STRUCTURAL CONTROL AND HEALTH MONITORING
Struct. Control Health Monit. 2016; **23**:470–486
Published online 19 August 2015 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/stc.1782

Efficient time synchronization for structural health monitoring using wireless smart sensor networks

Jian Li[1,*,†], Kirill A. Mechitov[2], Robin E. Kim[3] and Billie F. Spencer, Jr.[3]

[1]*Department of Civil, Environmental, and Architectural Engineering, The University of Kansas, Lawrence, KS, USA*
[2]*Department of Computer Science, University of Illinois at Urbana-Champaign, Urbana, IL, USA*
[3]*Department of Civil and Environmental Engineering, University of Illinois at Urbana-Champaign, Urbana, IL, USA*

SUMMARY

Wireless smart sensor networks (WSSNs) have shown great promise in structural health monitoring (SHM), because of their advantages of low cost, higher flexibility, robust data management, and ability to provide better understanding of structural behavior through dense deployment of sensors. However, implementation of wireless SHM systems poses many challenges, one of which is ensuring adequate synchronization of the collected data. This issue arises in WSSNs because each smart sensor in the network having an independent processor with its own local clock, and this clock 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 because of random delays from both software and hardware sources; that is, synchronized clocks do not necessarily yield synchronized sensing. Various algorithms have been proposed to achieve both synchronized clocks and sensing. However, these protocols still lack the desired performance for SHM applications for reasons of extended data collection time, temperature variations resulting in nonlinear clock drift, requirement for prompt response, and so on. In this paper, the unique features and challenges of synchronized sensing for SHM applications are discussed, followed by a numerical investigation of the effect of nonlinear clock drift on data synchronization accuracy. A new synchronized sensing strategy considering nonlinear clock drift compensation is proposed with two different implementations to meet various application requirements. Experimental results show that the proposed time synchronization approach can compensate for temperature effects on clock drift and provide efficient and accurately synchronized sensing ($<50\,\mu$s maximum error) for SHM, even for long sensing duration. Copyright © 2015 John Wiley & Sons, Ltd.

Received 25 October 2014; Revised 28 May 2015; Accepted 30 July 2015

KEY WORDS: wireless smart sensors; synchronized sensing; time synchronization; structural health monitoring; nonlinear clock drift

1. INTRODUCTION

Structural health monitoring (SHM) is a technology that encompasses a broad range of methods and applications with the goal of informing the current condition of a structure to assist with structural maintenance. Although visual inspection still remains one of the most important maintenance methods for civil infrastructure, it is time-consuming, labor intensive, expensive, and prone to error. One tragic example is the collapse of the I-35W highway bridge over Mississippi River in Minnesota in 2007, killing 13 people and injuring 145 even though it passed a visual inspection a year prior to failure. SHM aims to assess the structural integrity using sensing technology and signal processing approaches, enabling a long-term, continuous monitoring paradigm that can provide a more reliable and economic way of damage detection and maintenance of civil infrastructure.

*Correspondence to: Jian Li, Department of Civil, Environmental, and Architectural Engineering, The University of Kansas, Lawrence, KS, USA.
[†]E-mail: jianli@ku.edu

However, the high cost of traditional wired SHM systems, due in part to onerous and expensive cabling networks, prohibits high-density sensor deployment and hence limits the application and success of SHM for civil infrastructure. Wireless smart sensors (WSSs) are devices that have sensor, microprocessor, radio frequency (RF) transceiver, memory, and power source integrated into one small size unit and are therefore 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 micro-electro-mechanical systems (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. WSSs have the potential to change fundamentally the way civil infrastructure systems are monitored, controlled, and maintained [1].

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 [2–4]. Among them, the time synchronization issue arises in wireless smart sensor networks (WSSNs) because of the fact that each smart sensor in the network has an independent processor, with 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 of the following: (1) the sensors start sensing at different times because of random variations in the processing time in the sensor board driver; (2) the sampling frequencies among the sensor nodes are different because of the low quality of crystals; and (3) the sampling frequency for each individual sensor node can fluctuate over time because of jitter [5]. 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, can contain significant errors [5,6].

Time synchronization protocols for WSSNs 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) [7], Time-sync Protocol for Sensor Networks (TPSN) [8], and Flooding Time Synchronization Protocol (FTSP) [9]. To realize synchronization in the measured data for SHM, in addition to utilizing clock synchronization protocols to achieve synchronized clocks, Nagayama and Spencer [5] proposed a resampling-based approach to further process the data and remove the errors introduced by the three factors mentioned previously. However, these protocols may still lack the desired performance for SHM applications under certain demanding circumstances such as extended data collection time, temperature variation, and requirement for prompt response.

In the following sections, two sample application scenarios of SHM are presented followed by the discussion of time synchronization requirements for SHM applications. 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. The advantages and disadvantages of the new approach are discussed.

2. TIME SYNCHRONIZATION REQUIREMENTS FOR STRUCTURAL HEALTH MONITORING APPLICATIONS

Compared with other WSSN applications such as air/water quality monitoring, agricultural monitoring, traffic monitoring, and so on, SHM applications have their own characteristics that pose unique requirements for WSSNs in various aspects including data acquisition [10,11], radio communication [4], and time synchronization. Two SHM examples are described herein and used to illustrate the unique time synchronization requirements for SHM.

DOI: 10.1002/stc

2.1. Jindo Bridge structural health monitoring project

The second Jindo Bridge is one of the twin cable-stayed bridges located in the southwestern tip of the Korean Peninsula, linking Jindo Island to the Korean Peninsula (Figure 1). It is a three-span steel-box girder cable-stayed bridge with a 344-m main span and two 70-m side spans. The steel-box girder is supported by 60 stay cables connected to two A-shaped steel pylons supported by concrete piers. Figure 1 shows the picture of both the first (right) and the second (left) Jindo Bridges. The second Jindo Bridge is the object of the SHM project and will be hereafter referred to as the Jindo Bridge.

Two major deployments of wireless SHM system have been implemented on the Jindo Bridge with the goal of realizing a large-scale, long-term autonomous monitoring system for civil infrastructures using WSSN. The first WSSN was deployed in 2009 with 70 sensor nodes, each composed of an Imote2 (MEMSIC) with an SHM-A sensor board [12]. The entire network was divided into two sub-networks with two separate base stations. Most nodes were powered by D-cell batteries while a few nodes were equipped with Lithium-ion rechargeable batteries and solar panels. More details can be found in Jang *et al.* [13]. In the second deployment in 2010, both hardware and software were significantly improved. The number of sensor nodes was increased to 113, with total 669 sensor channels. The network was divided into four sub-networks. All sensor nodes were powered by solar panels in this deployment. Multi-hop communication protocol and improved fault tolerance features were added. After the second deployment, a number of further improvements and upgrades were made in 2011. More details can be found in Jo [14]. Figure 2 depicts the layout of sensor nodes in 2011.

