

Research Article

A Wireless Sensor Network with Enhanced Power Efficiency and Embedded Strain Cycle Identification for Fatigue Monitoring of Railway Bridges

Glauco Feltrin,¹ Nemanja Popovic,¹ Kallirroi Flouri,¹ and Piotr Pietrzak²

¹*Empa-Swiss Federal Laboratories for Materials Science and Technology, Ueberlandstrasse 129, 8600 Dübendorf, Switzerland*

²*Department of Microelectronics and Computer Science, Technical University of Lodz, 116 Zeromskiego Street, 90-924 Lodz, Poland*

Correspondence should be addressed to Glauco Feltrin; glauco.feltrin@empa.ch

Received 29 July 2015; Accepted 15 October 2015

Academic Editor: Lung-Ming Fu

Copyright © 2016 Glauco Feltrin et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Wireless sensor networks have been shown to be a cost-effective monitoring tool for many applications on civil structures. Strain cycle monitoring for fatigue life assessment of railway bridges, however, is still a challenge since it is data intensive and requires a reliable operation for several weeks or months. In addition, sensing with electrical resistance strain gauges is expensive in terms of energy consumption. The induced reduction of battery lifetime of sensor nodes increases the maintenance costs and reduces the competitiveness of wireless sensor networks. To overcome this drawback, a signal conditioning hardware was designed that is able to significantly reduce the energy consumption. Furthermore, the communication overhead is reduced to a sustainable level by using an embedded data processing algorithm that extracts the strain cycles from the raw data. Finally, a simple software triggering mechanism that identifies events enabled the discrimination of useful measurements from idle data, thus increasing the efficiency of data processing. The wireless monitoring system was tested on a railway bridge for two weeks. The monitoring system demonstrated a good reliability and provided high quality data.

1. Introduction

Due to increased traffic volume, loads, and speeds on the European railway network, many existing railway bridges are currently subjected to loads and speeds that exceed those which they were designed for. Under these conditions, a major concern for metallic bridges is the remaining fatigue life. Several investigations have demonstrated that monitoring provides important data for assessing the remaining fatigue life [1–4]. In general, monitoring allows reducing significantly the uncertainties associated with the performance of critical structural components such as crossbeams and stringers that are exposed to many high amplitude load cycles.

Among different concepts and technologies that have been tested for monitoring civil structures, wireless sensor networks (WSNs) exhibit the attractive properties of being highly automatable and cable-free and therefore are easy to deploy and operate [5]. Short and medium term deployments, which in practice are likely to represent the majority of

monitoring applications, may benefit from using WSNs since they allow minimization of installation time and costs. Nevertheless, several technical challenges of WSN must be handled to meet the requirements of structural monitoring. In contrast to tethered sensor networks, a WSN does not have an external power supply and a dedicated communication media. Therefore, battery-operated WSNs are only competitive if they can provide good quality data and can be operated reliably for a sufficiently long time with minimal maintenance (e.g., battery replacement and system rebooting).

Strain monitoring for fatigue life evaluation of railway bridges is characterized by several peculiarities. Since fatigue assessment is performed only by considering strain cycle amplitudes, the absolute strain usually does not need to be recorded. This requirement simplifies strain measurements since strain changes due to temperature can be neglected. Consequently, strain recording is only necessary when a train is crossing the bridge. The monitoring process is therefore event driven. Since on most bridges the average time interval

between trains is much greater than the average recording period during train crossings, the implementation of an event driven data acquisition mechanism is advantageous. It enables a significant reduction of data communication, thus reducing data loss due to a congestion of communication channels. In addition, since data communication is very expensive in terms of power consumption, it enables a significant saving of energy consumption, which is always a critical issue in medium and long term deployments of WSN.

A wireless sensor network designed specifically for railway bridge monitoring was proposed by Chebroly et al. [6]. The system included an event driven monitoring mechanism through a node mounted in trains (mobile head node) that was also used to alert the sensor nodes on the bridge. The train acted as a mobile sink node by collecting the data generated by additional nodes in the WSNs. The feasibility of the concept was studied by means of a field experiment that took into account the performance of the mobile head node and the WSN head node. An actual deployment of an event driven WSN for railway bridge monitoring was described in [7]. The event detection was carried out independently on each sensor node with an ultra-low power MEMS acceleration sensor that continuously measured accelerations on the bridge and detected approaching trains. If this occurred, the sensor generated an interrupt that immediately switched on the strain gage's conditioning board and started data recording.

In this work, we present event based monitoring of stress cycles with a wireless sensor network demonstrated by field application on a steel railway bridge. The contribution of this work is twofold: (a) a new strain sensor board with 24-bit quantization resolution and reduced energy consumption and (b) an embedded cycle filtering algorithm, which enables dramatic reduction of the data communication overhead but still provides high quality input data for cycle counting algorithms, are presented.

The rest of the paper is organized as follows: Section 2 describes the strain sensor board, Section 3 addresses the embedded data processing algorithm, Section 4 presents the deployment on the railway bridge, Section 5 describe the results and, finally, Section 6 concludes this work.

2. Strain Sensing

Despite their importance in practice, strain monitoring with WSNs has received much less attention than vibration monitoring. Nagayama et al. [8] developed a sensor board for electrical resistance strain gages for the Berkeley Mote platform Mica2. The board's accuracy was experimentally verified in the laboratory with a small scale 3-story building structural model and the working resolution was about $4 \mu\text{m/m}$. Similarly, Choi et al. [9] developed a prototype strain sensor board for the Mica2 platform. Whelan and Janoyan [10] and Bischoff et al. [7] have designed a signal conditioning board for electrical resistance strain gages for the Tmote Sky platform. Both developments were tested with bridge deployments. Recently, Jo et al. [11] developed a high precision strain sensor board for the Imote2 platform that is able to measure low-level ambient strain by precisely

balancing the Wheatstone bridge. The sensor board was tested with a deployment on a full-scale cable-stayed bridge. Functionally providing similar strain monitoring capabilities as WSNs is the in-service bridge monitoring system proposed by Howell and Shenton III [12]. This commercial system is based on a small single-board computer with analog inputs and includes a cellular digital packet data modem for communication via a cellular network.

2.1. Signal Conditioning Board. Strain measurements on civil structures are usually performed with resistance strain gages. While their use with wired monitoring devices does not present particular problems, operating resistance strain gages with battery powered monitoring devices introduce two important drawbacks. First, the power consumption of resistance strain gages is rather high. While in vibration monitoring the typical power consumption of a good quality MEMS accelerometer is about a few mW, the widely deployed 120 Ohm resistance strain gages with Wheatstone bridge signal conditioning unit consume approximately ten times more power (around 50 mW). This figure is comparable to the power consumption of the radio unit in transmission mode which is the largest power consuming hardware component of a wireless sensor node. Therefore, continuous operation of resisting strain gages implies a dramatic reduction of the sensor node battery lifetime.

One way to overcome this problem is to operate the strain sensing unit only when an event occurs. In the case of a railway bridge, monitoring is required only when a train crosses the bridge. When events are sufficiently sparse, the strain sensing unit is most of the time switched off, which results in a significant power saving. However, this operating policy requires a triggering mechanism that is able to recognize train arrivals in advance and switch the strain sensing unit on. Such a mechanism was implemented and tested by Bischoff et al. [7]. The drawback of this operation policy is that when the strain sensing unit is switched on shortly before the train crosses a bridge a signal bias is produced due to heating of the strain gage by the current flow. This bias can be compensated by a dummy gage mounted in such a way that it is subjected to the same temperature changes but not to the same strain changes of the primary strain gage. Since fatigue monitoring only requires information about the amplitude of strain cycles and their frequency, temperature compensation is not crucial and can be omitted to avoid an increase of deployment time and costs.

