators will change, leading to nonlinear clock drift. Uddin et al. (2010) investigated the clock skew of MICAz and TelosB sensor motes and showed that the variation of clock skew increases with the increase of temperature and decreases with the decrease of temperature. In addition, different motes show different relationship between clock skew and temperature. Similar behavior of clock skew under varying temperature has also been observed in Mica2 motes (Yang et al., 2012). These tests were performed under changing environmental temperature. To show the effect of temperature change on the clock skew of Imote2 nodes due to heat generated by the ADC chip, the clock drift

of a set of different four Imote2 leaf nodes was monitored with respect to a gateway node while the leaf node were performing sensing. In addition to clock offsets, the leaf nodes also record temperature readings using the on-board temperature sensor of the SHM-A sensor board. As shown in Fig. 7, the on-board temperatures of the Imote2 leaf nodes were raised by about 6℃ during the 600-second sensing period. As a result, severe nonlinearity is observed in the clock drift.

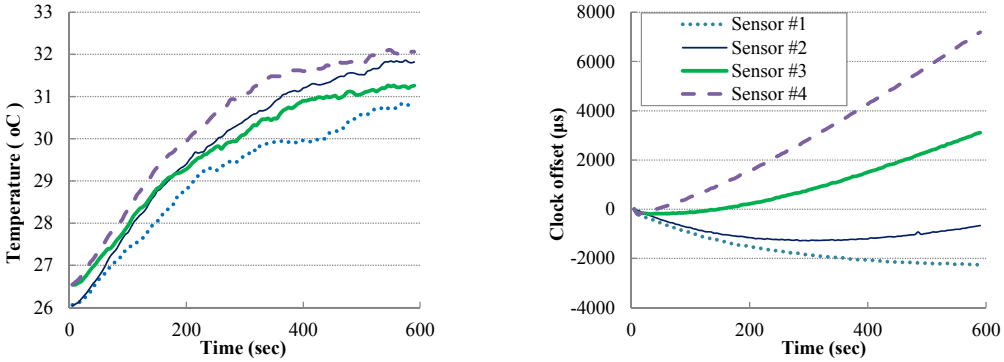


Figure 7. Temperature variation during sensing (left) and nonlinear clock drift (right)

Nonlinear clock drift poses an additional challenge to synchronized sensing for SHM. If not adequately compensated, nonlinear clock drift may significantly degrade the accuracy of data synchronization even though the data are resampled. In the next section, numerical investigation is performed to understand the effect of nonlinear clock drift on data synchronization accuracy.