2.2. Rock Island Arsenal Government Bridge Structural Health Monitoring project

The Government Bridge is an eight-span, double decker, steel truss bridge located over the Mississippi River between Rock Island, IL and Davenport, IA. The bridge was built in 1896 and is owned and operated by the US Army and the Army Corps of Engineers. The upper deck carries rail traffic, and the lower deck carries vehicular traffic. A distinct feature of the bridge is that the second span of the bridge can swing 360° to allow boats to pass, as shown in Figure 3.

A wireless SHM system consisting of 22 wireless smart sensor nodes (Imote2 + SHM-A) was installed on the swing span to monitor the acceleration response under ambient and traffic excitation [15]. Figure 4 shows the layout of the WSS nodes. Both the ‘locked’ and ‘open’ positions of the swing span were monitored.

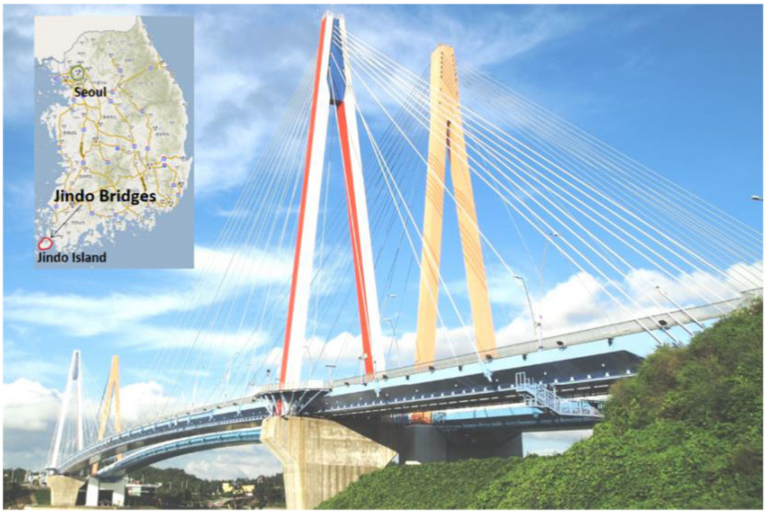


Figure 1. The first (right) and the second (left) Jindo Bridges.

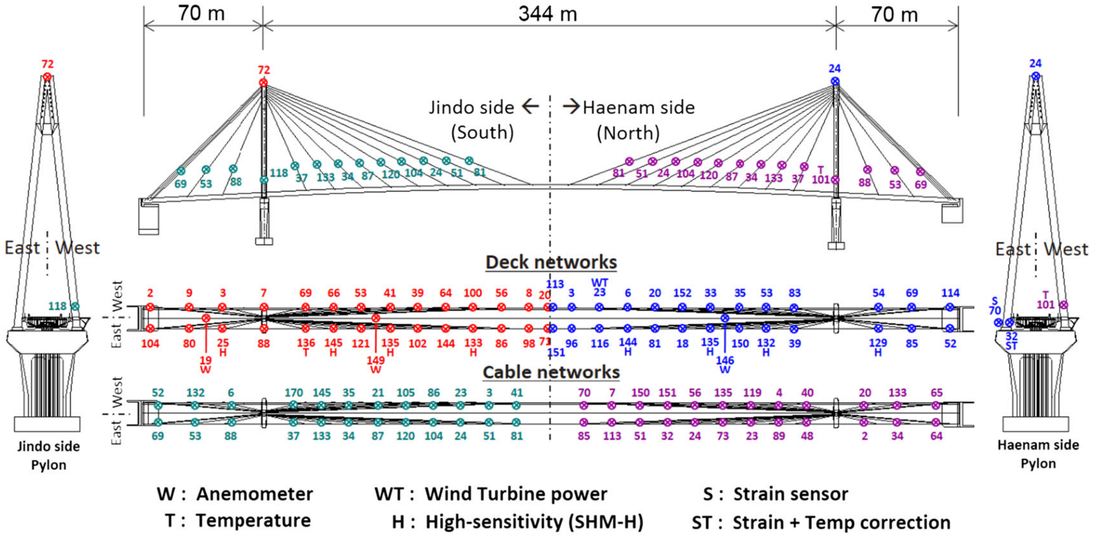


Figure 2. Wireless smart sensors deployed on the second Jindo Bridge in 2011.

Figure 3. The Government Bridge (Swing Span) [15].

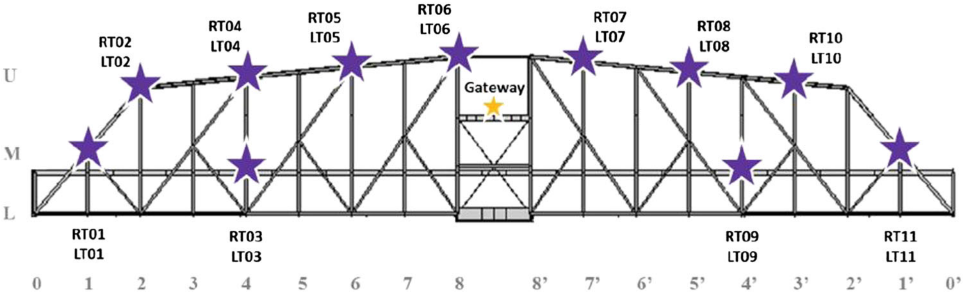


Figure 4. Sensor layout of the Government Bridge structural health monitoring system [15].

2.3. *Structural health monitoring-specific time synchronization challenges for wireless smart sensor networks*

As in many other WSSN applications, synchronization of the network is highly desirable in SHM applications. However, because of 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. Based on the issues and concerns identified from the aforementioned two SHM projects, we discuss the SHM-specific features and the imposed challenges in time synchronization.

2.3.1. Synchronized clocks versus synchronized sensing. 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 that is at least twice of the frequency of interest is needed to capture the 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° error in phase angle at 10 Hz and 36° error at 100 Hz [3]. Krishnamurthy *et al.* [6] performed experimental modal analysis on a simply supported beam and illustrated that a small time shift (30 μs) at one sensor location resulted in a noticeable error in mode shapes, especially for higher-order modes. For damage detection methods that rely on mode shape information, such as Modal Strain Energy-based Methods [16,17] and Modal curvature-based methods [18], the mode shape error due to such a small inaccuracy in time synchronization, depending on applications, can lead to false results in damage detection.