Insights on how to improve energy efficiency and reduce signal bias due to heating can be achieved by using the formula that describes the power consumption of a linear resistor:

$$P = \frac{U^2}{R}, \quad (1)$$

where U is the voltage and R is the resistance. According to (1) power consumption can be reduced either by increasing the resistance or reducing the voltage. The improvement in power consumption achieved by increasing the resistance is linear. Replacing a 120Ω strain gage by a 700Ω strain gage

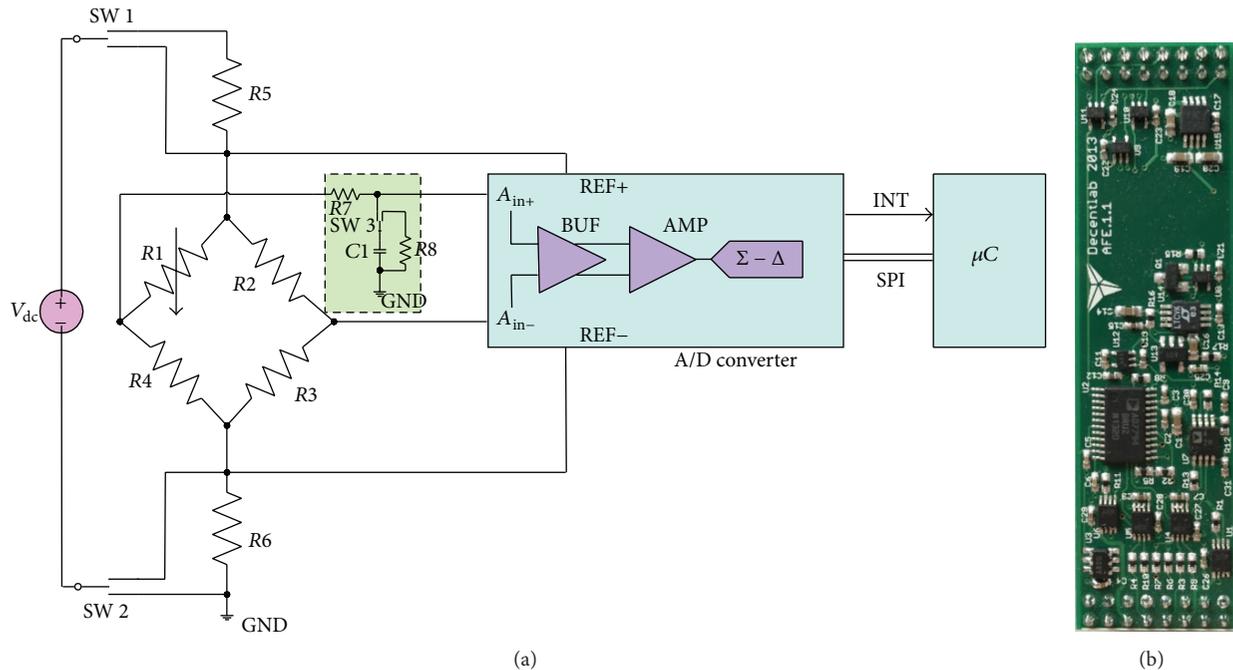


FIGURE 1: (a) Functional block diagram of the AFE and (b) AFE sensor board.

would already achieve a power saving by more than a factor 5. Another approach to reduce power consumption is to reduce the input voltage. Reducing the input voltage by a factor 2, for example, from 3 V to 1.5 V, would reduce the power consumption by a factor 4. Both options have the effect to decrease the signal-to-noise ratio. Since signal noise increases with increasing resistance of the strain gage, applying a 700 Ω gage instead of a 120 Ω strain gage will increase the noise. Reducing the input voltage will reduce the sensitivity of the signal conditioning circuit (Wheatstone bridge) since it is proportional to the input voltage [13]. Thus, reducing the input voltage by a factor two decreases the signal-to-noise ratio on the analog-to-digital converter (ADC) input, which may lead to loss of a bit at output. Common wireless sensor platforms with 12-bit ADCs may therefore be unable to provide the required performance in terms of amplitude resolution and range.

To overcome this problem, signal conditioning and amplification board were developed. Although not discussed in this paper, the board was designed to support not only resistance strain gages but also different kinds of analog sensors. As such, it will be referred to as analog front end (AFE). The functional block diagram of the AFE is presented in Figure 1. The core of the AFE is a $\Sigma - \Delta$ low noise ADC with 24-bit resolution [13]. Such an on-chip instrumentation amplifier, with programmable amplification gain that ranges from 1 to 128, provides a direct interface for small amplitude signals to the input ports. Its power consumption while operating is 400 μA @ 3 V, which is classified as low power. The AFE has six analog inputs and therefore can be used for various applications. The main drawback of such a high resolution ADC is that it is significantly slower than

common 12-bit ADCs since its maximum sampling rate is 470 Hz.

Measurement of strain gage resistance is performed using a Wheatstone bridge. In Figure 1, the strain gage is marked with R1. The fixed valued resistors R2 and R3 form the one-half of the Wheatstone bridge that is the source of the reference voltage. These resistors feature small tolerance and low temperature coefficient. R4 is either a resistor or a dummy gage for temperature compensation and is placed on the same half of the bridge with the strain gage. Ideally, this resistor or dummy gage features the same resistance as of the strain gage in order to develop balanced bridge and minimize nonlinearities in the bridge output.

By programming switches 1 and 2, one can include resistors R5 and R6 in the signal conditioning circuit. Including these resistors in the active circuit increases the equivalent resistance of the overall system and hence decreases the power consumption. If, for example, all resistors have the same reference resistance R, the power consumption is reduced by factor 3 since the system equivalent resistance is equal to 3R. Because the voltage on the Wheatstone bridge is less than the input voltage V_{dc} , to make the measurement circuit ratiometric to the reduced voltage, the voltage between resistors R5 and R6 and the Wheatstone bridge is sourced to the ADC as reference voltage. The ADC considers this reduced voltage as the full-scale range and the signal in this range is thus quantized with 24 bits. In a configuration with equal resistors, the excitation voltage across the Wheatstone bridge is reduced to one-third of V_{dc} .

The electrically programmable switch SW3 is used to include/omit a low-pass RC antialiasing filter in the circuit. By changing the values of resistor R7 and capacitor C1,

TABLE 1: Power consumption of a sensor node with AFE.

Resistance of strain gage [Ω]	Resistance of resistors $R5$ and $R6$ [Ω]	Current [mA]	Overall power consumption [mW]	Power consumption AFE [mW]
120	0	19.6	58.8	39.5
120	300	6.77	20.3	12.7
700	300	3.83	11.5	6.5

TABLE 2: Rms of noise with a 120 Ω strain gage.

Frequency [Hz]	Resistors $R5$ and $R6$	Rms noise [ADC out]	Rms noise [bits]	Rms noise [$\mu\text{m/m}$]
242	On	737	9.5	1.4
	Off	311	8.3	0.6
123	On	532	9.1	1.0
	Off	216	7.8	0.4
62	On	427	8.7	0.8
	Off	191	7.6	0.4

one can adjust the cut-off frequency. In the case that this filter is omitted from the signal conditioning circuit, resistor $R8$ is used to discharge capacitor $C1$.

The middle points of the two legs of the Wheatstone bridge are connected to the differential input of the ADC. The signal is buffered and amplified with a programmable gain ranging from 1 to 128 and converted to a digital value. To achieve the highest resolution at ADC output, the amplification gain should be set as high as possible. The choice of this value is only limited by the expected amplitude range of the input signal. For the configuration displayed in Figure 1, the amplitude range with respect to the amplification gain is given by

$$\varepsilon_r = \pm \frac{1}{k} \frac{4}{g-2}. \quad (2)$$

Using resistance strain gages with a gage factor k of 2 and an amplifier gain g of 128 the amplitude range of the AFE is approximately $\pm 15.900 \mu\text{m/m}$, which represents a strain of 1.59%. With an ADC operating at 16 bits the theoretical resolution is $0.5 \mu\text{m/m}$. Operating the converter at 24 bits, the theoretical resolution is further decreased to $0.002 \mu\text{m/m}$. Amplitude range and resolution, even with 16 bits, comply well with the requirements of strain measurements for fatigue assessment of civil steel structures.

2.2. Power Consumption. Power consumption measurements of the complete hardware (sensor node platform equipped with an AFE) were performed with a sensor node that was continuously performing strain measurement but did not process and transmit the data. The tests were performed with an AFE having resistors $R2$ and $R3$ of $160 \text{ k}\Omega$ and resistors $R5$ and $R6$ of 300Ω . Resistor $R4$ had always the same resistance as the strain gage. The power supply V_{dc} featured 3 V. Due to their high resistance, the effect of resistors $R2$ and $R3$ on the overall circuit resistance can be neglected. The overall resistance is thus the sum $R_{\text{tot}} = R_1 + R_2 + R_5 + R_6$. Since the power consumption of the AFE was not directly measured, it was estimated by multiplying the measured current with the

overall resistance of the active circuit R_{tot} and adding 2 mW to approximately consider the other components of the AFE.

The results are shown in Table 1. With resistors $R5$ and $R6$ being removed from the active circuit, which is equivalent to standard circuits, the power consumption using a 120Ω strain gauge is the largest. By including resistors $R5$ and $R6$, the power consumption can be reduced by a factor 3. Replacing a 120Ω strain gauge with a 700Ω strain gauge reduces the overall power consumption by approximately a factor 2. The estimated power consumption of the AFE drops from 12.7 mW to 6.5 mW since the equivalent resistance increases from 840Ω to $2 \text{ k}\Omega$. The power consumption of the sensor node platform, calculated as overall consumption less the AFE power consumption, increases with the increase of the current sourced to AFE. This effect is credited mainly to the voltage stabilizer of the sensor node platform. Thus, reducing the current of the AFE allows also achieving an important power saving on the sensor node platform as well.