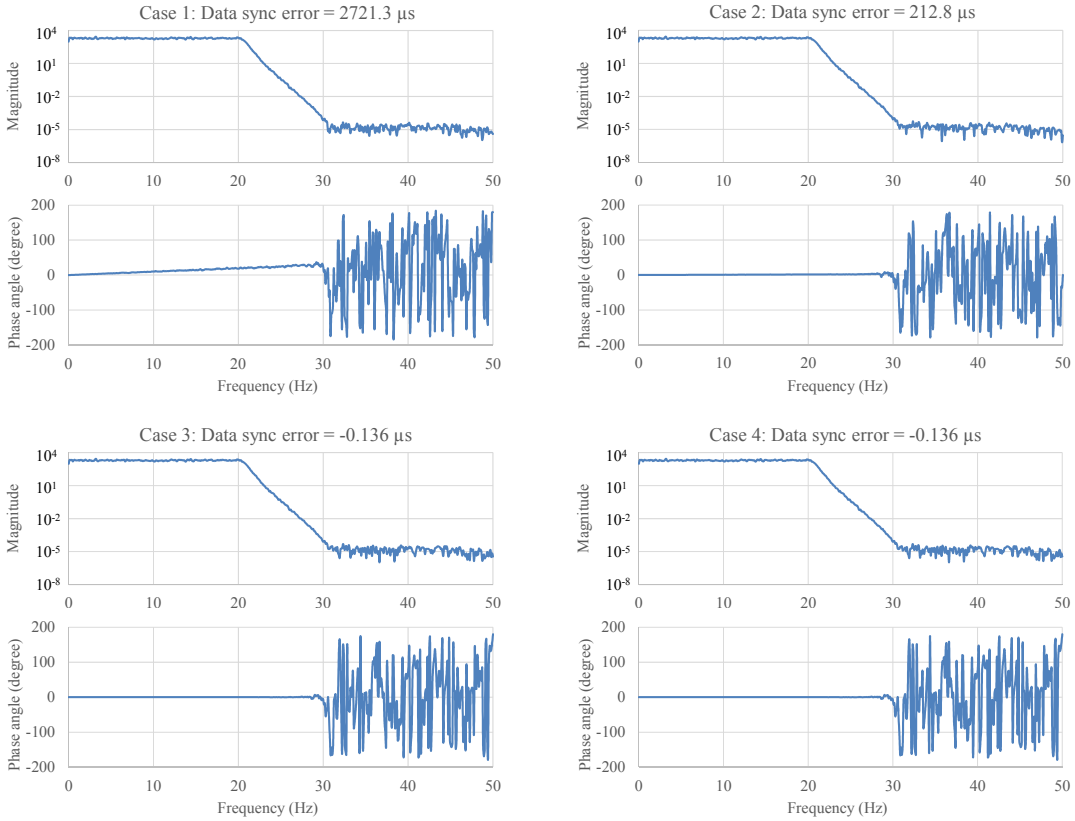
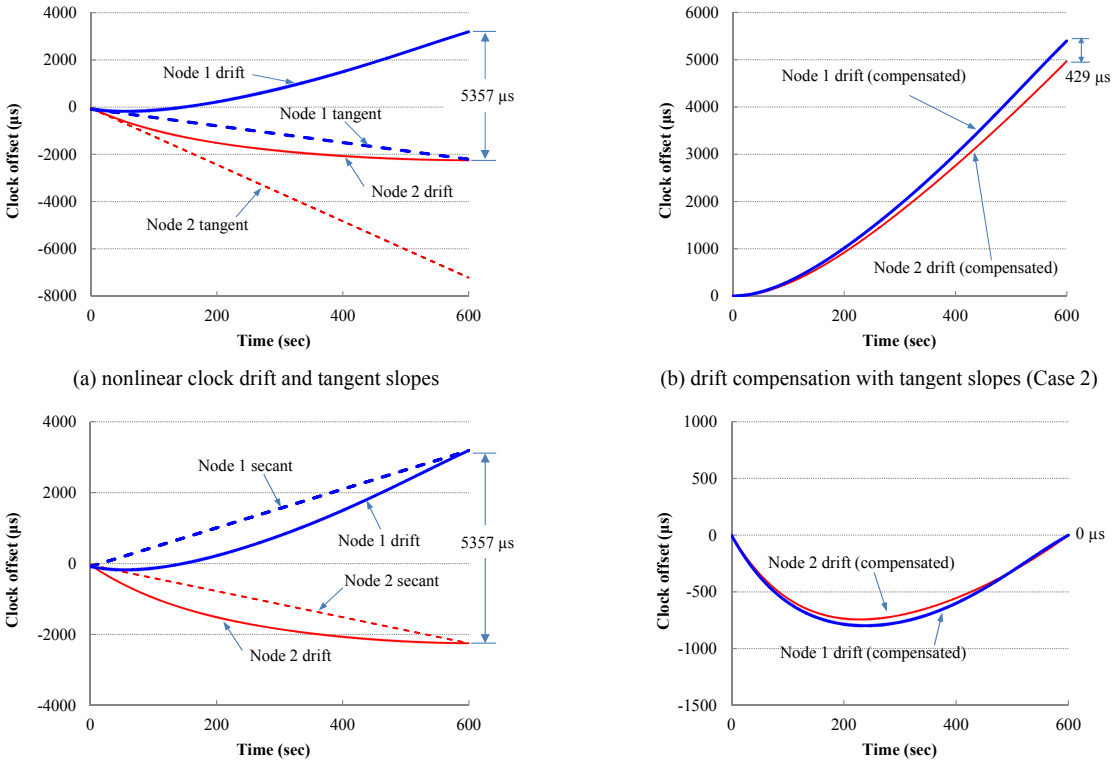


