

Developing a Pattern-Based Method for Detecting Defective Sensors in an Instrumented Bridge

Mohammad Sajjadul Islam¹, Ashutosh Bagchi² and Aly Said³

Abstract

It is important to assure the reliability of a structural health monitoring system before interpreting the monitoring data for the detection of structural anomalies. Finding a malfunctioning component such as a sensor is an important step in that direction. Damage detection techniques in civil structures fall in the following two categories: data-driven and structural model-based. The data-driven methods provide a direct approach to damage assessment in a structure without creating any structural model (e.g. finite element model). Existence of damage and its location are interpreted by pattern matching of the data series of strain gauges, and temperature gauges at different time ranges. The objective of this study was to explore such methods, including the autoregressive exogenous (ARX) model, and based on that, develop new techniques to detect defective sensors. As a case study, the structural health monitoring data from the Portage Creek Bridge, located in the British Columbia, Canada was utilized to assess the conditions a set of sensors in of an instrumented pier, using methods developed based on the concepts of the sequential and binary search techniques. Continuous data sets of strain and temperature gauges were filtered and normalized. Defective sensors were detected by pattern matching of simulated and real data, using sensitivity analyses of the developed models.

1. Introduction

According to Mirza and Haider (2003), more than 40% of the bridges in service in Canada are over 30 years old. Some of these bridges are in urgent need of diagnosis, rehabilitation, or even partial re-construction in order to make them adequately safe for traffic and prevent sudden collapse and down time. Chase and Washer (1997) conducted a similar survey for the bridges in the U.S., and found that about 187,000 bridges – representing more than 25% of all bridges – were found deficient at that time; in addition, about 5,000 bridges were becoming deficient every year. The above estimate was an improvement over previous years because of increased federal funding for building and rehabilitation of bridges.

A more recent statistics by the Research and Innovative Technology Administration (RITA 2007) put the number of deficient bridges to about 12% of the total national and state bridges in the U.S. Most of these bridges were built before 1970, and their health condition is yet to be determined by any instrumental and scientific approach. The reduction from 25% in 1997 to 12% in 2007 was perhaps due to the continued reconstruction and rehabilitation efforts, as mentioned earlier.

A study by Dubin and Yanev (2001) funded by the Federal Highway Administration (FHWA) concluded that visual inspection of bridges was labour-intensive and cost-intensive, and Phares (2001) found that visual inspections were subjective and unreliable. Furthermore, the effectiveness of these periodic inspections was highly constrained by the limited ability for timely damage detection. Lu (2008) concluded that to maintain overall highway operational safety with current funding limitations, development of bridge structural health monitoring (SHM) systems that are continuous, automatic, and low cost was greatly and urgently needed. Due

¹ PhD Student, University of Nevada, Las Vegas, USA

² Associate Professor, Concordia University, Montreal, Canada (Corresponding Author)

³ Associate Professor, University of Nevada, Las Vegas, USA

to the rapid development of computing, sensing, and telecommunication technology during the last two decades, computer-based, long-term SHM systems are becoming more widely utilized to provide timely information on a structure's condition. An SHM system includes data acquisition, storage, and broadcasting; often, it is used to continuously evaluate the status of an entire structure or structural component. Hence, monitoring a structure can assist in assessing structural conditions and detecting damage in structures caused by such factors as cracking, corrosion, reduction in material properties, reinforcement rupture fractured welds, and loosened bolts.

According to Sohn et al. (2000), sensors that measured strains and vibrations of a structure produced signals that always respond to changes in environmental and operational conditions. Each group of signals could be considered a pattern that has some relation to the structural and ambient conditions. These researchers proposed that if the effect of ambient conditions to the patterns was normalized, the strain and vibration measurements should be identical or close to one another for a similar vibration effect as long as the structural vibration property remained the same. Changes in physical properties, mainly stiffness, should reflect in the processed signal blocks or patterns.

While the above studies indicate the use and development of the pattern recognition techniques for assessing damage in a structure, the main objective of this paper was to develop new pattern recognition techniques to detect malfunctioning sensors in an instrumented bridge pier. Although damage detection was not a goal of this paper, references to 'damage detection' were made frequently in the discussion on the existing pattern recognition techniques as they were developed for the purpose of damage detection.

2. Pattern Recognition

The ability of a computer to identify and classify patterns in order to make a decision is an important aspect of the machine learning process. The feature that makes it attractive to create an automated SHM system is pattern recognition. Fugate et al. (2001) proposed a generalized approach for SHM by statistical pattern recognition that involves five steps: operational evaluation, data acquisition, data cleansing, data normalization, and statistical model development.

2.1 Operational Evaluation

The first step, operational evaluation, defines damage for the system being monitored. It also determines the operational and environmental conditions under which the system structure functions.

2.2 Data Acquisition

The second step, data acquisition, involves selecting the types of sensors and their placement location. The optimal number of sensors to be used are determined as well as the setup of data acquisition, storage, or transmission hardware.

2.3 Data Cleansing

Non-structural conditions, such as temperature, always vary with time. Sources that affect the variation of data and the structure monitored need to be identified and minimized. Usually, signals are gathered continuously. Strain data is significantly influenced by temperature and external loading, and data needs to be corrected for all of the external noises affecting the signals.

2.4 Data Normalization

It is necessary to normalize the data to make it compatible damage detection analysis. In the case of varying environmental or operational conditions, normalized data can be compared at similar times during an environmental or operational cycle. Sohn et al. (2001) suggested normalizing and standardizing the time history data $x(t)$ by means of the following equation:

$$x = \frac{x - \mu}{\sigma}, \quad (\text{Eq. 1})$$

where μ is the mean of the signal and σ is its standard deviation.