If nodes are equipped with two D-type batteries of capacity 14 mAh, with 120Ω strain gages, the replacement of batteries is necessary after almost 5 months, while for 700Ω strain gages this period extends to 8.5 months. However, on a real deployment, this period is smaller due to data processing and communication.

2.3. Signal Noise. Including resistors $R5$ and $R6$ in the active circuit has the benefit of reduced power consumption. However, this operation reduces the input voltage of Wheatstone bridge and thus its sensitivity. Furthermore, resistors $R5$ and $R6$ tend to increase the noise level of the circuit. The overall effect is thus a reduction of the signal-to-noise ratio. To evaluate the impact of signal noise, both 120Ω and 700Ω strain gauges were mounted to a large laboratory testing machine with a heavy mass in order to minimize the effect of noise induced by ambient vibrations. Configurations with and without resistors $R5$ and $R6$ were tested for three different sampling frequencies. The data were recorded with an amplifier gain of 128 and a resolution of 24 bits. The results are presented in Tables 2 and 3.

TABLE 3: Rms of noise with a 700 Ω strain gage.

Frequency [Hz]	Resistors $R5$ and $R6$	Rms noise [ADC out]	Rms noise [bits]	Rms noise [$\mu\text{m}/\text{m}$]
242	On	381	8.6	0.7
	Off	365	8.5	0.7
123	On	319	8.3	0.6
	Off	285	8.2	0.5
62	On	267	8.1	0.5
	Off	251	8.0	0.5

The tables show that the rms of signal noise has an amplitude that varies between 8 and 10 bits. A significant variation of the noise amplitude is observed with a strain gage of 120 Ω (Table 2) when resistors $R5$ and $R6$ are included in the active circuit. The increase of the rms of signal noise varies between a factor of 2.1 and 2.5. This figure corresponds to a resolution loss that ranges from 1 bit to 2 bits. As expected, signal noise increases with increasing frequency bandwidth. In terms of strain units, the greatest rms was 1.4 $\mu\text{m}/\text{m}$ at a sampling rate of 242 Hz.

When resistors $R5$ and $R6$ are removed from the active circuit, the rms noise increases when using a strain gage with 700 Ω resistance instead of a strain gage with 120 Ω resistance (Table 3). This increase, however, is less than 1 bit and therefore small. Introducing $R5$ and $R6$ in the active circuit with 700 Ohm strain gage produces an additional but very small increase of noise such that the noise level is smaller than that of a strain gage with 120 Ω .

In summary, by including the resistors $R5$ and $R6$, the signal noise generally increases and circuits with 120 Ω strain gages are more sensitive than circuits with 700 Ω strain gages. The signal part of the records is contained in almost 15 bits which at the largest amplification gain leads to a resolution of approximately 1 $\mu\text{m}/\text{m}$ rms. This figure is still good enough for fatigue assessment of civil structures. The gain of 3 to 4 additional bits for the effective signal quantization that can be achieved with a 24-bit ADC with respect to a 12-bit ADC removes in practice the tradeoff between resolution and amplitude range. Clearly, for the tested AFE a 16-bit ADC would provide the same data quality. Since the AFE supports the signal conditioning of other sensors that may exhibit a smaller signal noise, a 24-bit ADC was still a better choice in terms of dynamic range.

3. Embedded Data Processing

Embedded data processing is a powerful tool for extending the lifetime of sensors nodes. Power saving is achieved by reducing significantly the data communication of data intensive monitoring applications. A variety of data processing algorithms were implemented and tested whose complexity depended on the performance of the WSN platform [14–18]. The feasibility and advantages of embedded data processing were successfully demonstrated in long term deployments [17, 19]. A strategy for fatigue life estimation for WSN with embedded data processing was proposed by O'Connor et al.

[20]. Strain data was continuously acquired and processed by a rainflow counting algorithm. Laboratory tests showed a good agreement between the cycle counting histograms obtained by embedded and offline data processing.

In this work, strain data is also permanently acquired. Embedded data processing is performed in two data processing steps. The first data processing step consists in verifying if after recording the raw data a given threshold is exceeded. If the threshold is not exceeded, the following data processing steps are skipped and the buffer with the recorded raw data is overwritten by a subsequent data record. On the contrary, if the threshold is exceeded, the recorded data is pipelined to the second data processing step.

The algorithm of the first data processing step is very simple and computes the minimum and maximum of a data record. The threshold is considered as exceeded if in a data record the absolute value of the difference between minimum or maximum is greater than the threshold. This simple software-triggering mechanism avoids the processing and transmission of idle data and thus contributes to power consumption reduction.

The second step consists of calculating the cycles of a record. Since in fatigue assessment methods based on cumulative fatigue damage or fatigue crack propagation the impact of small amplitude cycles is usually negligible, the algorithm removes all cycles that are smaller than a given threshold. Therefore, the algorithm works as a small amplitude cycle filter. As a side effect, all cycles that are generated by signal noise are removed too. The output is a sequence of local maxima and minima and can be used as input of a cycle counting algorithm (e.g., rainflow counting algorithm) to determine the statistics of the cycle's amplitudes. Cycle counting algorithms require as input a strain history where each data point is either a local minimum or maximum [21].

Since the cycle filtering algorithm performs two distinct tasks, finding all local minima and maxima and removing all cycles that have smaller amplitude than the threshold, each task could be implemented by a distinct algorithm. The first algorithm finds all local minima and maxima and the second removes the small cycles. The algorithm that finds the minima and maxima is trivial and does not have to be discussed. The pseudocode of the cycle filtering algorithm is given as follows:

- (1) Read the next data sample d_j (if out of data then stop)
- (2) If the vector of read data has less than 4 samples go to step (1)

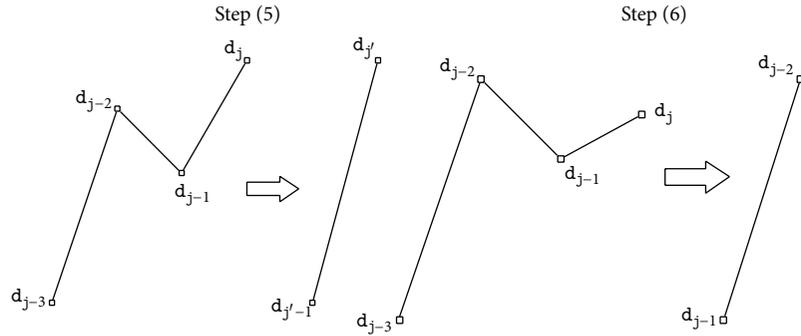


FIGURE 2: Removal of cycles by steps (5) and (6) of the cycle filtering algorithm.

- (3) Compute the segment widths $\Delta d_j = \text{abs}(d_j - d_{j-1})$ and $\Delta d_{j-1} = \text{abs}(d_{j-1} - d_{j-2})$. If Δd_{j-1} is smaller than the threshold go to step (4) else go to step (1)
- (4) If Δd_{j-1} is smaller than Δd_j go to step (5) else go to step (6)
- (5) Remove the samples d_{j-1} and d_{j-2} from the data vector and go to step (1)
- (6) Remove the samples d_j and d_{j-1} from the data vector and go to step (1)

In order to establish data continuity between subsequent records, the algorithm keeps the first and last sample of the data vector. This feature requires that a record consists of at least 4 samples. In addition, at the beginning or end of the output data vector a cycle may occur with an amplitude that is smaller than the threshold. These cycles can be removed from the processed data vector with a postprocessing step on the server.

The algorithm removes small cycles while processing the data vector. Cycle removal occurs in steps (5) and (6) depending on the width of the last two line segments (Figure 2). The two samples associated to the smaller line segment are removed from the data vector. By removing the cycles, the size of the data vector shrinks. Since sample removal does not affect the unprocessed data, the output can be stored in the same physical memory space as of the original raw data. The algorithm thus does not need additional memory except for storing temporary data such as segment widths and indices.

An alternative to the algorithm described above is a one-pass algorithm that performs simultaneously both tasks: extrema search and small cycle filtering. In this case the data consists of the raw data record and not the array of local minima and maxima. The core of the algorithm does not have to be changed except step (1), which reads extremal points and therefore has to be replaced. The pseudocode of the algorithm is as follows:

- (1) Read the next data sample x_k (if out of data then stop) If x_k is not an extremum go to step (1) else store x_k into the array of extrema d and go to step (2)
- (2) If the array of extrema d has less than 4 samples go to step (1)

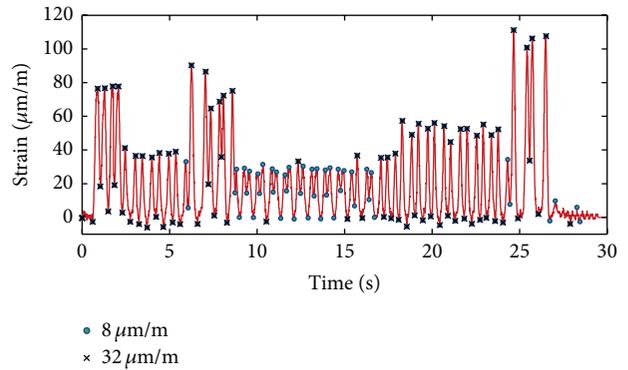


FIGURE 3: Strain record on a crossbeam.