Figure 8. Data sync error with different drift compensation strategies (Case 1: no clock drift compensation; Case 2: clock drift compensated by the initial tangent slope of the nonlinear clock drift curve; Case 3: clock drift compensated by the secant slope of the nonlinear clock drift curve; Case 4: clock drift fully compensated by the nonliear clock drift curve)

V. EFFECT OF NONLINEAR CLOCK DRIFT ON DATA SYNCHRONIZATION ACCURACY

A sensing process with a nominal sampling frequency of f_s = 100 Hz and a duration of 600 seconds is simulated for a WSSN with two leaf nodes (sensing units), including leaf node 1 and leaf node 2, and one gateway node. Due to the heat generated during sensing, the clocks of leaf nodes 1 and 2 are assumed to drift away from the gateway node in nonlinear fashion. Leaf nodes 1 and 2 are assumed to follow the nonlinear drift curves of Sensor 1 and Sensor 3 shown in Fig. 6, respectively. One of the three factors that affect data sync accuracy discussed in Section 2.1, the uncertainty in crystal accuracy, is considered in the numerical simulation. The uncertainty in crystal accuracy is assumed to result in slightly different sampling frequencies of the two leaf nodes from the nominal sampling frequency. Therefore, leaf node 1 samples at f_{s1} = 99 Hz and leaf node 2 samples at f_{s2} = 101 Hz even though they are expected to sample at 100 Hz. In this numerical simulation, the raw signal to be sampled by the two leaf nodes is a BLWN signal with 20 Hz bandwidth. The sampling frequency of the raw signal is 9999 Hz, which is the product of f_{s1} and f_{s2}.

The two-stage TS strategy proposed by Nagayama and Spencer (2007) described in section 3.2 is selected as the basic TS approach for this investigation. Four cases are considered to study the effect of nonlinear clock drift on data synchronization accuracy under different clock drift compensation strategies: Case 1 does not implement any drift compensation and serves as a baseline for the comparison; Case 2 uses linear drift compensation based on the tangent slope of the nonlinear clock drift curve estimated at the beginning of sensing; Case 3 compensates clock drift through the secant slope of the clock drift curve assuming the first and last points of the curve are available; Case 4 completely compensates the nonlinear clock drift assuming the entire nonlinear clock drift curve is available. The basic idea of this two-stage TS method is that the algorithm first calculates the actual sampling frequency of the sampled data based on the drift-compensated data timestamps. Resampling is then performed to ensure the data is sampled at the correct sampling frequency. Detailed implementation of the resampling algorithm for WSSs can be found in Nagayama and Spencer (2007). Therefore, if the clock drift is not appropriately compensated, leading to inaccurate timestamps of the data, the estimated sampling frequency will be inaccurate, hence introducing error to the resampled data.



(a) nonlinear clock drift and tangent slopes

(b) drift compensation with tangent slopes (Case 2)

(c) nonlinear clock drift and secant slopes

(d) drift compensation with secant slopes (Case 3)

Figure 9. Illustration of clock drift compensation with tangent and secant slopes

To estimate the synchronization error of the data collected by the two leaf nodes, CPSD between the two signals is calculated. Any synchronization error will result in a non-zero slope of the phase angle of the CPSD. Therefore, a linear curve is fit to the phase angle between 0 Hz and 20 Hz and the slope θ of the linear curve can be converted to synchronization error through Eq. (1).

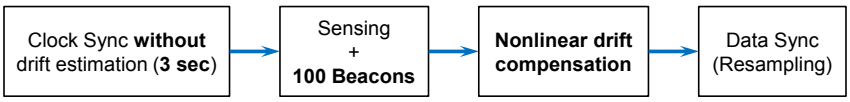
$$TS_{error} = \frac{\theta}{2\pi} \times 10^6 \ (\mu s) \tag{1}$$