2.5. Model Development

Statistical model development is concerned with the implementation of algorithms that analyze the distribution of the extracted features in an effort to determine the damage state of the structure. Appropriate algorithms to use depend on their ability to perform supervised and unsupervised learning. ‘Supervised learning’ refers to the case where examples of data from damaged and undamaged structures are available. ‘Unsupervised learning’ refers to the case where data is only available from the undamaged structure (Da Silva et al., 2007).

For linear prediction of damage diagnosis in structural systems, unsupervised learning uses the following two models: a) a one-step auto-regressive (AR) model and b) a two-step prediction model that combines auto-regressive (AR) and auto-regressive with exogenous inputs (ARX) techniques. Sohn et al. (2000) used an approach that detects damage based on monitoring of residual error, which is the difference between the actual acceleration measurements for the new signal and the prediction obtained from the AR and ARX model developed from the selected reference signal.

In this study, the ARX model for unsupervised learning was utilized. Further information on ARX modeling and applications in SHM can be found elsewhere (Erdoğan and Güral, 2009; Fugate et al., 2001; Yang and Makis, 2010; Zhang, 2007; Yao and Pakzad, 2012).

2.6. Autoregressive with Exogenous Inputs (ARX) Model

The basic input-output configuration of the ARX model is shown in Figure 1. Assuming a unit sampling interval, there is an input quantity or signal and an output quantity or signal; the signals are related by a linear system. The model is designed to establish a relationship among the data pattern represented in the input and output signals by means of a statistical system identification process (Zhang, 2007). This model can be represented by the following equation:

$$X(n) = \sum_{i=1}^a \alpha(i)X(n-i) + \sum_{j=1}^b \beta(j)u(n-k-j+1) + e(n), \quad (\text{Eq. 2})$$

where $\alpha(i)$ and $\beta(j)$ are the ARX model coefficients; a and b are the auto-regressive order and the exogenous order, respectively; k is the pure time delay from input to output; $e(n)$ is the residual term, modeling the disturbances that act on the input-output process and assumed to be white; and $u(n)$ is prediction error of the linear prediction processor, assuming that the prediction error is caused mainly by unknown external inputs.

ARX model must be trained with a set of known input and output data such that the difference between the model output and the actual output is minimized, using the Least Square method. The Least Square method is the most efficient polynomial estimation method because it solves linear regression equations analytically. For linear models, model estimation can be done using the time domain or the frequency domain data. For nonlinear models, only time-domain data can be used for both estimation and validation.

Validation is the process by which a trained model is tested for its performance, using a set of input and output data not previously seen by the model during the training process. If the model output and the actual output from the validation set are close to each other, as determined using a predefined comparison norm (e.g., the parameter r in Eq. 3), the model is said to be validated. Otherwise, it must be retrained with an expanded set of data. Then, a validated model can be used for estimation or prediction of the output data at a future time when the input data is available.

The parameter used for estimating the degree of closeness between the measured and simulated output data is:

$$\text{Best fit, } r = \left(1 - \frac{y - y_1}{y - y_2} \right) \times 100, \quad (\text{Eq. 3})$$

where, y is the measured output, y_1 is the simulated or predicted model output, and y_2 is the mean of y . The value of 100% corresponds to a perfect fit, while 0% indicates that the fit is no better than guessing the output to

be a constant ($y_1 = y_2$). Because of the definition of Best Fit, it is possible for this value to be negative. A negative value of r is worse than 0% and can occur because the estimation algorithm failed to converge.

3. Details of the Monitored Structure

In this current study, the Portage Creek Bridge – located in the City of Victoria, British Columbia (BC), Canada – was used as a case study and SHM data source (Figure 2). The bridge is a 124-m (407-ft) long, three-span structure with a reinforced concrete deck supported on two reinforced concrete piers and having abutments on H piles. The deck has a roadway width of 16.2 m (53 ft) with two 1.98 m (6'-6") sidewalks and aluminum railings. Eight bi-directional rosette type electrical strain gauges are located on each column, as shown in Figure 3. Also, a temperature sensor is located on Column 2 (Figure 3). Each bi-directional strain gauge has two channels, one for horizontal strain and another for vertical strain, producing a total of 16 sets of strain data for each column.

As shown in Figure 3, for this study, each strain channel was referred to as a separate sensor, with Sensor 1 through 16 in Column 1 and Sensor 17 through 32 in Column 2. The strain data from the 16 sensors in Column 2 (#17 through #32), and the temperature sensor, T, were used in this study. The sensor data from the instrumented bridge was obtained from the Intelligent Sensing for Innovative Structure (ISIS) Canada Research Network.

4. Methodology

This section briefly describes the data harvesting procedure and introduces well-known sequential and binary search techniques, followed by a description of the methods that were developed using the previously mentioned search techniques.

4.1. Data Harvesting

For this study, the data available from the bridge site was recorded by ISIS Canada from 2004 to 2006. The sampling rate of the data was 32 Hz, and the length of each data segment was 256 points (8 s). The time duration of each block of data retrieved from the database was 8 seconds, since a vehicle passing on the bridge was never found to take more than this amount of time (Huffman, 2006).