- (3) Compute the differences $\Delta d_j = \text{abs}(d_j - d_{j-1})$ and $\Delta d_{j-1} = \text{abs}(d_{j-1} - d_{j-2})$. If Δd_{j-1} is smaller than the threshold go to step (4) else go to step (1)
- (4) If Δd_{j-1} is smaller than Δd_j go to step (5) else go to step (6)
- (5) Remove the samples d_{j-1} and d_{j-2} from the array of extrema and go to step (1)
- (6) Remove the samples d_j and d_{j-1} from the array of extrema and go to step (1)

It is important to make sure that, in the implementation of step (1), the array of extrema d contains only minima or maxima and not saddle points.

The outcome of the cycle filtering algorithm is displayed in Figure 3, which presents a typical strain time history where each axle passing over the crossbeam produces a cycle. The record was sampled with a sampling rate of 100 Hz and consists of 3000 samples. The extremal points identified by the algorithm are plotted with markers. Two different filter thresholds were used. The greater threshold is, as expected, more effective in reducing the data size (83 versus 130 samples). However, it misses identifying many cycles, in particular in the time span between 8 and 15 seconds. In this case, the smaller threshold reproduces much better the cycle pattern of the original record. The essential information about the event is contained in the 130 extremal points, which are only 2.8% of the original data size. Storing only the extremal



FIGURE 4: (a) Münchenstein Bridge and (b) crossbeams and longitudinal stringer supporting the sleepers.

TABLE 4: Processing time of the algorithms.

Algorithm	Description	Computation time [ms]
Software triggering	Returns data record if threshold is exceeded	4
Cycle filtering	Computes local extremes of a data record	83

points allows for a significant removal of redundant data. Such a dramatic data reduction to a few % of the original data size is typical for fatigue monitoring applications. Due to the irregular time interval between the extremal points, the time information is lost. However, since only strain cycle amplitude and frequency (cycle counting) are of concern for fatigue assessment, the loss of time information can be considered as negligible.

The algorithms were implemented in the software framework described by Flouri et al. [18]. The computation time of the two algorithms on a TI MSP430 microcontroller [22] is depicted in Table 4. A sample of 2048 samples is processed in less than 100 ms. The data processing is extremely fast and does not represent any limitation for the data acquisition since with the maximum sampling rate of 470 Hz the recording time is 4.36 seconds.

4. Field Test

The field test deployment was performed on a riveted railway bridge near Münchenstein in Switzerland. The Münchenstein Bridge was constructed in 1892 and crosses the river Birs. It replaced an older riveted bridge designed by Gustave Eiffel that collapsed in 1891. The bridge, which has a span width of approximately 45.2 m, consists of two lateral longitudinal trusses that are connected by crossbeams and cross-bracings (Figure 4(a)). The existing bridge supports are skewed at nearly 45 degrees. The railway sleepers are mounted on two longitudinal stringers that are connected at their ends to crossbeams (Figure 4(b)). These beams are particularly susceptible to fatigue loading since each passing axle produces a loading cycle with high strains. The bridge is mainly used by passenger trains that travel from Basel into

the valley of the river Birs. The trains travelling in the opposite direction use the twin bridge.

4.1. WSN Platform. The wireless sensor network is based on the commercial sensor node of the company Decentlab [23]. The core of the sensor node is the 16-bit microcontroller TI MSP430 from the family of reduced instruction set computing (RISC) mixed-signal microprocessors [22]. The most important features of this system-on-chip are the ultra-low power consumption $0.5 \mu\text{A}$ (standby mode), 116 kB of flash memory, 8 kB of RAM memory, and 16 MHz CPU speed. The output of the 24-bit ADC was truncated to 16 bits by removing the 8 less significant bits. This truncation did not have any effect on data quality since according to Tables 2 and 3 the rms of signal noise was approximately 8 bits. Wireless delivery of data is enabled using low power radio transceiver working in European SRD Band from 863 to 870 MHz. By operating in this frequency band, opposed to the conventionally used 2.4 GHz band, nodes can achieve better signal propagation in environments where there is no line of sight. The nominal transmission rate is 20 kbit/s. The sleep mode current consumption of the radio chip is $0.6 \mu\text{W}$, while during reception and transmission the consumption reaches 27.6 mW and 51 mW, respectively. The platform features a voltage stabilizer that provides a constant input voltage of 3 V. Fading batteries or voltage fluctuations due to temperature changes do not have any effect on the data quality.

4.2. Monitoring Setup. The deployment consisted of 5 sensor nodes, 2 relay nodes, and a base station. Each sensor node was connected to a strain gage. Two crossbeams were monitored, each with 2 strain gages. The strain was measured on the lower flange with a commercial strain checker [24] (Figure 5). With these devices the coupling between the structure and the strain gage is achieved through friction by pressing the strain gage on the structure using a magnet. This feature enabled a fast deployment. Two checkers were mounted on each crossbeam to enable a more reliable evaluation of the recorded strains. The strain checkers used a strain gage with 120 Ohm and 6 mm length. Another strain gage was mounted on one of the carbon fiber lamellas of the strengthening system [25]. This strain gage was glued and had a resistance of 700Ω (Figure 6). The housing of the sensor nodes was



FIGURE 5: Strain checkers mounted on a crossbeam.



FIGURE 7: Wireless sensor node.

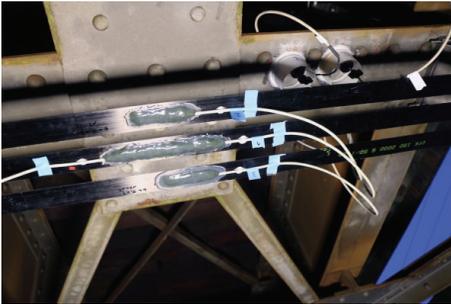


FIGURE 6: Strain sensors mounted on CFRP lamellas.

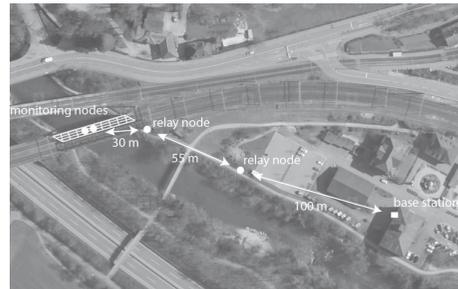


FIGURE 8: Overview of the overall deployment.

equipped with 4 magnetic footings, thus allowing a fast mounting on the steel crossbeam (Figure 7).

The base station was placed inside a building at a distance of nearly 200 m from the bridge and was powered by the main supply. Since another building was situated between the base station and the sensor nodes, two relay nodes were placed in between to improve the link quality (Figure 8). Since the steel bridge significantly reduced the range of the radio, the first relay node was placed nearby on the twin bridge and the second one was mounted on a pole, approximately in the middle distance between the bridge and the building hosting the base station.

4.3. Operation Mode. The sensor nodes were operated with a permanent data acquisition with a sampling frequency of 123 Hz. The recorded data were organized in records of 128 samples each, which corresponded to a recording time interval of 1.04 seconds. Such small records were used for testing purposes. Since a typical train crossing lasted between 4 and 8 seconds, 5 to 10 records were necessary to cover the event. The records were stored alternating in two distinct buffers. While one buffer was used to store the currently recorded data, the other buffer, which hosted the previous record, was used by the data processing algorithms. The threshold of the software triggering algorithm was set to $19 \mu\text{m/m}$. The threshold of the cycle filtering algorithm was set to $4.7 \mu\text{m/m}$. Assuming a module of elasticity of steel of 200 GPa this threshold corresponds to a stress amplitude of approximately 1 MPa.

The output of embedded processing was stored in packets consisting of maximum 16 data samples and metadata that

described the data packet (e.g., node number, sequence number, data type, etc.). This metadata was used by the data server to associate the incoming data to a specific sensor node in order to reconstruct the sequence of data in time. The data packets were finally forwarded to the radio for transmission. Data packets were carrying only output data of one raw data record. Because of the small sample size and significant data reduction rate the packets carried often less than the maximum allowable payload of 16 data samples. At arrival, the base station assigned a time stamp to each packet. This time stamp was used in the postprocessing for reconstructing the strain cycles in time of a particular train crossing event.