The CPSD curves of the data and the associated synchronization errors are shown in Fig.8 for all four cases. First of all, in Case 1, when no drift compensation is implemented, a large data sync error of 2721.3 μs is observed. In fact, at the end of the sensing period, as shown in Fig. 9(a) and (c), the clocks of the two leaf nodes have drifted away from each other by 5357 μs. Therefore, drift compensation in TS is very important for SHM due to the extended sensing duration. In Case 2, linear drift compensation based on the initial tangent slope reduces the data sync error, but a relatively large error of 212.8 μs still exists in the data. The reason is that, as illustrated in Fig. 9(b), after clock drift compensation, a 429-μs offset still remains between the two clocks at the end of the sensing period. The error can increase further if sensing is performed for a longer duration. In addition, the data sync error depends on the nonlinearity in clock drift, so the error shown in this example is not the upper limit for the Case 2 approach. In Case 3, drift compensation using secant slope achieves very high accuracy of data sync with a -0.136 μs error, in that the secant slope accurately captures the trend of the nonlinear clock drift and is able to remove most of the error from the timestamps as shown in Fig. 9(d). The error in this simulation is in fact a numerical artifact introduced during the resampling process, since Case 4 shows the same accuracy even though the nonlinear clock drift has been completely compensated.

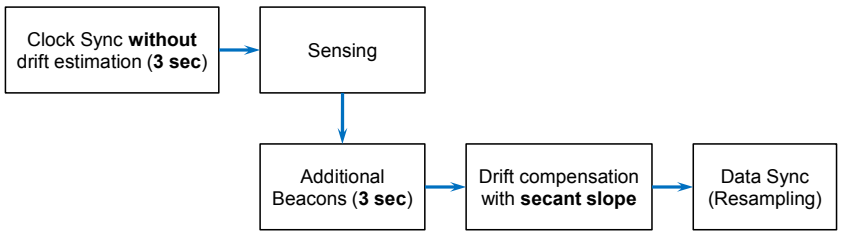
The numerical investigation demonstrates that clock drift compensation is very important and has tremendous impact on the synchronization accuracy of the collected data. Without compensation, the clocks can easily drift away from each other by thousands of microseconds and the sampled data can be rendered useless. Liner drift compensation based on the constant drift rate estimated before sensing, as proposed by Nagayama and Spencer (2007), can be effective if the clock drifts are linear. Even though the clocks drift nonlinearly, this approach may still achieve a certain level of accuracy, but the accuracy can decrease dramatically as the degree of nonlinearity and sensing duration increases. Drift compensation using the secant slope of the nonlinear clock drift showed excellent performance in correcting the timestamps, since the secant slope captures the trend of the nonlinear clock drift and gives more robust result than using the tangent slope. The conclusions drawn in this section will serve as the basis of the design of an efficient and robust TS method for SHM applications.

VI. AN EFFICIENT TIME SYNCHRONIZATION METHOD FOR SHM APPLICATIONS

To address the challenges of synchronized sensing in SHM described in section 2.3, a TS strategy is proposed based on the two-stage approach depicted in Fig. 5 by incorporating the two nonlinear clock drift compensation methods in Case 3 and Case 4, therefore leading to two different implementations. The two implementations are introduced first followed by the discussion on their advantages and disadvantages under different SHM scenarios.



(a) TS implementation 1: drift compensation using the complete nonlinear clock drift curve



(b) TS implementation 2: drift compensation using the secant slope of the nonlinear clock drift curve

Figure 10. Two implementations of the proposed TS method

6.1 First implementation: continuous beacons during sensing

As illustrated in Fig. 10(a), the first implementation aims to capture the full picture of nonlinear clock drift during the sensing period so as to fully compensate the nonlinear clock drift in data timestamps. Therefore, only a single synchronization message/beacon is needed to synchronize the clocks and then all leaf nodes in the network can start sensing at roughly the same time. Meanwhile, the initial clock offset (Δt_0) is estimated. During sensing, the gateway node continues to broadcast beacons periodically with its global time (t_{gb}) during sensing. Upon receiving the beacons, the leaf nodes time-stamp the beacons (t_{lb}) and compute the offsets (Δt_b). Once sensing is finished, the recorded local timestamps (t_{lb}) and offsets (Δt_b) are used to depict the complete history of clock drift during the sensing period through nonlinear regression analysis. Subsequently, the data timestamps (t_{gd}) can be corrected using the fitted nonlinear curve of clock drift. Finally, resampling is performed based on the

drift-compensated timestamps to achieve data synchronization. In order to take into account potential beacon loss due to packet collision during broadcast, 100 beacons are transmitted during sensing to ensure the accuracy and robustness of the nonlinear regression analysis.