The following steps have been followed in this study to harvest data from the database:

1. The time interval between points, the number of data points, and the data channels to query were selected.
2. Data points were saved as comma-separated values.
3. Data collected in every second was converted into 'second' data by averaging the 32 data points in a second; the 'second' data were converted into 'minute' data, and the 'minute' data were converted into hourly data ($\mu\epsilon/\text{hr}$).
4. In order to train the statistical model for this study, the data from the months of December 2005, January 2006, and February 2006 were utilized. The total number of data points for each strain gauge sensor was 2160, out of which only 1738 were valid; the rest were Not a Number (NaN), and were removed from the data set. Then, the data was normalized using the method described earlier.
5. The testing data were taken from the month of March 2006, due to the presence of novel events or heavy traffic volume during that period.

4.2. Binary Search

'Binary search' is an algorithm for locating the position of an element in a sorted list; this is done by checking the middle, eliminating half of the list from consideration, and then performing the search on the remaining half. If the middle element is equal to the sought value, then the position is considered to be found; otherwise, the upper half or lower half is chosen for a search, based on whether the element is greater than or less than the middle element. The method reduces the number of elements needed to be checked by a factor of two each time, and finds the target value.

4.3. Sequential Search

‘Linear search’, also known as sequential search, is a search algorithm that is suitable for searching a list of data for a particular value. It operates by checking every element of a list one at a time in sequence until a match is found. The best case is that the value is equal to the first element tested, in which case only one comparison is needed. The worst case is that the value is not in the list or else it appears only once at the end of the list; in this case, n comparisons are needed, where n is the number of items in the list.

If just a few elements are to be searched, the linear search method is less trouble than more complex methods that require preparation, such as sorting the list to be searched, or more complex data structures; this is especially true when entries may be subject to frequent revision. Also, when certain values are much more likely to be searched for than other values, a linear search can arrange such values to be among the first to be considered in the list.

4.4 Analogy of the Sequential and Binary Search Methods as Applied in the Present Work

When a statistical model, such as an ARX model, was constructed and trained with the input and output data from the sensors, the relationship among the data patterns in those sensors were represented in the model. In an immediate future when the data from these sensors are to be validated with the previously trained model, the model would ideally generate an output similar to what was actually observed, provided that the data patterns in any of the sensors were not changed. In the event of a sensor generating data that was not in line with the same pattern as it had before, there would be a large difference between the simulated output data and the actual output.

In order to isolate the sensor responsible for such differences, it was necessary to construct a series of ARX models (i.e., sub-models) by omitting one of the sensors, in turn, from the input and then retraining the sub-models with the same data with which the original model was trained. All these sub-models were then validated with the same test data, which contained data from a potentially defective sensor. In that case, the sub-model from which the input data from the defective sensor was omitted performed the best, and the sub-models containing the data from the defective sensor with performed poorly.

This is a typical process of elimination of unknown variables, used in solving mathematical problems. While the ‘sequential search’ analogy was used here, the proposed method does not really seek to find a matching item, as in the regular sequential search method described earlier; rather, it seeks to isolate the item that does not match the pattern. When the analogy of a ‘binary search’ is used, rather than omitting one sensor in a sub-model, half of the sensors are omitted when the number of inputs is large. In this case, the sub-model containing the defective sensor will produce the worst result.

The assumption here was that the target sensor was not defective. If the target sensor itself was defective, the above method would fail. In that case, a different target sensor would be chosen. It also was assumed that only one of the input sensors was defective. However, if multiple input sensors were defective, the comparison of the well performing and poorly performing sub-models would reveal the composition of the defective sensors. If the target was found to be not defective, and none of the input sensors could be held responsible for the change the data pattern, it was possible that the pattern of all sensors data were changed; and in that case, it is most likely that the structure had a change in its response due to damage or altered load patterns.

The following sub-sections describe in detail the methods developed in this study.

4.5. ARX Model Development Based on the Sequential Search Method

Using the instrumented bridge column as an example, ARX model development based on the sequential search method involved a total of 17 sensors, including a temperature sensor were installed in Bridge Column 2 (sensors 17 through 32, and T). For unsupervised learning, the sensor data were collected during three consecutive months – December 2005, January 2006, and February 2006 – to develop or train the ARX model. The data from Sensor 17 were fed to the system identification tool as the target data; and other 16 sensors, including the temperature sensor, provided input data into the system identification tool. In this way, the system was trained so that it could build a mathematical relationship among the data signals from the input and target sensors.

In the next step, data from the same 16 sensors from a different period, for example, the first half of March 2006, were used for validation purposes. The target data were simulated and compared with the actual or real data from Sensor 17 for the same period. The comparison norm, r , was calculated in percentage, using Eq. 3. If the simulated and the real data for the target sensor did not match in good percentage, it would imply that there could be damage in the structure or there existed at least one defective sensor. A schematic diagram of ARX model development and testing sensor data is presented in Figure 4.

To determine if a sensor was defective or producing an erroneous signal, the above procedure was repeated; however, during each repetition, one of the input sensors was removed – say, Sensor 18 – from the input to create a separate ARX model (i.e., sub-model) with 15 input signals (Figure 5). The sub-model was then trained and tested in the same way as the original model. After that, another sub-model was created by replacing the previously removed sensor (i.e., Sensor 18 in this example) with another sensor (say, Sensor 19) in the training data, and the training and testing was performed as before. Thus, 16 sub-models were created by removing the input sensors one at a time. The simulated data produced by each model were then compared to the actual data or real data, using Eq. 3. In each particular case, the comparison of the simulated and original data for the target sensor indicated whether or not the sensor that was absent in the input data was the one responsible for the poor match in the original model. If the comparison was good only when a particular sensor was omitted from the input data, then that sensor was considered defective. If none of the cases produced a good match, this implied that damage was present in the structure. The target sensor could be changed, and the above process could be repeated for a more reliable prediction of a defective sensor or damage in the structure.