5. Results

5.1. Unprocessed Data. Unprocessed strain records measured on a steel crossbeam with a sensor node are shown in Figure 9. These records were acquired prior to start of the monitoring phase without performing embedded data processing in order to evaluate the quality of raw data. The raw data was stored in the external flash memory and eventually downloaded using a USB interface. Figure 9(a) displays the strain time history induced by an intercity train (ICN) and Figure 9(b) by a suburban train (SB). The strain cycles induced by the bogies are clearly distinguishable. The record demonstrates that the signal quality is very good. In fact, the noise is smaller as $1 \mu\text{m/m}$ rms and agrees well with the laboratory tests described in Section 2.3.

5.2. Processed Data. The processed data generated by an ICN train that was recorded by the monitoring system is displayed in Figure 10. The event occurred on November 21

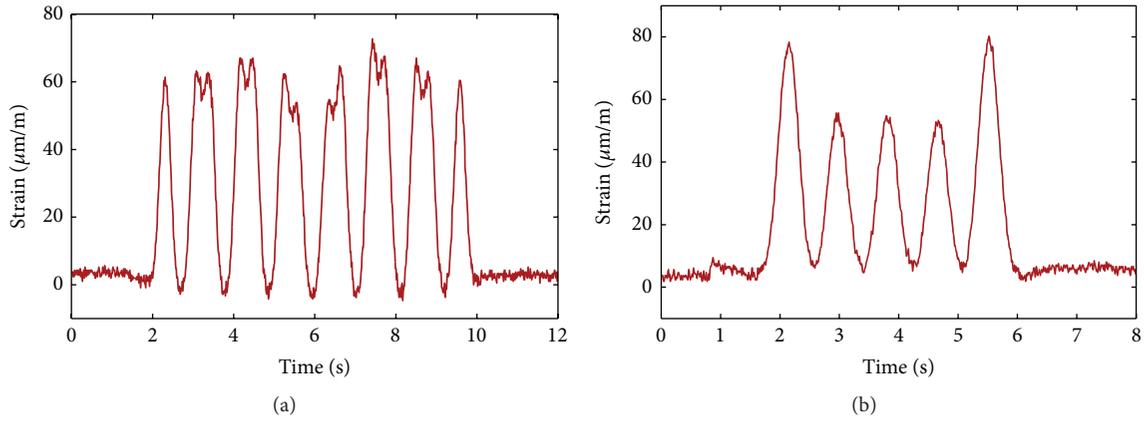


FIGURE 9: Typical strain records induced by an ICN train (a) and an SB train (b).

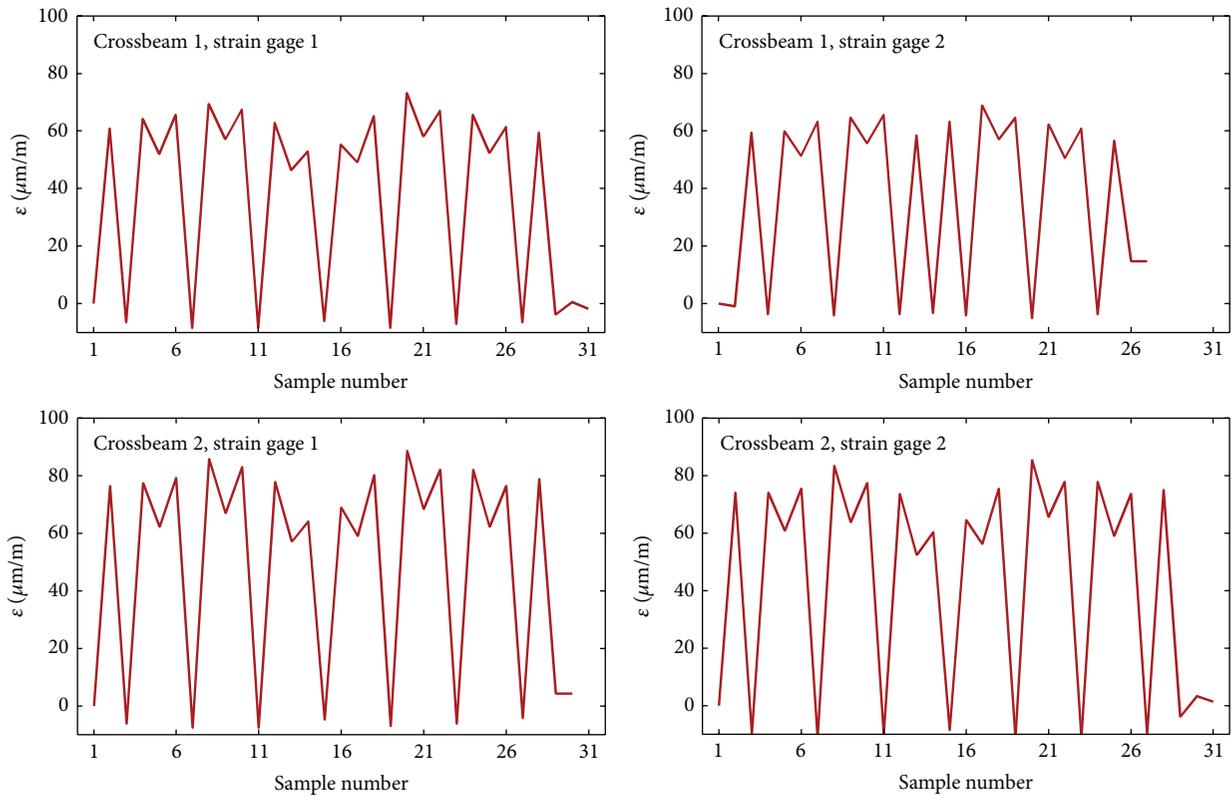


FIGURE 10: Records with strain cycles induced by an ICN train.

at 12:07. The train was similar to that generating the record that is displayed in Figure 9(a). The 8 bogies are clearly distinguishable on all 4 strain gages. Also the pattern of the cycles agrees well with the unprocessed record. Only the 4th and 5th cycles of the second strain gage, which was mounted on crossbeam 1, differ from all other records. This difference was probably induced by the threshold that cancelled the small amplitude cycles. In fact, these cycles in the other records had amplitudes varying between 6 and 8 $\mu\text{m/m}$ which is very close to the threshold (5 $\mu\text{m/m}$) of the cycle filtering algorithm. The record shows that strain

cycle amplitudes of crossbeam 2 are greater than the ones of crossbeam 1. The crossbeams are identical and vary only on their position. Since both strain amplitudes recorded on crossbeam 1 agree quite well, the greater amplitudes of crossbeam 2 should not be induced by a bad mounting of the strain checker but should be originated by a physical reason. An additional confirmation is provided by a finite element analysis which shows that crossbeams located at the bridge ends are subjected to greater cycle amplitudes rather than crossbeams located in the middle of the span [26].

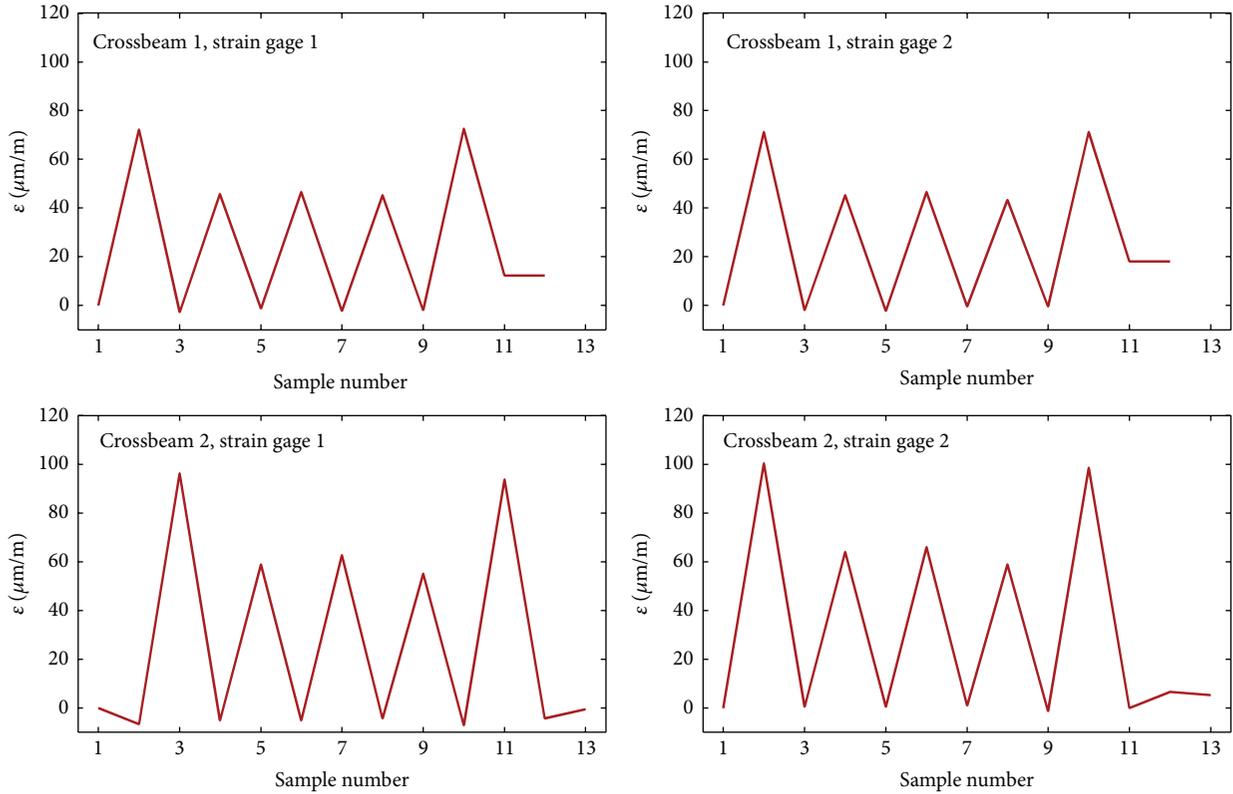


FIGURE 11: Records induced by an SB train.