One issue with this implementation is that the leaf nodes are performing sample acquisition and RF communication for beacon messages at the same time, which can potentially cause conflict and result in outliers in the received beacon data. The reason is that the operating system, TinyOS, employs a First In, First Out (FIFO) method when scheduling and executing tasks. If the entire processing related to sending/receiving packets is finished in the window between sample timestamping, there is no conflict. However, if the message processing overlaps with the sample acquisition and timestamping, message time-stamping may get delayed. As presented in the authors previous work (Li et al., 2012), outlier detection based on Cook's Distance (Cook, 1977) is implemented to remove the outliers before nonlinear regression is performed.

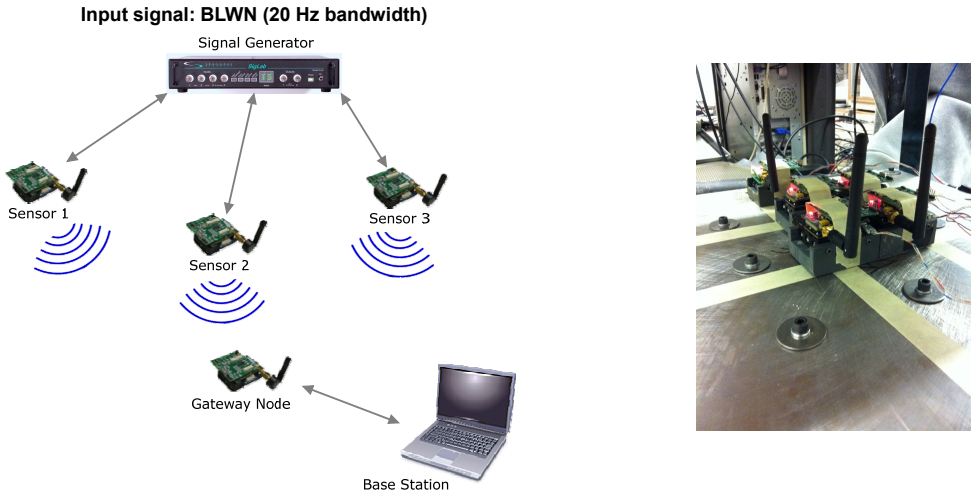
6.2 Second implementation: beacons before and after sensing

The second implementation is based on the drift compensation strategy utilizing the secant slope of the nonlinear clock drift. As illustrated in Fig. 10(b), similar to the first implementation, a single beacon is sent to the leaf node to synchronize the clocks. The beacon is time-stamped upon reception by the leaf nodes and the initial offset is calculated and saved. The leaf nodes then start sensing at roughly the same time. Once sensing is completed, an additional beacon is broadcast to the leaf nodes to record a second data point, which is then combined with the first data point collected before sensing to construct the secant line of the nonlinear clock drift. To compensate for packet loss, transmission is repeated a few times for both pre-sensing and post-sensing beacon messages. The secant slope is then used to correct data time-stamps. Resampling is finally carried out for the data based on the correct timestamps.

6.3 Comparison between the two implementations

As discussed in section 5, in theory, both implementations are able to eliminate the effect of nonlinear clock drift if the nonlinear clock drift curve or the associated secant slope can be accurately estimated. However, in practice, the first implementation should show better performance in terms of stability and accuracy because the nonlinear clock drift curve can be more robustly estimated using the large number of beacons collected during sensing, whereas the second implementation only collects limited beacons before and after sensing. The disadvantage of the first implementation, on the other hand, is the requirement of maintaining communication between the gateway and leaf nodes during the sensing period. If the communication environment during sensing is stable, the first implementation is recommended.

The second implementation is preferred when the communication environment between the gateway and leaf nodes is unstable. For example, when a SHM system is monitoring the bridge vibration under a train passing the bridge, the communication between the gateway and leaf nodes can be affected by the electromagnetic field of the train and the train blocking the line of sight between the sensors. Another example is the monitoring of moveable structures such as swing bridges. If the WSSN is to capture the bridge response when the bridge is moving, the gateway node may lose communication with some moving leaf nodes. In such cases, the second implementation shows advantages because the gateway node does not need to maintain communication with the leaf nodes during sensing. Moreover, outliers of the beacon messages is not a concern, since the leaf nodes no longer perform two tasks concurrently. Both implementations have improved efficiency compared with Nagayama and Spencer (2007) since the delay before start of sensing has been reduced significantly.