For the Jindo Bridge WSSNs, the deck and cable vibrations were measured at 25 and 50 Hz, respectively. The cable network uses a higher sampling frequency because the maximum cable modal frequency required for calculating cable tension is around 20 Hz [19]. For the Government Bridge, acceleration responses were measured at 50 Hz in order to capture damage-sensitive modal information [15]. Other civil engineering structures such as low-rise buildings can have even higher natural frequencies, and therefore, higher sampling rates are required when monitoring these structures.

Because of 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 *et al.* [3]: (1) because of the random variation 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; and (3) the sampling frequency for each individual sensor node can also fluctuate over time because of jitter. Note that because of software timing issues, compared with real-time Operating Systems, the randomness in sensing start time is more prominent in sensors implemented on event-driven operating system platforms such as TinyOS (http://www.tinyos.net).

2.3.2. 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 SHM 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 can be achieved. In addition, extended sensing duration is required to fully capture a transient event such as the swinging process of the Government Bridge and the forced vibration due to a train crossing the bridge.

In the case of Jindo Bridge network, for each sensing task, 10,000 data points were collected at 25 Hz, resulting in 400 s of sensing duration. For the Government Bridge, to capture the entire vibration response during a swing event, 10 min of data was collected at 50 Hz because the swinging takes about 7 min. Also to capture the complete record of the forced vibration caused by a train crossing the bridge, which takes about 10 min, longer sensing duration is required.

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 because of 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 timestamps, which in turn leads to synchronization error in the sampled data. Nagayama *et al.* [3] tested Imote2 nodes and estimated the maximum clock drift rate among the tested set of nodes to be around 50 μs per second, which can lead to 20 ms synchronization error after a 400-s measurement. Appropriate clock skew compensation is necessary to eliminate or reduce such an impact.

2.3.3. 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 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.* [20] performed 10-min 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 because of the heat generated by the Imote2 CPU and the analog-to-digital converter (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.* [21] showed clear correlation between temperature and different pairs of clocks exhibit different clock skew change patterns with respect to temperature. In the aforementioned experiments by Li *et al.* [20], 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.

2.3.4. 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 starting sensing. Therefore, a delay is introduced between the reception of the command and the start of sensing because of the need for resynchronization. Such a delay may cause the entire transient event such as earthquakes to be missed. In the case of the Jindo Bridge network, the resynchronization takes 30 s to collect beacon packets to estimate the linear clock drift rate, which is used later to perform drift compensation for the data timestamps. In the Government Bridge case, when the sentry sensor detects the movement of the swing span, it has to wake up the entire network first and perform resynchronization before the collection of data; therefore, a significant portion of the swing event is 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.

3. SURVEY OF TIME SYNCHRONIZATION IN WIRELESS SMART SENSOR NETWORKS

3.1. Clock synchronization protocols

Among the various clock synchronization protocols for WSSNs, three have been widely used, including the Reference Broadcast Synchronization algorithm (RBS; [7]), the Time-sync Protocol for Sensor Networks (TPSN; [8]), and the Flooding Time Synchronization Protocol (FTSP; [9]).

Reference Broadcast Synchronization is a receiver–receiver method in which a reference node broadcasts reference messages that are timestamped at the receivers upon the 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.

Time-sync Protocol for Sensor Networks, on the other hand, is a sender–receiver approach in which the reference messages are timestamped at both the sender and receiver sides. Therefore, TPSN has both send-side and receive-side delays. However, TPSN eliminates some delays on both sides by timestamping radio messages at the Medium Access Control (MAC) layer for multiple times and then averaging those times, by which the random delays associated with encoding, decoding, and interrupt handling can be eliminated. Therefore, TPSN can achieve higher accuracy than the RBS. TPSN does not compensate for clock skew and requires a fixed spanning tree of the network to be established before time synchronization.

Flooding Time Synchronization Protocol is similar to TPSN in the sense that it is also a sender–receiver method and is based on MAC layer timestamping. However, because flooding is used to disseminate the 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 previous described methods can achieve very high accuracy in synchronization of clocks of wireless sensors. For example, the experiments based on the Mica2 motes showed that the FTSP achieved 1.5 μs in the single hop scenario and the average precision of 0.5 μs per hop in the multi-hop case [9]. Based on the Mica motes, also in the single hop scenario, the RBS and the TPSN achieved 11 and 17 μs, respectively. Note that the achievable accuracy depends on the platform architecture. For example, Mica motes can generally achieve better accuracy with a given protocol than Imote2s because of less concurrent tasks, and hence less delay, in the software. The achievable accuracy is a function of the processor, the radio chip, and the drivers. Nevertheless, as discussed in Section 2.3.1, accurately synchronized clocks do not guarantee synchronized data. Moreover, clock skew compensation in FTSP and RBS is based on the assumption that the 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.

3.2. Time synchronization applications in structural health monitoring

Several SHM applications with WSSNs have attempted to achieve synchronization in wireless collection of vibration data. Kim *et al.* [22] 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 is adopted for synchronizing the clocks of the sensor nodes. Temporal jitter is 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.* [23] 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-min period. Whelan *et al.* [24] developed another prototype wireless SHM system using a similar time synchronization approach as Wang *et al.* [23]. The approach uses a single command to initiate sensing in the network and relies on accurate and stable crystal oscillators to achieve adequate time synchronization of the network for the sampling duration. Bocca *et al.* [25] adopted a TS protocol called μ-sync [26] for wireless SHM, which is derived from FTSP. Therefore, MAC layer timestamping and linear drift skew compensation are also utilized. The synchronization accuracy in the collected data was not reported, but an accuracy of 10 μs was observed in clock synchronization.

In the aforementioned SHM applications, the performance of the TS algorithms was reported in terms of clock synchronization, but not in the collected data. Sazonov *et al.* [27] proposed a hierarchical time synchronization architecture in which local cluster nodes are synchronized by beacon signals, whereas spatially distributed cluster nodes are synchronized by the GPS time reference. Synchronization accuracy was calculated by looking at the phase of the collected sinusoidal signals using the wireless sensors. An accuracy 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 large such that temperature effect becomes significant.

Nagayama and Spencer [5] proposed a two-stage TS strategy, which is able to achieve tight synchronization between the sampled data. As illustrated in Figure 5, in the first stage, FTSP is adapted to provide clock synchrony. Before sensing, a 30-s 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

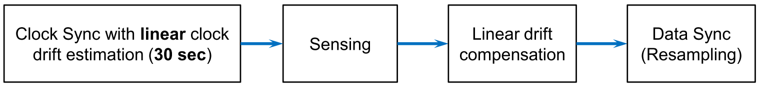

Clock Sync with **linear** clock drift estimation (**30 sec**)
Sensing
Linear drift compensation
Data Sync (Resampling)

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

DOI: 10.1002/stc

are then used to correct the clocks so that the samples can be accurately timestamped during sensing. After sensing is finished, in the second stage, the data are resampled to remove the sources of error due to the three factors discussed in Section 2.3.1 so as to achieve tight synchrony between the sampled data. Using Imote2s and SHM-A sensor boards 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 nonlinearity in clock skew becomes significant because of temperature effect. In addition, the 30-s period for broadcasting beacon messages before sensing delays the start of sensing and is therefore undesirable in SHM as discussed in Section 2.3.4.