4.6. ARX Model Development based on the Binary Search Method

In order to accelerate the process of identifying a defective sensor, as outlined in the previous section, the 16 input sensors, including the temperature sensor, were divided into two groups of 8 sensors (Figure 6); each group was used to create a sub-model by keeping the target the same as Sensor 17. The first sub-model contained Sensors 18 through 24 as input, and the second sub-model contained Sensors 25 through 32 as well as T as input.

Data from December 2005, January 2006, and February 2006 were used for training the two sub-models. The sub-model then was validated with the corresponding sensors' data for a different time frame, the first half of March 2006, for example. The simulated and the measured data from Sensor 17 were compared in order to find out which sub-model produced a better match. For example, if the second sub-model produced a better match than the first one, the defective sensor would be assumed to be in the first group of sensors. At that point, the first group would be further divided into two groups of four sensors (Figure 7), and the above procedure would be repeated to find the sub-group containing the defective sensor.

When a subgroup size is small (e.g., less than four), a sequential method could be applied to the subgroup to identify the particular sensor responsible for altering the data pattern.

5. Performance of the Proposed Methods

5.1 Test 1: Sensitivity Analysis - Identification of a Known Defective Sensor

In Test 1, Sensor 27 was made artificially defective by replacing the original data with random numbers. The approach presented in Section 4.5 was followed to find the defective sensor. When any other sensors – for instance, Sensor 19 – were removed from the input in an ARX sub-model, the measured and simulated target data matched -267%, as shown in Figure 8(a). All the 16 sub-models were tested the same way and the comparison metric r was calculated, as shown in Table 1. It is noted from Table 1 that when Sensor 27 was removed from the input, the simulated output had the best match with the measured output ($r = 45\%$), indicating that the data from Sensor 27 had some problem, as shown in Figure 8 (b). While, $r = 45\%$ may not indicate a high degree of closeness between the simulated and measured data in the absolute term, for the given data sets it represents the best positive match considering the the model with the complete set of sensors as input and the reduced models with all the cases of sensor removal. From Table 1, it is clear that when Sensor 27 was included in a model, it reduced the value of r to a level lower than 45%. Thus, Sensor 27 was interpreted as

being defective. A number of similar tests were conducted by making different sensors defective and choosing different target sensors. The current method had identified the defective sensor correctly in each case.

5.2 Test 2: Identifying a Defective Sensor using the Sequential Method

In Test 2, the sequential search method was employed, as discussed in Section 4.5. In this case, data from the second half of March 2006 were used as a test data set in which the presence of any defective sensor was not known beforehand. However, the test showed that the simulated and the real target data did not match well, indicating the presence of one or more defective sensors or else structural damage.

As in the previous case, 16 sub-models were constructed by removing one input sensor at a time and training the model. The simulated target data produced by the sub-models with the test data sets and corresponding real target data were compared to see which sub-model produced the best match. Figure 9(a) shows that when Sensor 18 was removed from the input, the real or measured output match was 10.54%. Figure 9(b) shows that when Sensor 24 was removed from the input, the measured and simulated output did not match well (-109.7%). In this way, all 16 sub-models were tested, and the best match was derived. Table 2 showed that when Sensor 24 was removed, the measured and simulated outputs indicated a maximum positive match (15.28%). Therefore, Sensor 24 was considered to be the defective sensor.

5.3 Test 3: Identifying Defective Sensor using the Binary Method

Sequential search can render the process tedious if the number of sensors is high. Taking this issue into consideration, a binary search was used in Test 3. In this case, the same test data set as used in Test 2 was utilised. Initially, the 16 input sensors were divided into 2 groups of 8 sensors each and two sub-models were constructed using the same target sensor (Sensor 17).

The measured and simulated target data of the second group of 8 sensors matched well (Figures 10b) as compared to the first group of 8 sensors (Figure 10a). Therefore, the defective sensor was assumed to be contained in the first group of 8 sensors, as shown in Table 3(a). By further dividing the data from the first group into two subgroups of 4 sensors, and repeating the same procedure, the measured and simulated model output of the first group of 4 sensors were found not to match well as compared to the second group of 4 sensors. Therefore, the defective sensor was assumed to be contained in the first 4 sensors (Sensors 18 through 21) of the first group of 8 sensors, as shown in Table 3(b).

Finally, a sequential search was performed on these 4 sensors to determine that Sensor 24 was defective. By comparing Tests 2 and 3, and by evaluating a few other similar tests conducted in the study, it was observed that both the full sequential method and the binary method produced the same results. However, the binary search method required less effort.

6. Conclusions

In this study, a pattern recognition approach was applied to analyze the strain and temperature data of an instrumented bridge column. The goal was to evaluate the feasibility of the approach developed in this study in order to determine the presence of a defective sensor using the concepts of sequential search and/or binary search techniques. Numerous tests were performed to identify the defective sensors. The following observations were made:

- Removing a particular sensor at a time improved the pattern matches of the established relationship among the remaining sensors; the excluded sensors were identified as the ones responsible for the change in the relationship.
- Both sequential and binary methods as proposed here yielded identical results in the cases considered here. In identifying a defective sensor the binary method is recommended as it is more efficient.