(a) Test 1 (b) Test 2

Figure 11. Test setups for TS accuracy evaluation

VII. EXPERIMENTAL EVALUATION

To evaluate the performance of the proposed synchronized sensing strategy, two tests are performed for the first implementation. In the first test, three Imote2 sensors with SHM-A sensor boards are connected to a signal generator which generates a BLWN signal with 20 Hz bandwidth. The test setup is illustrated in Fig. 11(a). Identical signals are therefore measured by the three wireless smart sensors at a sampling frequency of 100 Hz. Eq. (1) is used to calculate the TS errors once the CPSDs between the measured signals are estimated. To verify the effectiveness of the proposed TS strategy for long duration sensing, three sensing durations are considered including 1 minute, 10 minutes, and 30 minutes. A few tests are repeated for each case and the averaged results are summarized in Table 1. Overall, high synchronization accuracies (< 30 μs) have been achieved in the collected data for all cases. The nonlinear drift compensation strategy showed improved accuracy for all tests compared with the ones with drift compensation by initial slope. In particular, the biggest improvements are achieved in the tests with the 30-miniute sensing duration, for which the clock drift of each sensor is likely to experience the most significant nonlinearity due to temperature effect. In the second test, as shown in Fig. 11(b), five Imote2 sensors with SHM-A sensor boards are installed on top of a shake table. The shake table produces a BLWN excitation with 40Hz bandwidth. The shake table excitation is sampled at 100 Hz by the sensors. 100 seconds of data was collected and the minimum pair-wise synchronization error among the sensors is estimated as 6.46 μs and the maximum pair-wise synchronization error is 42.22 μs.

Tests have also been performed to validate the second implementation. Preliminary results show that the beacons collected before and after sensing are able to estimate the secant slope of the nonlinear clock drift, which ensures the accuracy of the drift compensation and in turn the TS accuracy of the collected data. These test results have proved the effectiveness of the proposed synchronized sensing strategy considering nonlinear clock drift.

Table 1. Data synchronization accuracy (Test 1)

Sensing Duration	Drift compensation	Pairwise synchronization Error (μs)		
		Pair 1	Pair 2	Pair 3
1 minute	Initial tangent slope	26.3	6.8	22.8
	nonlinear	25.5	6.7	22.6
	difference*	-2.98%	-1.21%	-0.87%
10 minutes	Initial tangent slope	19.2	10.6	25.2
	nonlinear	18.4	10.4	24.8
	difference*	-4.25%	-2.05%	-1.57%
30 minutes	Initial tangent slope	11.3	15.0	26.3
	nonlinear	6.9	13.0	19.9
	difference*	-38.55%	-13.37%	-24.15%

* difference = (nonlinear - linear)/(linear)*100%

VIII. CONCLUSION

In this paper, the unique features and challenges of synchronized sensing have been discussed for SHM applications using wireless smart sensor networks. The stringent requirement on data sync accuracy, the extended sensing duration, temperature variation, and the need for rapid response for transient vibration events are the main factors that set synchronized sensing in SHM apart from other WSSN applications. The influence of nonlinear clock drift on data sync accuracy has been investigated. The synchronized sensing strategy based on the initial slope of clock drift works well when clocks drift linearly; however, the accuracy decreases when the clock drift curves become nonlinear. The proposed strategy using the secant slope of the clock drift achieves the same level of accuracy as the one with fully compensated nonlinear clock drift. Two improved TS implementations have been proposed by incorporating the verified drift compensation methods. The two implementations have improved accuracy and robustness against nonlinear clock drift and have better efficiency since the delay before sensing due to clock synchronization has been significantly reduced. The advantages and disadvantages of both implementations have also been discussed based on different applications scenarios. Tests have been performed which verified the effectiveness of the new TS implementations.

REFERENCES

[1] Spencer Jr., B.F., Ruiz-Sandoval, M, and Kurata, N. (2004), "Smart Sensing Technology: Opportunities and Challenges," *Structural Control and Health Monitoring,* 11, 349– 368.