4. 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-s period. FTSP is utilized to enable MAC layer timestamping when sending and receiving beacon packets. The gateway node is programmed to transmit beacon packets to the four receivers every 2 s. The beacon packet is timestamped, and the send time is attached to the beacon right before it is transmitted. Upon the reception of the beacon packet, the receivers timestamp 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 that are plotted in Figure 6. No sensing was performed during the process, and CPU was maintained at a low clock speed. 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.

When temperature changes, the resonant frequency of clock crystal oscillators will change, leading to nonlinear clock drift. Uddin *et al.* [28] 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 [21]. 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 same approach was used to monitor the clock drift of a set of four

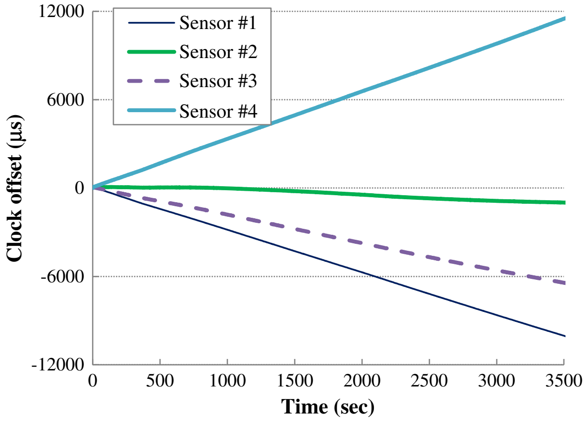


Figure 6. Linear clock drift under constant temperature.

Imote2 leaf nodes 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 Figure 7, the on-board temperatures of the Imote2 leaf nodes were raised by about 6 °C during the 600-s sensing period. As a result, severe nonlinearity is observed in the clock drift. It should be mentioned that nonlinear clock drift can be minimized if temperature-compensated crystal oscillators were employed by the wireless sensors. However, such clocks are more power hungry than traditional oscillators and therefore may not be good solutions for low-power wireless sensors.

Nonlinear clock drift poses an additional challenge to achieving synchronized sensing for SHM. If not appropriately compensated, nonlinear clock drift may significantly degrade the accuracy of data synchronization even though resampling is used to remove the sources of inaccuracy such as randomness in processing time, uncertainty in crystal accuracy, and temporal jitter, as described in Section 2.3.1. In the next section, numerical investigation is performed to better understand the effect of nonlinear clock drift on data synchronization accuracy.

5. EFFECT OF NONLINEAR CLOCK DRIFT ON DATA SYNCHRONIZATION ACCURACY

A sensing process with nominal sampling frequency of $f_s = 100$ Hz and sensing during a period of 10 min is simulated for a WSSN with two leaf nodes (sensing units), including leaf node 1 and leaf node 2, and one gateway node. Because of the heat generated during sensing, the clocks of leaf nodes 1 and 2 are assumed to drift away from the gateway node's clock in a nonlinear fashion by following the curves of Sensor 1 and Sensor 3 shown in Figure 6, respectively. One of the three factors that affect data sync accuracy discussed in Section 2.3.1, 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 realized by the two leaf nodes compared with the nominal sampling frequency. Therefore, leaf node 1 actually samples at $f_{s1} = 99$ Hz and leaf node 2 samples at $f_{s2} = 101$ Hz even though they are configured 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 *et al.* [3] 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 has linear drift compensation based on the tangent slope of the nonlinear curve of clock drift at the beginning of sensing; Case 3 compensates clock drift through the slope of the secant line of the clock drift curve, assuming the first and last points of the curve can be captured; Case 4 completely compensates the nonlinear clock drift, assuming the entire 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 timestamps. Resampling is then performed to ensure the data are sampled

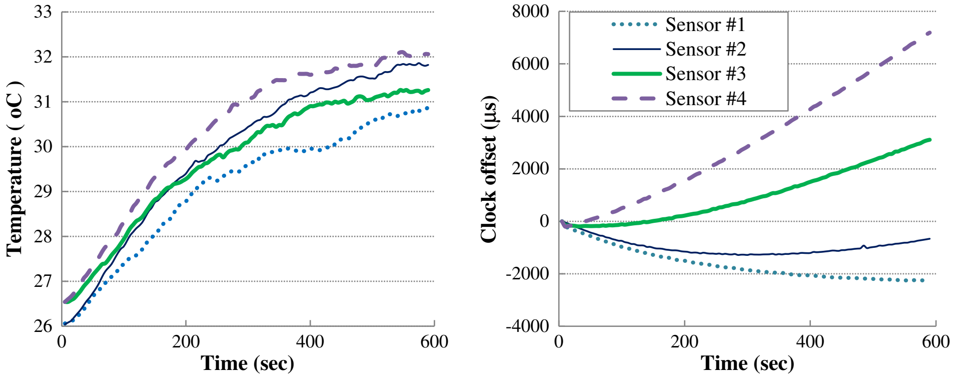


Figure 7. Temperature variation during sensing (left) and nonlinear clock drift (right).

DOI: 10.1002/stc

at the correct sampling frequency. Detailed implementation of the resampling algorithm for wireless smart sensors is described in Nagayama *et al.* [3]. Therefore, if the clock drift is not appropriately compensated, which leads to inaccurate timestamps of the data, the estimated sampling frequency will not be accurate, introducing error to the resampled data.

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

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

The CPSD curves of the data and the associated synchronization errors are shown in Figure 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 10-min sensing period, the clocks of the two leaf nodes have drifted away from each other by 5357 μs as shown in Figure 9(a) and (c). Therefore, drift compensation in TS is very important for SHM because of the extended sensing duration. In Case 2, linear drift compensation based on the initial tangent slope reduces the TS error, but a relatively large error of 212.8 μs still exists in the data. The reason is that, as illustrated in Figure 9(b), after the 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 TS error in the data depends on the specific nonlinearity in clock drift, so the error shown in this example is not the upper limit for the approach in Case 2. In Case 3, drift compensation using the secant slope achieves very high accuracy of TS in the data with a −0.136 μs error, because the secant slope accurately captures the trend of the nonlinear clock drift and is able to remove most