While the current study showed success of the technique on the structure under study, using data collected for a certain period of time, further research should be performed to verify its general applicability under various conditions, such as seasonal changes in temperature or traffic loads. Also, further study is required to verify if the following interpretation of the change in data patterns were valid:

- Once the responsible sensor was identified, physical tests could be conducted to verify whether or not the sensor was actually defective. In case the sensor was not found to be defective, it was likely that the physical property of the structure had changed. However, appropriate inspection of the structure is required to ascertain that.
- If a group of sensors was found to be defective, it was likely that damage existed in the structure.
- In the absence of unusual patterns, in order to determine the rate of gradual change in the data pattern, which would indicate the rate of deterioration in the structure, the relationships simply were updated with new data and compared with the initial pattern of relationships.

Acknowledgements

The work presented here formed a part of the first author's Master's thesis at Concordia University, Montreal, under the supervision of the second author. The authors are grateful for Prof. Aftab A. Mufti, President of ISIS Canada Research Network, University of Manitoba, Winnipeg, for providing encouragement and the monitoring data for the Portage Creek Bridge. The support of the Natural Sciences and Engineering Research Council of Canada is also gratefully acknowledged.

References

- Chase S.B. and Washer G. (1997) Non-Destructive Evaluation for Bridge Management in the Next Century. *Public Roads*, V.61, No. 1, pp. 16-25.
- Dubin E.E. and Yanev B.S. (2001) Managing the East River Bridge in New York City, *Proceedings of SPIE Health Monitoring and Management of Civil Infrastructure Systems* Newport Beach, CA, pp. 60-74.
- Da Silva S., Junior M.D. and Junior V.L. (2007) Damage Detection in a Benchmark Structure using AR-ARX Models and Statistical Pattern Recognition. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, V. 29, No. 2, pp. 174-184.
- Erdoğan H. and Güral E. (2009) Identification of Dynamic Systems using Multiple Input–Single Output (MISO) models. *Nonlinear Analysis: Real World Applications*, V. 10, No. 2, pp. 1183-1196
- Fugate M., Sohn H. and Farrar C.R. (2001) Vibration-Based Damage Detection Using Statistical Process Control. *Mechanical Systems and Signal Processing*, V.15, No. 4, pp. 707-721.
- Huffman S., Bagchi A., Mufti A., Neale K., Sargent D., and Rivera E. (2006) GFRP Seismic Strengthening and Structural Health Monitoring of Portage Creek Bridge Concrete Columns. *The Arabian Journal for Science and Engineering*, V. 31, No. 1C, pp. 25-42.
- Islam M.S. (2009) Application of Statistical Pattern Recognition Techniques in Structural Health Monitoring. Master of Applied Science in Civil Engineering Thesis, Concordia University. Montreal, Canada.
- Knuth D. (1997) *The Art of Computer Programming, Volume 3: Sorting and Searching*, Third Edition. Addison-Wesley. ISBN 0-201-89685-0. Section 6.2.1: Searching an Ordered Table, pp. 409-426.
- Lu, P. (2008) A Statistical Based Damage Detection Approach for Highway Bridge Structural Health Monitoring. Ph.D. Dissertation, Iowa State University.
- Mirza S.M. and Haider M. (2003) *The State of Infrastructure in Canada: Implications for Infrastructure Planning and Policy*. Infrastructure Canada, Ottawa, Canada.
- Mufti A.A. (2001) *Guidelines for Structural Health Monitoring*. ISIS Canada design Manual No.2., ISIS Canada Research Network, University of Manitoba, Winnipeg, Canada.
- Phares B. (2001) Highlights of Study of Reliability of Visual Inspection. Presentation at the Annual Meeting of TRB Subcommittee A2C05 (1) Non-destructive Evaluation of Structures, FHWA Report Nos. FHWA-RD-01-020 and FHWA-RD-01-021.
- RITA (2010) Research and Innovative Technology Administration, U.S. Department of Transportation (US DOT), 1200 New Jersey Avenue, SE, Washington, DC, http://www.bts.gov/publications/national_transportation_statistics/html/table_01_27.html (accessed, October, 2012).
- Rytter A (1993) *Vibration Based Inspection of Civil Engineering Structure*. Ph.D. Dissertation, Department of Building Technology and Structural Engineering, University of Aalborg, Denmark.

Sohn H., Czarnecki J.J. and Farrar C.R. (2000) Structural Health Monitoring Using Statistical Process Control. ASCE Journal of Structural Engineering, V. 126, No. 11, pp. 1356-1363.

Sohn H., Farrar C.R., Hunter N.F. and Worden K. (2001) Structural Health Monitoring Using Statistical Pattern Recognition Techniques. ASME Journal of Dynamic Systems, Measurement, and Control, V.123, No. 4, pp. 706-711.

Yao R. and Pakzad S.N. (2012) Autoregressive Statistical Pattern Recognition Algorithms for Damage Detection in Civil Structures. Mechanical Systems and Signal Processing, V. 31, pp. 355-368

Yang M. and Makis V. (2010) ARX Model-Based Gearbox Fault Detection and Localization under Varying Load Conditions. Journal of Sound and Vibration, V. 329, No. 24, pp. 5209-5221.

Zhang Q.W. (2007) Statistical Damage Identification for Bridges Using Ambient Vibration Data. Computers and Structures, V. 85, No. 7-8, pp. 476-485

Table 1 Percentage match of measured and simulated output for each sensor removal.

Sensor Removed	Match (%)
18	31.6
19	-267
20	36.9
21	40.5
22	35.5
23	-26.5
24	24.4
25	21.7
26	36
27	45
28	-32
29	38
30	43
31	-59
32	-36
T	35

Table 2 Percentage match of measured and simulated output for each sensor removal.