The record of this event is described by 26 and 31 samples. The shortest record refers to the second strain gage of crossbeam 1 that does not show the small amplitude cycles on the 4th and 5th large amplitude cycles. Assuming that the train speed in this event is similar to the event displayed in Figure 9(a), the train passage lasted for approximately 8 seconds. Thus, even considering the extra samples resulting from the front and trail of each data record, the embedded processing reduced the communicated data to in average 6% of the original data size.

Analogously, the processed data of a suburban train that crossed the bridge on November 28 at 9:12 is displayed in Figure 11. Since suburban trains do not produce any small amplitude cycles, the cycle patterns of the all 4 records match exactly. The cycle pattern is characterized by 5 cycles having greater cycles in the front and rear and 3 smaller cycles in between. Also in this event, the strain cycle amplitudes of crossbeam 2 are greater than the one of crossbeam 1. The size of the event records consists of 12 samples. Assuming a crossing time of approximately 4.4 seconds (Figure 9(b)), the embedded processing reduces the communicated data to in average 8% of the original data size. It is possible to achieve further reduction by increasing the period of a single record. In this way, extra samples in the front and trail of a processed record are minimized by reducing the number of records forming a complete event.

5.3. Reliability. Reliability is always an important issue when operating such complex distributed systems like wireless

sensor networks over a significant period of time. Poor reliability is equivalent to data loss. Data loss per se does not necessarily imply a loss of information. The data loss that was experienced in the vibration monitoring of a footbridge [17] would not have any impact on the vibration serviceability assessment. If the data loss is moderate and is governed by a random process and the data pool is large enough, the exceedance probability of peak vibration amplitudes would not change. In this case, data loss produces only a loss of efficiency as more time is needed to achieve a representative data pool.

However, in fatigue life estimation the situation is different. Since data loss reduces the number of recorded cycles, any assessment based on cycle counting will be biased. Therefore, reliable information about data loss is important in order to be able to estimate their effect on the assessment.

5.3.1. Event Hit Rate. A first insight into system reliability is provided by analysing the recorded events. The events were identified using the time stamps of the data packets. An overview of the recorded events and their time of occurrence is displayed in Figure 12. In this figure, green circles represent events that were recorded by all nodes (complete events). Blue circles represent events that were not recorded by at least one node (incomplete events). Such a failure does not necessarily mean that the triggering algorithm missed identifying a train. Data loss due to poor communication could be as well a possible cause. The figures on the right-hand side are the number of total recorded daily events (complete and

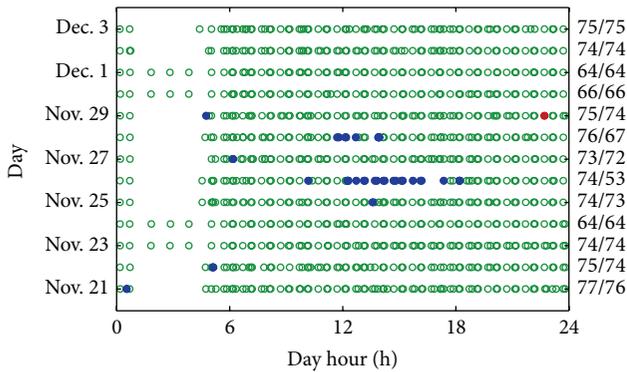


FIGURE 12: Recorded train crossing events.

incomplete) and the number of recorded complete events during the day. According to the service timetable of the Swiss Railways, on working days 59 passenger trains (41 suburban trains and 18 intercity trains) are crossing the bridge. On Saturdays and Sundays there are 3 additional suburban trains between 1 and 4 am. The recorded numbers of trains per day, however, do not fit with this timetable. This discrepancy is due to intercity trains that drove over the monitored bridge, instead of being routed as planned over the twin bridge.

The majority of incomplete events (blue disc) are concentrated on two days: November 26 and 28. On November 26, the strengthening of the crossbeam was deployed and the two sensor nodes that were mounted on this crossbeam were removed between 12 and 18 o'clock. On November 28, between 11 and 14 o'clock, other work related to strengthening was performed and sensor nodes were removed. The number of incompletely detected events that occurred outside the periods of strengthening work is 6 (Figure 12). According to the service timetable of the Swiss Railways, only two of these events, on November 26 at 10:10 am and on November 27 at 6:11 am, can be associated with passenger trains. Based on the recorded data, 4 remaining incompletely recorded events are service trains. On November 29, the suburban train at 10:42 pm was not recorded (red marker). A failure of all 5 sensor nodes or a detour of this train over the twin bridge may be the cause. Even if considering this missed train as a failure of the triggering mechanism, its failure rate was about 0.6%.

5.3.2. Data Loss. An additional source of data loss is the communication. In this deployment, setting up a good communication link between the sensor nodes mounted on the bridge and the base station was a challenging task. The missing line of sight between the bridge and the building that hosted the base station and the highly reflective metallic construction of the bridge were the main causes of the communication problems. Two relay nodes had to be placed between the bridge and the building to assure a sufficiently reliable data link.

The communication protocol used for delivering data is based on the collection tree protocol (CTP) [23]. It is primarily designed for low traffic rates and does not promise a 100% safe delivery of data packets. Although it contains

several mechanisms that improve reliability, CTP only promises that it will “try very hard” to deliver each packet.

When a monitoring system performs periodical measurements with constant time intervals, the estimation of data loss is very simple since the theoretical number of data records is given by monitoring period and the time interval between consecutive measurements. The data loss is calculated as the difference between the theoretical data samples and the effectively received ones. An estimate of the data loss of a monitoring system operating in event driven mode is much more difficult since there is no constant time interval between two events and even the data size varies from event to event.

In the current investigation, the content of the data records can be used to estimate the data loss. The passenger trains are usually operated with a few standard configurations. In this particular case, the recorded data revealed that the suburban trains crossing the Münchenstein Bridge operated with 3 different configurations. Since each train configuration produces a very specific cycle pattern, data loss becomes identifiable by a deviation from these cycle patterns. The 3 cycle patterns are displayed in Figure 13. Each complete cycle pattern is characterized by the number of cycles or, equivalently, the sample size: 11 data points for configuration 1, 21 data points for configuration 2, and 31 data points for configuration 3.

The data loss can thus be estimated by computing missing data points for each recorded event. The analysis was restricted to the suburban trains as their cycle patterns are much simpler than the cycle patterns of the intercity trains. 38 suburban trains crossed the bridge regularly each day. These trains represent more than 50% of the total number of trains recorded during the monitoring period and are therefore a representative sample for estimating the data loss. The data loss analysis was also restricted to the 4 nodes that were operated during the complete monitoring period.

The observed average data loss of all 4 nodes was 4.2%. The individual data losses of the nodes were 3.9%, 4.1%, 4.2%, and 4.7%. These figures do not differ significantly, hence suggesting that, in average, all nodes performed equally well. The observed data loss is in accordance with other investigations that addressed data loss of WSN monitoring systems [17].

The distribution of data loss over the day does not show particular patterns such as better network performance during the night (Figure 14(a)). This data loss was computed by averaging the data loss of the trains scheduled at a fixed time during the monitoring period. The evolution of average data loss with monitoring time is displayed in Figure 14(b). The days before the implementation of the strengthening and the installation of the 5th node, which occurred on day 6, are characterized by a lower average data loss and a smaller scattering than the last 7 days. In this period, the average data loss of all 5 nodes was 5.2% and the individual average data losses of the nodes were 4.8%, 5.7%, 5.1%, 5.9%, and 5.2%. These figures are 1% higher than the figures of the first 6 days with 4 operating sensor nodes. The observed performance drop may be caused by the 5th node which shared the same transmission channel as the 4 other nodes, thus increasing the risk of packet collision.