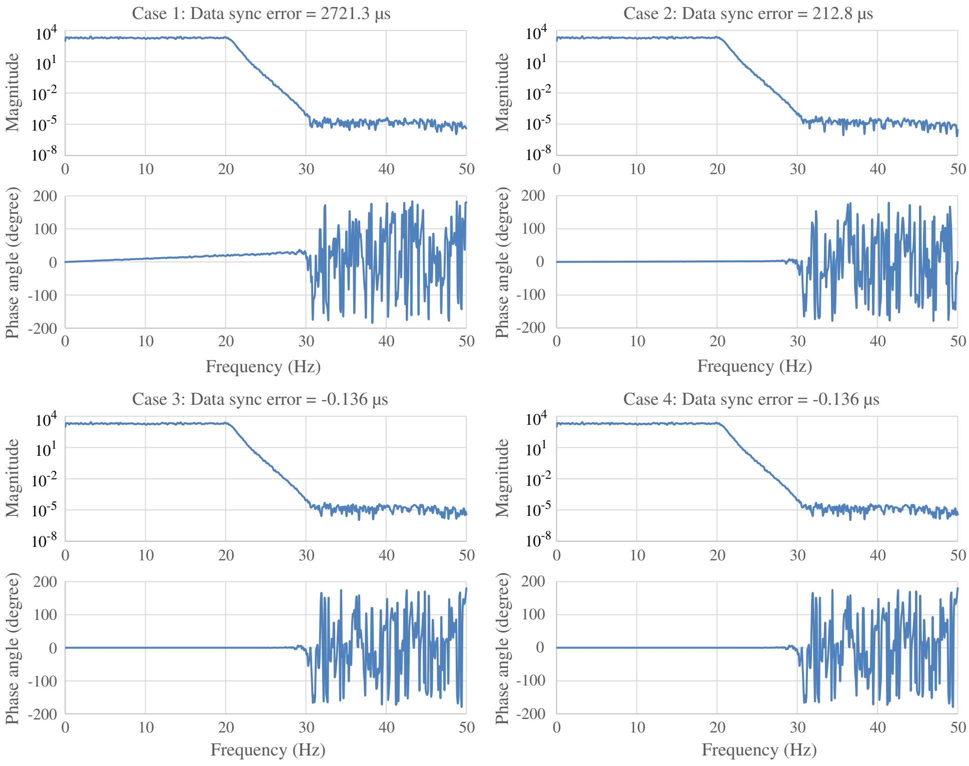
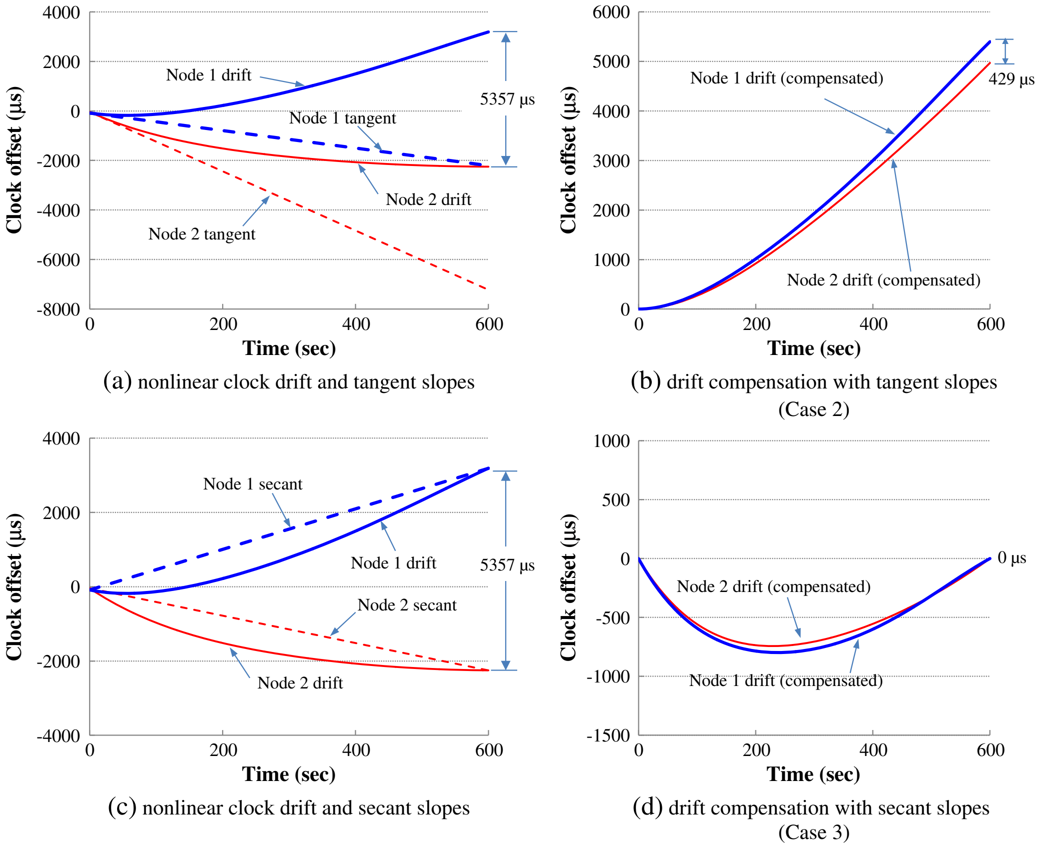


Figure 8. Data sync error with different drift compensation strategies (Case 1: no clock drift compensation; Case 2: clock drift compensated by the initial tangent slope of the nonlinear clock drift curve; Case 3: clock drift compensated by the secant slope of the nonlinear clock drift curve; Case 4: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



(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. (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).

of the error from the timestamps as shown in Figure 9(d). The error in this particular numerical simulation is in fact a numerical artifact introduced during the resampling process, because 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 data can therefore be rendered useless. Linear drift compensation based on the drift rate estimated before sensing, as proposed by Nagayama *et al.* [3], can be effective if the clock drifts are linear. Even though the clocks drift nonlinearly, this approach can still achieve certain 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 removing the timestamp errors because the secant slope captures the trend of the nonlinear drift and is much more robust than the tangent slope. The conclusions drawn in this section will be used to guide the design of an efficient and robust TS method for SHM applications.

6. AN EFFICIENT TIME SYNCHRONIZATION METHOD FOR STRUCTURAL HEALTH MONITORING 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 Figure 5 [3] by incorporating the two nonlinear clock drift compensation methods in Case 3 and Case 4, therefore leading to two implementations. The two implementations are introduced first followed by the discussion on their advantages and disadvantages under different SHM scenarios.

DOI: 10.1002/stc

6.1. First implementation: continuous beacon messages during sensing

As illustrated in Figure 10, 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 at the beginning, 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 timestamp 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 entire 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 the broadcast, 100 beacons are transmitted during sensing to ensure the accuracy and robustness of the nonlinear regression analysis. The number of beacons during sensing can be adjusted to take into account of the length of sensing.