Sensor removed	Match (%)
18	10.54
19	5.94
20	-8.32
21	7.98
22	7.95
23	8.53
24	15.28
25	11.2
26	4.664
27	-22.95
28	-109.7
29	8.22
30	10.1
31	7.89
32	0.34
T	10.48

Table 3 Degree of matching of the simulated output with the measured response.

Sensors	Match (%)	Comments
1 st 8	-94.34	Defective sensor group
2 nd 8	7.01	Non-defective sensor group

(a)

Sensors	Match (%)	Comments
1 st 4 of defective group	-10.49	Defective sensor group
2 nd 4 of defective group	3.03	Non-defective sensor group

(b)

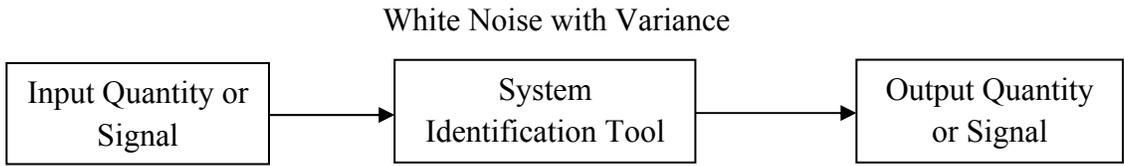


Figure 1: Basic Input-output configuration of the ARX model.

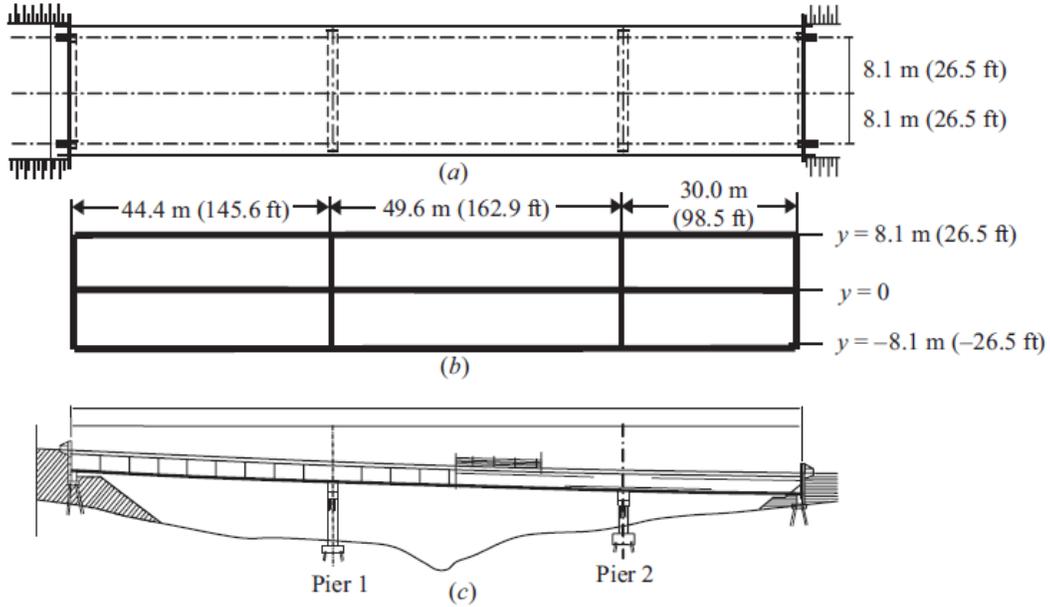


Figure 2: Plan and elevation of the Portage Creek Bridge [12].

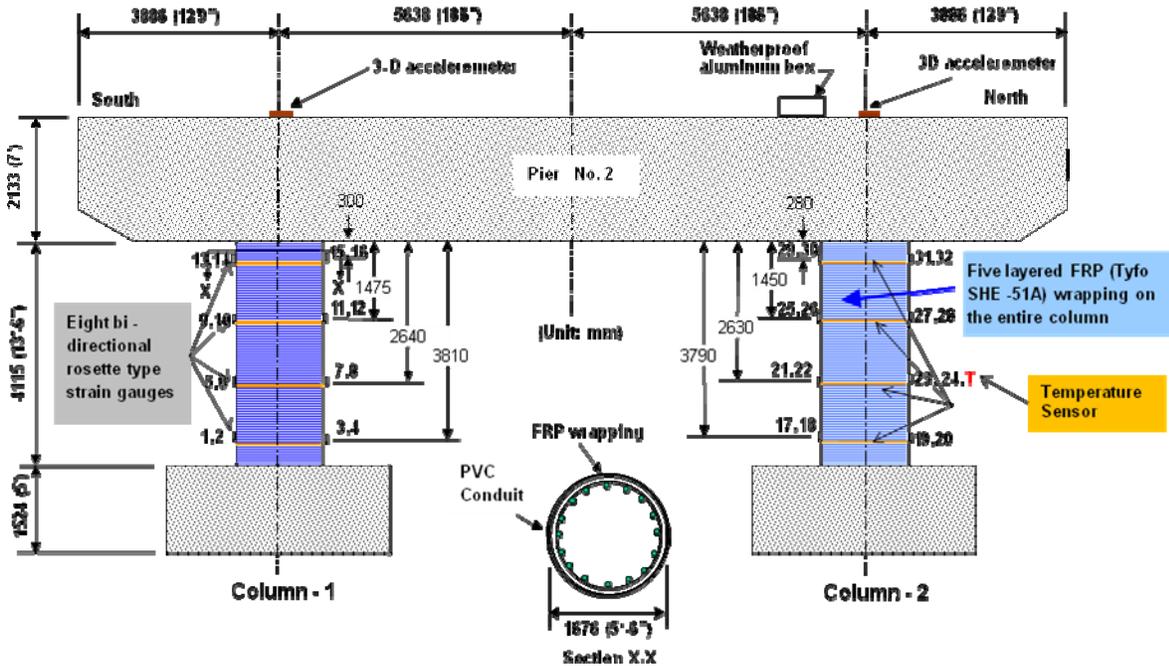


Figure 3: Sensor locations on columns of Pier 2 of the Portage Creek Bridge [12].