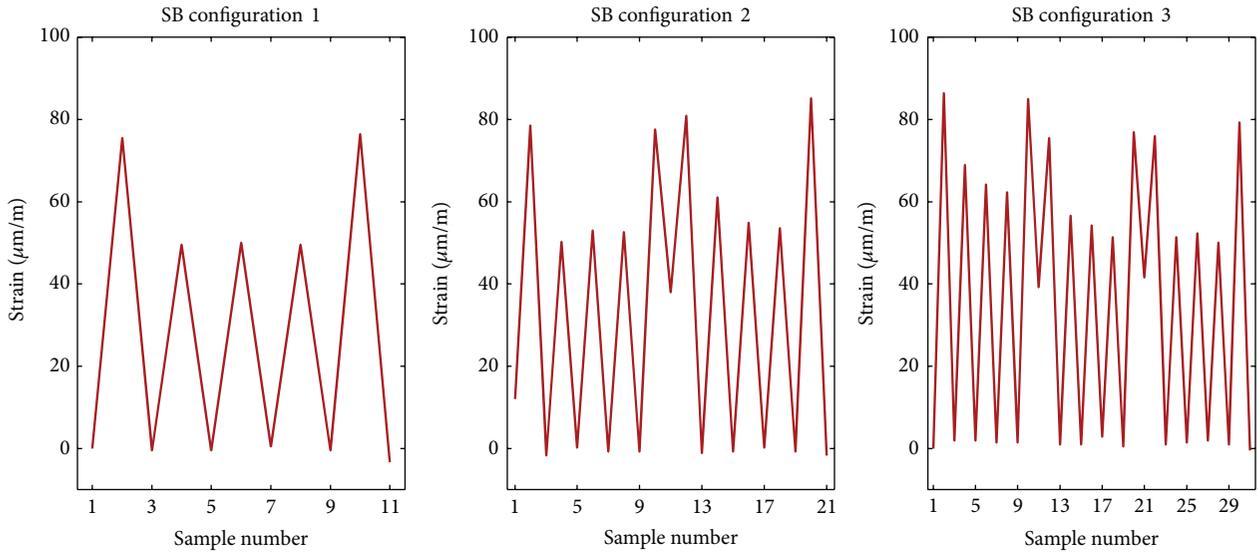


FIGURE 13: Cycle patterns of suburban train configurations.

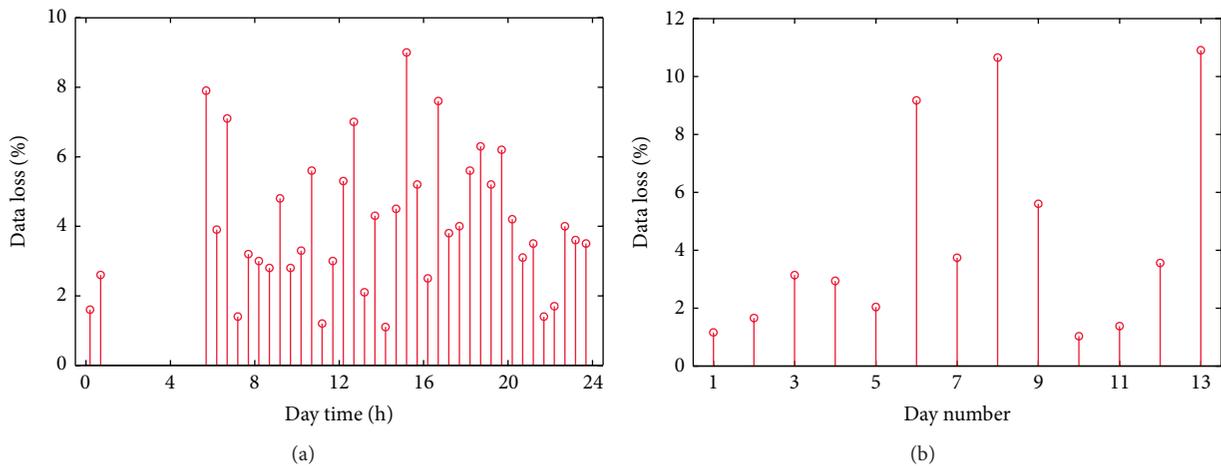


FIGURE 14: Distribution of data loss over day time (a) and monitoring days (b).

Due to the small difference in data delivery performance and the short deployment period of 13 days, an interpretation of the data in terms of poorer performance in the second half of the monitoring period would be unreliable. Such a trend was clearly observed in [17] for a monitoring period of one year.

The analysis of the data loss per train configuration displays, in contrast, an evident pattern. Trains characterized by 11 samples had an average data loss of 3.1%. Those characterized by 21 samples had a data loss of 5.1% and the longest train with 31 samples had a data loss of 8.1%. The data loss increases with the train length with an approximately constant gradient of 2.5% per 10 samples. An analogue analysis using IC trains reveals that the shorter configuration, described by 17 samples, had a data loss of 3.9%, while the longer configuration, described by 31 samples, had a data loss of 4.6%. IC trains, hence, show also an increase of the data loss with longer trains.

The increase rate of data loss cannot be explained by a random data loss. In fact, a longer train produces more data to be transferred. The greater amount of data, however, is generated over a longer time period since a longer train needs more time to cross the bridge. On the other hand, the data rate per time interval is constant which should lead to a data loss that is independent of the event. Considering the position of the sensor nodes relative to the first relay node and the highly reflective metallic environment, a realistic source of the correlation of data loss with train length can be identified. While, in order to prevent acts of vandalism, the sensor nodes are deployed around 1 meter below the rails, the first relay node was placed approximately on the same level of the rails so that there was no line of sight between sensor nodes and relay node. Communication signals reached the relay node over multiple reflections on the metallic structure and not by a direct path. Due to their metallic hull, trains modified the reflection properties of the environment. This might have

disturbed the link quality between sensor nodes and relay node. Due to the short recording period, most of the data packets were transmitted during the crossing of a train, thus resulting in a data loss that was correlated to train length.

6. Conclusions

This paper investigated different mechanisms for improving the performance of WSNs when used for monitoring strain cycle on civil structures such as railway bridges. The first performance improvement is achieved with a signal conditioning board that allows reducing significantly the power consumption of a sensor node. The improvement concerning power consumption is correlated to a reduction in sensitivity and therefore a reduction of signal-to-noise ratio. This drawback, however, is neutralized by the 24-bit ADC which allows achieving high resolution without penalizing the amplitude range. The achieved noise level of about $1 \mu\text{m/m}$ rms confirms that the resolution matches well the requirements of engineering practice. The sampling rate of 470 Hz ADC is not an impressive figure but hardly represents a limitation for strain cycles monitoring applications on civil structures.

Embedded data processing represents the second performance improvement. Because of the particular cycle structure produced by trains and the limited duration of the event, the cycle filtering algorithm achieves a data reduction by roughly a factor 10 to a factor 20 (5% to 10% of the original data size). In this investigation, the embedded data processing was ended with the cycle filtering algorithm. A cycle counting algorithm (rainflow algorithm) could be easily implemented and appended as a subsequent data processing step. Because of its simplicity and the small input data size, it would be even faster than the cycle filtering algorithm. Therefore, the total processing time would still be much smaller than the time to acquire raw data records. A cycle counting algorithm would further reduce the data size by roughly a factor 2, which, in terms of samples, is roughly 10 times less than what the cycle filtering algorithm achieves. However, in order to be applied correctly, a cycle counting algorithm must be executed on the full strain time history of an event [21]. Because of the memory limitations of WSN nodes, a complete event is likely to consist of several records. This requires temporarily storage of intermediate results of the cycle filtering algorithm and eventually a reallocation of this data into memory when a new data record is processed. These operations have to be repeated until the last record of the same event is processed. A further data reduction by a roughly factor 2 does not produce such a great energy saving to justify the greater complexity of the embedded data processing algorithm. The cycle counting algorithm can be easily performed on the data server. The input consists of the output of the cycle filtering algorithm that is linked to an event by time stamps. This choice keeps the embedded data processing on the sensor node simple, straightforward, and finally more reliable. Furthermore, collecting minima and maxima of a strain history enables a better verification of the performance of the monitoring system in terms of data quality and reliability rather than a histogram of data cycles.

Finally, the information provided by minima and maxima of a strain history can be used for calibrating structural models.

A last performance improvement is achieved with the software triggering mechanism. It is fast, simple, and reliable and prevents the processing and communication of useless raw data records. On the Muenchenstein Bridge the total time period of train crossings sums up to roughly 10 minutes per day. This figure corresponds to less than 1% of the raw data that was acquired in a day. Thus, in terms of minimizing data processing, the software triggering mechanism proved to be very efficient. However, the extremely small size of relevant data compared to the acquired data reveals also the intrinsic weakness of the software triggering mechanism. An event triggering mechanism based on ultra-low power auxiliary sensor nodes that trigger the strain sensor nodes only when an event occurs would be more efficient. According to the authors' experience, the lifetime of a sensor node with the software triggering mechanism and powered by two commercial 1.5 V alkaline D batteries having a capacity of 18.000 mAh is about 50 days. The data provided by such a monitoring period is often already sufficient for fatigue assessment purposes.