One issue with this implementation is that the leaf nodes need to perform two tasks at the same time, namely, sample acquisition and RF communication for beacon signals. For wireless smart sensors such as the Imote2 with limited resources and running an event-driven operating system such as TinyOS, performing two tasks at the same time can potentially cause jitter and result in outliers in the received beacon data, because tasks are run non-preemptively in First In, First Out order. If the entire processing related to sending/receiving packets is finished in the window between sample timestamping during sample acquisition, there is no interference. However, if the message processing overlaps with the sample acquisition and timestamping, message timestamping may be delayed.

A test aimed to evaluate the effect of such a conflict was performed by the authors [20] and is shown here for completeness. The test was performed with Imote2s and SHM-A sensor boards. One hundred beacon messages were sent from the gateway node to the leaf node during a two-min period, while the leaf node was also performing sensing. Three tests were carried out with three sampling frequencies for sensing, including 25, 100, and 280 Hz. The leaf node collected the clock offset calculated from the beacon messages, which are plotted against time in Figure 11. As can be seen, at low sampling frequency (25 Hz), the chance of conflict between sample acquisition and RF communication is low because of the relatively long time interval between two sample acquisitions. However, when sampling frequency is increased, the chance of conflict is higher, leading to outliers in the received beacon messages. One can imagine that further increase of the sampling frequency will lead to even more outliers,

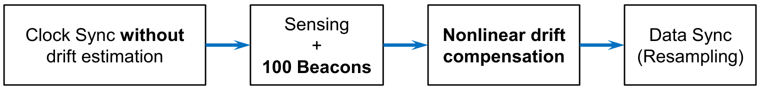
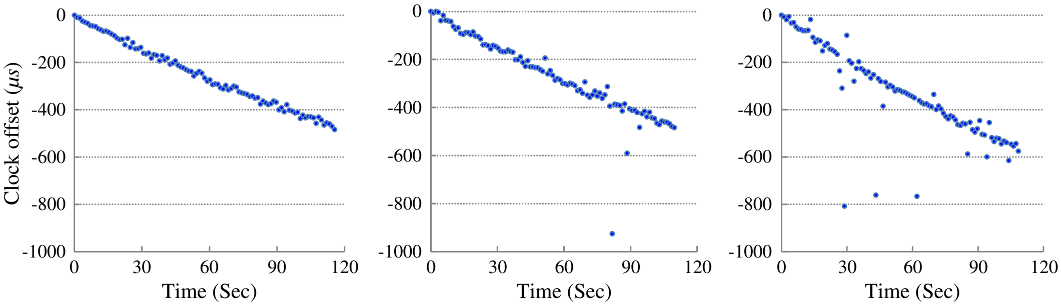

Clock Sync **without** drift estimation
Sensing + **100 Beacons**
Nonlinear drift compensation
Data Sync (Resampling)

Figure 10. The first implementation of the proposed time synchronization method.



(a) $f_s = 25$ Hz (b) $f_s = 100$ Hz (c) $f_s = 280$ Hz

Figure 11. Beacon messages collected during sensing at different sampling frequencies. (a) $f_s = 25$ Hz, (b) $f_s = 100$ Hz, and (c) $f_s = 280$ Hz.

which will significantly reduce the accuracy of nonlinear regression result and hence that of the clock drift compensation.

To ensure the accuracy of nonlinear clock drift compensation, Cook's distance [29] is employed in the implementation to identify and subsequently remove the outliers in the beacon signals. Cook's distance measures the effect of removing one certain data point and can be calculated as

$$D_i = \frac{\sum_{j=1}^{n}\left(\hat{y}_j - \hat{y}_j(\mathrm{i})\right)^2}{(p+1)s^2}, \quad i{=}1{,}2{,}\ldots{,}n \tag{2}$$

in which $\hat{y}_j$ is the regression estimate of the full data set and $\hat{y}_j(i)$ is the same estimate but with the ith data point removed. s^2 is the mean square error of the regression model. p is the order of the regression model. As a rule of thumb, if Equation (3) is satisfied, the beacon message is considered as an outlier and is removed from the data set.

$$D_i \geqslant \frac{4}{n-(p+1)} \tag{3}$$

6.2. Second implementation: beacons before and after sensing

In the second implementation, the secant slope of nonlinear cock drift is used for drift compensation, in light of the simulation result presented in Section 5. As shown in Figure 12, to begin synchronized sensing, similar to the first implementation, one beacon message is broadcast to the leaf nodes to quickly synchronize the clocks of the network. After the initial clock synchronization, the leaf nodes store the beacon message and then start sensing roughly at the same time. After sensing is completed, an additional beacon message is sent to the leaf nodes to capture the second data point of clock drift. The secant slope of clock drift can then be computed by combing the two data points collected before and after sensing. To compensate packet loss, beacon transmission is repeated a few times for both pre-sensing and post-sensing beacon messages. Data timestamps are corrected accordingly before resampling is performed to achieve synchronized data.

6.3. Comparison between the two implementations

As demonstrated in the numerical simulation in Section 5, theoretically, both implementations can eliminate the effect of nonlinear clock drift if the full nonlinear curve or the secant slope of the curve is accurately estimated. However, in practice, the first implementation should show better performance in terms of stability and accuracy because the full 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 gateway and leaf nodes during the sensing period. If the communication environment during sensing is stable, the first implementation is preferred.

The second implementation is recommended when the communication environment between gateway and leaf nodes is unstable. For example, when monitoring the Government Bridge while the swing span is moving to the 'open' position, the gateway node may lose communication with some moving leaf nodes and hence is not able to deliver the beacon messages to them. Even for monitoring the bridge in the 'locked' position under train loading, the direct communication can be blocked by the moving train or affected by the electromagnetic field of the train. In such cases, the second implementation shows advantages because the gateway node does not need to maintain communication with the

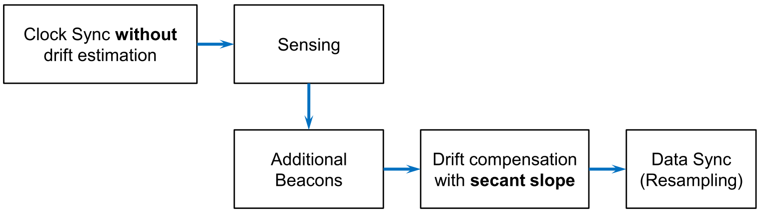

Clock Sync **without** drift estimation
Sensing
Additional Beacons
Drift compensation with **secant slope**
Data Sync (Resampling)