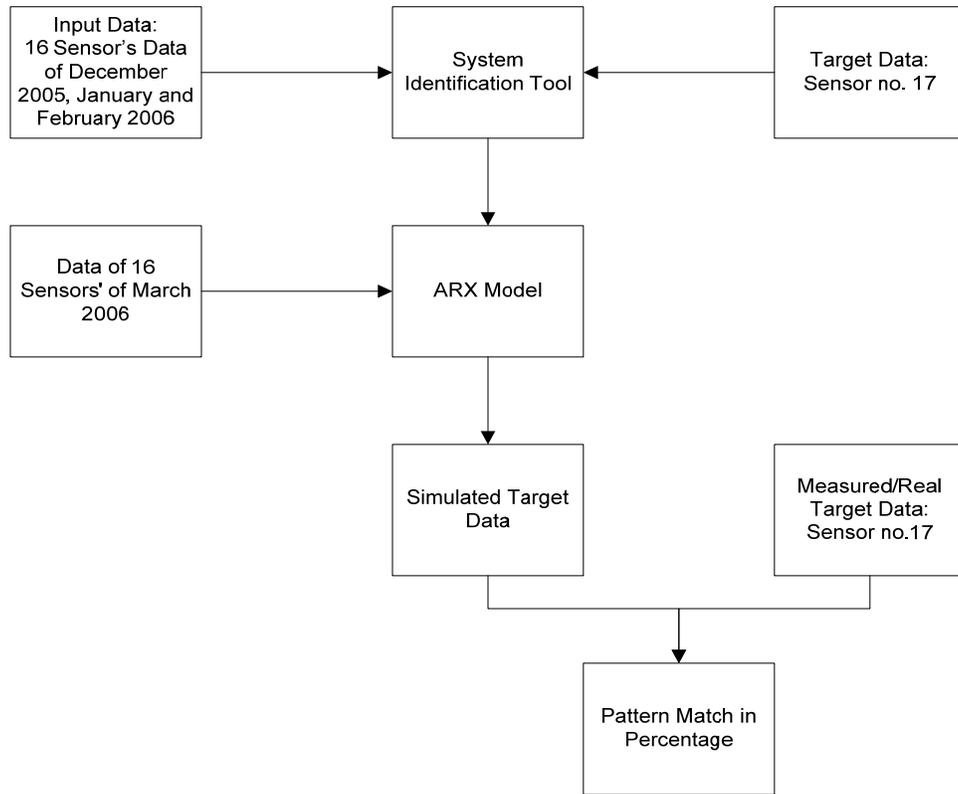


Figure 4: Schematic diagram for building the ARX model with Sensor 17 as the target.

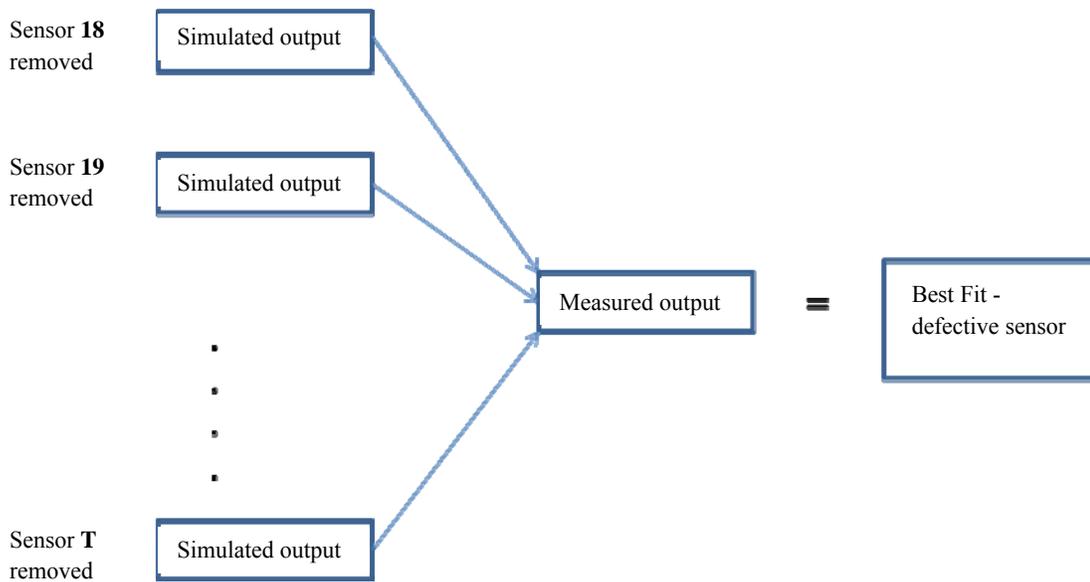


Figure 5: Schematic diagrams for comparing all simulated output with the measured output.