The data loss occurring during the field test was moderate although the situation with the highly reflexive metallic bridge and the long distance to the base station was rather difficult in terms of communication link quality. However, such an environment is likely to be representative for strain monitoring of railway bridges. The locations of the sensor nodes and relay nodes were identified by trial and error until a stable connectivity was achieved. Better layouts in terms of link quality are likely to exist. Quantitative tools for evaluating concurrent layouts would be very helpful to improve link quality and reduce data loss. Nevertheless, a moderate average data loss of several % does not necessarily invalidate the monitoring campaign. If data loss is governed by random processes and the monitoring period is sufficiently long the histogram of strain cycle amplitudes is not significantly affected by data loss. In cases where train configurations are highly standardized, increasing the number of cycles by the data loss rate provides a good estimate of the unbiased result. A small data loss is therefore acceptable and does not bias the fatigue assessment.

To conclude, although there is still space for improvements in the field of event triggering, layout optimization, and data loss, this investigation demonstrates that a wireless sensor network has the potential to perform strain cycles monitoring of metallic railway bridges in an effective and efficient way.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This investigation was part of the research projects PSRP-124/2010 titled Innovative Structural Health Monitoring in

Civil Engineering Infrastructure Sustainability, Tulcoempa, funded by the Polish Swiss Research Programme 2009–2017, and the project Hydronet 2, funded by Swisselectric Research and the Competence Center Energy and Mobility. The authors express their gratitude to the funding agencies and to the Swiss Federal Railways for their support during the field tests on the Münchenstein Bridge. In addition, they would like to thank their former colleagues Rouven Christen and Rocio Maldonado for their important contributions.

References

- [1] X. W. Ye, Y. Q. Ni, K. Y. Wong, and J. M. Ko, "Statistical analysis of stress spectra for fatigue life assessment of steel bridges with structural health monitoring data," *Engineering Structures*, vol. 45, pp. 166–176, 2012.
- [2] J. Leander, A. Andersson, and R. Karoumi, "Monitoring and enhanced fatigue evaluation of a steel railway bridge," *Engineering Structures*, vol. 32, no. 3, pp. 854–863, 2010.
- [3] M. Bosshard, P. Steck, C. Meyer, E. Brühwiler, M. Tschumi, and S. Haldimann, "Fatigue safety of riveted bridges—part 2: verification based on the monitoring data of the project 'Railway Bridge at Eglisau,'" *Stahlbau*, vol. 81, no. 11, pp. 868–874, 2012.
- [4] Z. X. Li and T. H. T. Chan, "Fatigue criteria for integrity assessment of long-span steel bridge with health monitoring," *Theoretical and Applied Fracture Mechanics*, vol. 46, no. 2, pp. 114–127, 2006.
- [5] P. Wang, Y. Yan, G. Y. Tian, O. Bouzid, and Z. Ding, "Investigation of wireless sensor networks for structural health monitoring," *Journal of Sensors*, vol. 2012, Article ID 156329, 7 pages, 2012.
- [6] K. Chebroly, B. Raman, N. Mishra, P. K. Valiveti, and R. Kumar, "BriMon: a sensor network system for railway bridge monitoring," in *Proceedings of the 6th International Conference on Mobile Systems, Applications, and Services (Mobisys '08)*, pp. 2–14, June 2008.
- [7] R. Bischoff, J. Meyer, O. Enochsson, G. Feltrin, and L. Elfgren, "Event-based strain monitoring on a railway bridge with a wireless sensor network," in *Proceedings of the 4th International Conference on Structural Health Monitoring on Intelligent Infrastructure (SHMII-4 '09)*, Zürich, Switzerland, July 2009.
- [8] T. Nagayama, M. Ruiz-Sandoval, B. F. Spencer Jr., K. A. Mechitov, and G. Agha, "Wireless strain sensor development for civil infrastructure," in *Proceedings of the 1st International Workshop on Networked Sensing Systems*, pp. 97–100, 2004.
- [9] H. Choi, S. Choi, and H. Cha, "Structural health monitoring system based on strain gauge enabled wireless sensor nodes," in *Proceedings of the 5th International Conference on Networked Sensing Systems (INSS '08)*, pp. 211–214, IEEE, Kanazawa, Japan, June 2008.
- [10] M. J. Whelan and K. D. Janoyan, "Design of a robust, high-rate wireless sensor network for static and dynamic structural monitoring," *Journal of Intelligent Material Systems and Structures*, vol. 20, no. 7, pp. 849–863, 2009.
- [11] H. Jo, J.-W. Park, B. F. Spencer Jr., and H.-J. Jung, "Development of high-sensitivity wireless strain sensor for structural health monitoring," *Smart Structures and Systems*, vol. 11, no. 5, pp. 477–496, 2013.
- [12] D. A. Howell and H. W. Shenton III, "System for in-service strain monitoring of ordinary bridges," *Journal of Bridge Engineering*, vol. 11, no. 6, pp. 673–680, 2006.
- [13] Analog Devices, *6-Channel, Low Noise, Low Power, 24-/16-Bit, Σ - Δ ADC with On-Chip in-AMP and Reference*, Analog Devices, Norwood, Mass, USA, 2007.
- [14] A. T. Zimmerman, M. Shirashi, R. A. Swartz, and J. P. Lynch, "Automated modal parameter estimation by parallel processing within wireless monitoring systems," *Journal of Infrastructure Systems*, vol. 14, no. 1, pp. 102–113, 2008.
- [15] S.-H. Sim, J. F. Carbonell-Márquez, B. F. Spencer Jr., and H. Jo, "Decentralized random decrement technique for efficient data aggregation and system identification in wireless smart sensor networks," *Probabilistic Engineering Mechanics*, vol. 26, no. 1, pp. 81–91, 2011.
- [16] J. P. Lynch, Y. Wang, K. J. Loh, J.-H. Yi, and C.-B. Yun, "Performance monitoring of the Geumdang bridge using a dense network of high-resolution wireless sensors," *Smart Materials and Structures*, vol. 15, no. 6, pp. 1561–1575, 2006.
- [17] G. Feltrin, K. E. Jalsan, and K. Flouri, "Vibration monitoring of a footbridge with a wireless sensor network," *Journal of Vibration and Control*, vol. 19, no. 15, pp. 2285–2300, 2013.
- [18] K. Flouri, O. Saukh, R. Sauter et al., "A versatile software architecture for civil structure monitoring with wireless sensor networks," *Smart Structures and Systems*, vol. 10, no. 3, pp. 209–228, 2012.
- [19] G. Feltrin, J. Meyer, R. Bischoff, and M. Motavalli, "Long-term monitoring of cable stays with a wireless sensor network," *Structure and Infrastructure Engineering*, vol. 6, no. 5, pp. 535–548, 2010.
- [20] S. O'Connor, J. Kim, J. P. Lynch, K. H. Law, and L. Salvino, "Fatigue life monitoring of metallic structures by decentralized rainflow counting embedded in a wireless sensor network," in *Proceedings of the ASME Conference on Smart Materials, Adaptive Structures and Intelligent Systems*, vol. 2, Philadelphia, Pa, USA, September-October 2010.
- [21] S. D. Downing and D. F. Socie, "Simple rainflow counting algorithms," *International Journal of Fatigue*, vol. 4, no. 1, pp. 31–40, 1982.
- [22] Texas Instruments Incorporated, *MSP430 Ultra-Low-Power Microcontrollers*, Texas Instruments Incorporated, Dallas, Tex, USA, 2014.
- [23] Decentlab GmbH, *Single Channel Sensor Node*, Decentlab GmbH, Dübendorf, Switzerland, 2014.
- [24] Tokyo Sokki Kenkyujo, *FGMH-I, Strain Checker*, Tokyo Sokki Kenkyujo, Tokyo, Japan, 2005.
- [25] E. Ghafoori, M. Motavalli, A. Nussbaumer, A. Herwig, G. S. Prinz, and M. Fontana, "Design criterion for fatigue strengthening of riveted beams in a 120-year-old railway metallic bridge using pre-stressed CFRP plates," *Composites Part B: Engineering*, vol. 68, pp. 1–13, 2015.
- [26] E. Ghafoori, G. S. Prinz, E. Mayor et al., "Finite element analysis for fatigue damage reduction in metallic riveted bridges using pre-stressed CFRP plates," *Polymers*, vol. 6, no. 4, pp. 1096–1118, 2014.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