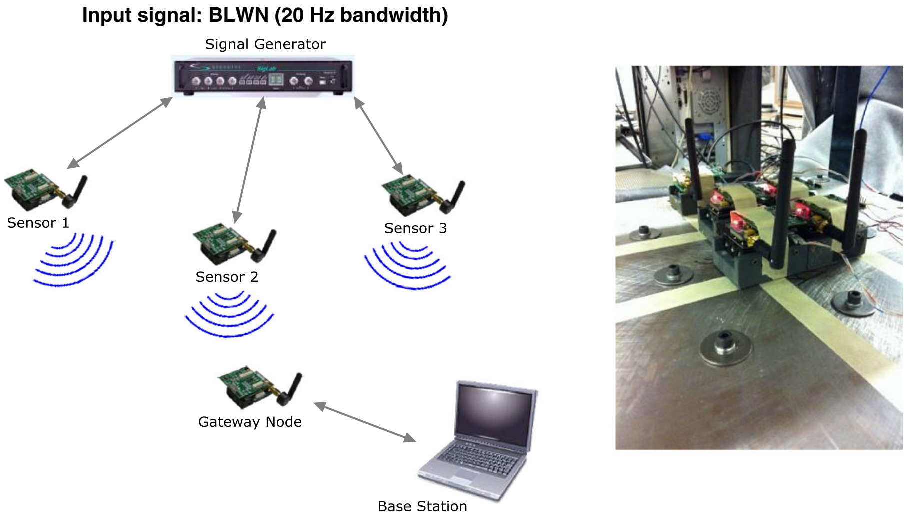
Figure 12. The second implementation of the proposed time synchronization method.

leaf nodes during sensing. Moreover, outliers of the beacon messages is not a concern, because the leaf nodes no longer perform two tasks concurrently. Both implementations have improved efficiency compared with Nagayama and Spencer [5] because the delay before start of sensing has been reduced significantly.

7. EXPERIMENTAL EVALUATION

Two test setups were created to evaluate the performance of the proposed synchronized sensing method. As illustrated in Figure 13(a), the first test setup is composed of a network of Imote2 sensors with three leaf nodes and one gateway node. The SHM-A senor boards of the leaf nodes were connected to a signal generator that provides identical input analog signals to the three leaf nodes so that the synchronization accuracy can be accurately evaluated with Equation (1). The gateway node is connected to a base station computer and communicates with the leaf nodes wirelessly for data acquisition. In the second test setup, as shown in Figure 13(b), five Imote2 sensor nodes were installed on top of a uni-axial shake table, so the sensors are expected to measure the same signal produced by the shake table. Equation (1) will also be used to calculate the synchronization error among the measured acceleration data.

To evaluate the performance of the first implementation of the proposed method, in test setup 1, the signal generator produced a BLWN signal with 20 Hz bandwidth, which was then sampled at 100 Hz at each leaf nodes. To verify the effectiveness of the method for long duration sensing, three sensing durations were considered, including 1, 10, and 30 min. In the meantime, linear regression of the beacon signals was also included to demonstrate the benefit of nonlinear regression. Each test was repeated for multiple times, and the averaged results are presented in Table I. Note that the tests were performed indoor where ambient temperature was relatively stable; temperature variation on the sensors was mostly caused by heat generated during sensing. Overall, the proposed method based on continuous beacon messages during sensing achieved high synchronization accuracy with errors less than 30 μs for all cases. Tests with nonlinear regression for drift compensation showed consistent improvements over the ones with linear regression, demonstrating the importance of accurately capturing the nonlinear behavior of clock drift during sensing. In particular, the biggest improvements were achieved in the 30-min tests, for which the clock drift of each sensor is likely to experience the most significant nonlinearity because of temperature effect. In the second test setup, the shake table produces a BLWN excitation with 40 Hz bandwidth, and the sensors measured the excitation at a 100 Hz sampling



(a) Test setup 1 (b) Test setup 2

Figure 13. Test setups for time synchronization accuracy evaluation. (a) Test setup 1 and (b) Test setup 2. BLWN, band-limited white noise.

Table I. Data synchronization accuracy (test setup 1).

Sensing duration	Drift compensation	Pairwise synchronization error (μs)		
		Pair 1	Pair 2	Pair 3
1 min	Linear regression	26.3	6.8	22.8
	Nonlinear regression	25.5	6.7	22.6
	Difference*	−2.98%	−1.21%	−0.87%
10 min	Linear regression	19.2	10.6	25.2
	Nonlinear regression	18.4	10.4	24.8
	Difference*	−4.25%	−2.05%	−1.57%
30 min	Linear regression	11.3	15.0	26.3
	Nonlinear regression	6.9	13.0	19.9
	Difference*	−38.55%	−13.37%	−24.15%

*Difference = (nonlinear – linear)/(linear) * 100%

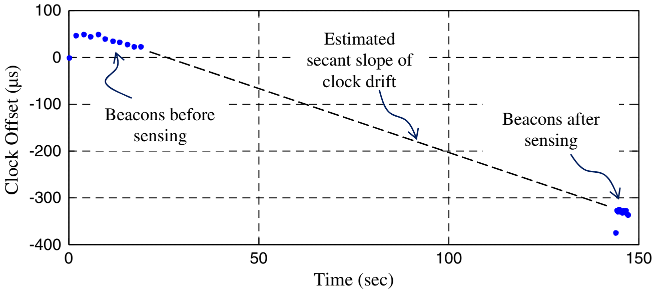


Figure 14. Beacon messages collected before and after sensing (second implementation).

frequency. One hundred seconds of shake table excitation was measured using the proposed method with continuous beacon. The maximum pairwise synchronization error is 6.46 μs, and the maximum is 42.22 μs.

Tests were also performed for the second implementation. A sample test result showing the beacon signals collected before and after sensing is presented in Figure 14. 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 aforementioned test results have proved the effectiveness of the proposed synchronized sensing strategy considering nonlinear clock drift.

8. CONCLUSION

In this paper, the unique features and challenges of synchronized sensing have been discussed for SHM applications using WSSNs. The stringent requirement on data sync accuracy, high sampling rate, 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 numerically. Synchronized sensing strategies with different clock drift compensation approaches have been analyzed. First of all, without clock drift compensation, the clocks on the leaf nodes can quickly drift away from each other, leading to significant error in the data timestamps and hence the sync accuracy of the data. The previous strategy based on the initial slope of clock drift works well when clocks drift linearly; however, the accuracy can decrease dramatically when nonlinear clock drift is introduced because of temperature change. The impact is even larger when sensing duration increases. In light of the numerical simulation results, a new synchronized sensing strategy is needed.