[2] Lynch, J. P. and Loh, K. J. (2006). "A Summary Review of Wireless Sensors and Sensor Networks for Structural Health Monitoring." *The Shock and Vibration Digest*, 38(2), 91-128.

[3] Nagayama, T. and Spencer Jr., B. F. (2007). "Structural Health Monitoring using Smart Sensors." *NSEL Report Series, No. 1*, University of Illinois at Urbana-Champaign.

[4] Nagayama, T., Moinzadeh, P., Mechitov, K., Ushita, M., Makihata, N., Ieiri, M., Agha, G., Spencer Jr., B. F., Fujino, Y., and Seo, J.-W. (2010). "Reliable multi-hop communication for structural health monitoring." *Smart Structures and Systems*, 6(5), 481-504.

[5] Krishnamurthy, V., Fowler, K., and Sazonov, E. (2008). "The effect of time synchronization of wireless sensors on the modal analysis of structures." Smart Materials and Structures, 17, 13.

[6] Elson, J.E. (2003). *Time Synchronization in Wireless Sensor Networks*. Ph.D. Thesis, University of California, Los Angeles.

[7] Ganeriwal, S., Kumar, R., and Srivastava, M.B. (2003). "Time-Sync Protocol for Sensor Networks." *The First ACM Conference on Embedded Networked Sensor System (SenSys)*, p. 138-149.

[8] Maroti, M., Kusy, B., Simon, G., and Ledeczi, A. (2004). "The flooding time synchronization protocol." *Proceedings of 2nd International Conference on Embedded Networked Sensor Systems*, Baltimore, MD, 39-49.

[9] Li, J., Nagayama, T., Mechitov, K.A., and Spencer Jr., B.F. (2012). "Efficient Campaign-type Structural Health Monitoring Using Wireless Smart Sensors." *Proc. SPIE Smart Structures/NDE 2012*, San Diego, CA.

[10] Yang, Z., Cai, L., Liu, Y., and Pan, J. (2012). "Environmental-Aware Clock Skew Estimation and Synchronization for Wireless Sensor Networks." *2012 Proceedings IEEE INFOCOM*.

[11] Kim, S., Pakzad, S., Culler, D., Demmel, J., Fenves, G., Glaser, S., and Turon, M. (2007). "Health Monitoring of civil infrastructures using wireless sensor networks." *Proceedings of the 6th International Conference on Information Processing in Sensor Networks*, Cambridge, Massachusetts, USA.

[12] Wang, Y., Lynch, J.P., and Law, K.H. (2007). "A Wireless Structural Health Monitoring System with Multithreaded Sensing Devices: Design and Validation." *Structure and Infrastructure Engineering*, 3(2), 103-120.

[13] Whelan, M.J., Gangone, M.V., Janoyan, K.D., and Jha, R. (2009). "Real-Time Wireless Vibration Monitoring for Operational Modal Analysis of an Integral Abutment Highway Bridge." Engineering Structures, 31(10), 2224-2235.

[14] Bocca, M. and Eriksson, L.M. (2011). "A Synchronized Wireless Sensor Network for Experimental Modal Analysis in Structural Health Monitoring." *Computer-Aided Civil and Infrastructure Engineering*, 26, 483-499.

[15] Mahmood, A. and Jäntti, R. (2009)."Time Synchronization Accuracy in Real-time Wireless Sensor Networks." *Proceedings of the 2009 IEEE 9th Malaysia International Conference on Communications*, Kuala Lumpur, Malaysia.

[16] Sazonov, E., Krishnamurthy, V., and Schilling, R. (2010). "Wireless Intelligent Sensor and Actuator Network – A Scalable Platform for Time-synchronous Applications of Structural Health Monitoring." *Structural Health Monitoring*, 9(5), 465-476.

[17] Uddin, M.B. and Castelluccia, C. (2010). "Toward Clock Skew Based Wireless Sensor Node Services." *Wireless Internet Conference (WICON), 2010 The 5th Annual ICST*.

[18] Cook, R. Dennis, "Detection of Influential Observations in Linear Regression". *Technometrics*, 19 (1): 15–18 .