To address the issue of nonlinear clock drift, two strategies have been proposed for better drift compensation: one uses the secant slope and the other uses the fully captured nonlinear curve for drift compensation. Both of the two proposed strategies have achieved a high level of accuracy of data synchronization in the numerical simulation. The first implementation is designed for higher robustness, but it requires a stable wireless communication environment during the sensing period. The second implementation does not require communication during sensing, so can be used in the cases when radio environment changes over time during sensing. Both implementations have achieved higher efficiency, because the previous initial clock synchronization time has been removed, and hence, the network can respond more quickly to transient events. Finally, the performance of the proposed synchronized sensing strategy has been validated through experimental testing. The maximum data sync error is less than 30 μs from the first test setup with accurately control input signals and is less than 50 μs from the second test setup with a shake table.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the partial support of this research by the National Science Foundation under the grants nos. CMS 06–00433, CMMI 07–24172, CMS 09–28886, and CPS 10–35773.

REFERENCES

1. Spencer BF, Jr., Ruiz-Sandoval M, Kurata N. Smart sensing technology: opportunities and challenges. *Structural Control and Health Monitoring* 2004; **11**:349–368.
2. Lynch JP, Loh KJ. A summary review of wireless sensors and sensor networks for structural health monitoring. *The Shock and Vibration Digest* 2006; **38**(2):91–128.
3. Nagayama T, Sim SH, Miyamori Y, Spencer BF Jr. Issues in structural health monitoring employing smart sensors. *Smart Structures and Systems* 2007; **3**(3):299–320.
4. Nagayama T, Moinzadeh P, Mechitov K, Ushita M, Makihata N, Ieiri M, Agha G, Spencer BF Jr, Fujino Y, Seo J-W. Reliable multi-hop communication for structural health monitoring. *Smart Structures and Systems* 2010; **6**(5):481–504.
5. Nagayama T, Spencer Jr. BF. Structural health monitoring using smart sensors. NSEL Report Series, No. 1, 2007, University of Illinois at Urbana-Champaign.
6. Krishnamurthy V, Fowler K, Sazonov E. The effect of time synchronization of wireless sensors on the modal analysis of structures. *Smart Materials and Structures* 2008; **17**:1–13.
7. Elson JE. Time synchronization in wireless sensor networks. Ph.D. Thesis, University of California, Los Angeles, 2003.
8. Ganeriwal S, Kumar R, Srivastava MB. Time-sync protocol for sensor networks. *The First ACM Conference on Embedded Networked Sensor System* (*SenSys*), 2003; p. 138–149.
9. Maroti M, Kusy B, Simon G, Ledeczi A. The flooding time synchronization protocol. Proceedings of 2nd International Conference on Embedded Networked Sensor Systems 2004, Baltimore, MD, 39–49.
10. Rice JA, Spencer Jr. BF. Structural health monitoring sensor development for the Imote2 platform. *Proc. SPIE Smart Structures/NDE* 2008, Vol. 6932, San Diego, CA.
11. Jo H, Sim SH, Nagayama T, Spencer BF Jr. Development and application of high-sensitivity wireless smart sensors for decentralized stochastic modal identification. *ASCE Journal of Engineering Mechanics* 2012; **138**(6):683–694.
12. Rice JA, Spencer BF. Flexible smart sensor framework for autonomous full-scale structural health monitoring. NSEL Report Series, No. 18, 2009, University of Illinois at Urbana-Champaign. (http://hdl.handle.net/2142/13635)
13. Jang S, Jo H, Cho S, Mechitov K, Rice JA, Sim SH, Jung HJ, Yun CB, Spencer BF Jr, Agha G. Structural health monitoring of a cable-stayed bridge using smart sensor technology: deployment and evaluation. *Smart Structures and Systems* 2010; **6**(5–6):439–460.
14. Jo H. Multi-scale structural health monitoring using wireless smart sensors, Ph.D. Thesis, University of Illinois at Urbana-Champaign, Urbana, 2013.
15. Giles R. Development of a long-term, multimetric structural health monitoring system for a historic steel truss swing bridge. Ph.D. Thesis, University of Illinois at Urbana-Champaign, Urbana, 2013.
16. Cornwell P, Doebling SW, Farrar CR. Application of the strain energy damage detection method to plate-like structures. *Journal of Sound and Vibration* 1999; **224**(2):359–374.
17. Sazonov E, Klinkhachorn P, Halabe UB, GangaRao H. Non-baseline detection of small damages from changes in strain energy mode shapes. *Nondestructive Testing and Evaluation* 2002; **18**(3–4):91–107.
18. Pandey AK, Biswas M, Samman MM. Damage detection from changes in curvature mode shapes. *Journal of Sound and Vibration* 1991; **145**(2):321–332.
19. Sim S, Li J, Jo H, Park J, Cho S, Spencer BF Jr, Jung H. Wireless smart sensor network for automated monitoring of cable tension. *Smart Materials and Structures* 2014; **23**(2). doi:10.1088/0964-1726/23/2/025006.
20. Li J, Nagayama T, Mechitov KA, Spencer Jr. BF. EC-tSHMUWS sensors, *Proc. SPIE Smart Structures/NDE* 2012, San Diego, CA.
21. Yang Z, Cai L, Liu Y, Pan J. Environmental-aware clock skew estimation and synchronization for wireless sensor networks. *Proceedings IEEE INFOCOM*, 2012.
22. Kim S, Pakzad S, Culler D, Demmel J, Fenves G, Glaser S, Turon M. H monitoring of civil infrastructures using wireless sensor networks. *Proceedings of the 6th International Conference on Information Processing in Sensor Networks* 2007, Cambridge, Massachusetts, USA.
23. Wang Y, Lynch JP, Law KH. A wireless structural health monitoring system with multithreaded sensing devices: design and validation. *Structure and Infrastructure Engineering* 2007; **3**(2):103–120.

DOI: 10.1002/stc

24. Whelan MJ, Gangone MV, Janoyan KD, Jha R. Real-time wireless vibration monitoring for operational modal analysis of an integral abutment highway bridge. *Engineering Structures* 2009; **31**(10):2224–2235.
25. Bocca M, Eriksson LM. A synchronized wireless sensor network for experimental modal analysis in structural health monitoring. *Computer-Aided Civil and Infrastructure Engineering* 2011; **26**:483–499.
26. Mahmood A, Jäntti R. TS accuracy in real-time wireless sensor networks. *Proceedings of the IEEE 9th Malaysia International Conference on Communications* 2009, Kuala Lumpur, Malaysia.
27. Sazonov E, Krishnamurthy V, Schilling R. Wireless intelligent sensor and actuator network – a scalable platform for time-synchronous applications of structural health monitoring. *Structural Health Monitoring* 2010; **9**(5):465–476.
28. Uddin MB, Castelluccia C. Toward clock skew based wireless sensor node services. *Wireless Internet Conference* (*WICON*), 2010 *The 5th Annual ICST*.
29. Cook RD. Detection of influential observations in linear regression. *Technometrics* 1977; **19**(1):15–18.

DOI: 10.1002/stc