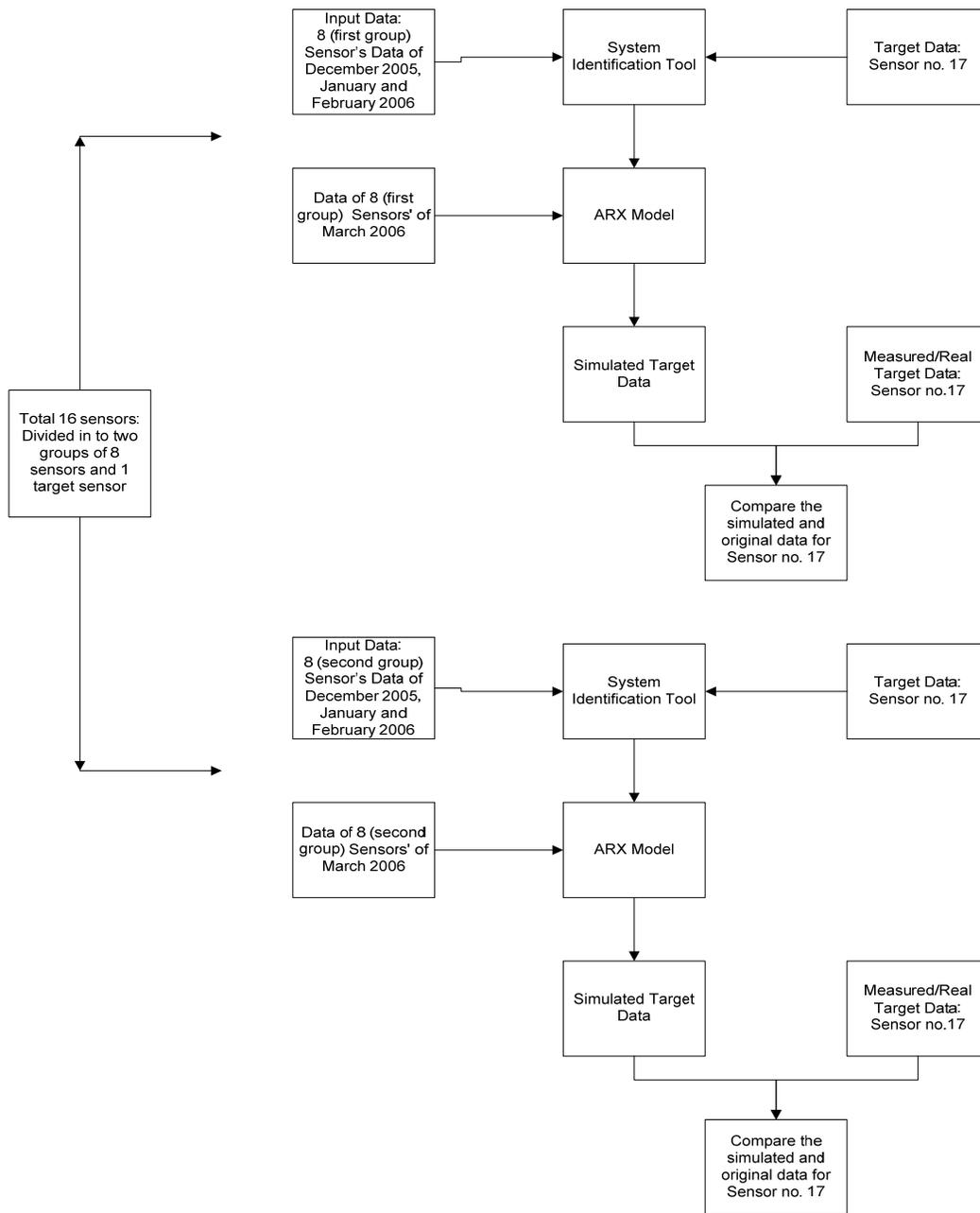


Figure 6: Schematic diagram for identifying a defective sensor by the ARX model with a binary search method (first stage).

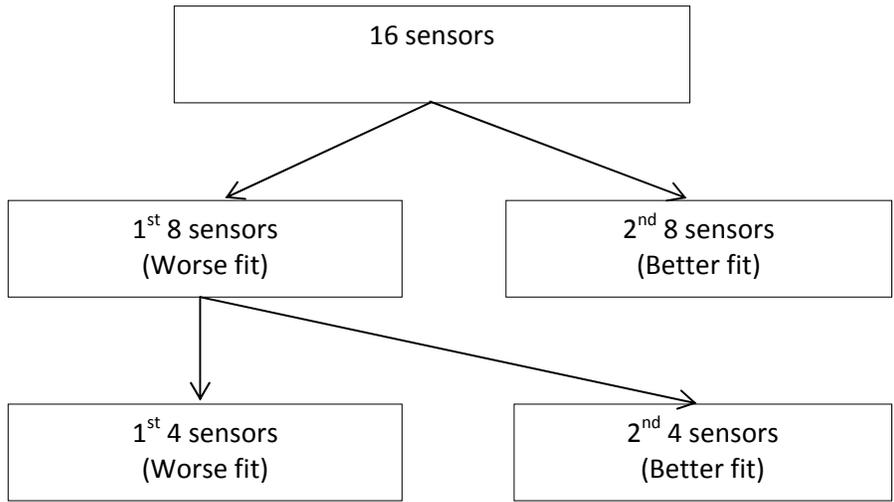


Figure 7: Schematic diagram for identifying defective sensor by the ARX model with a binary search method.

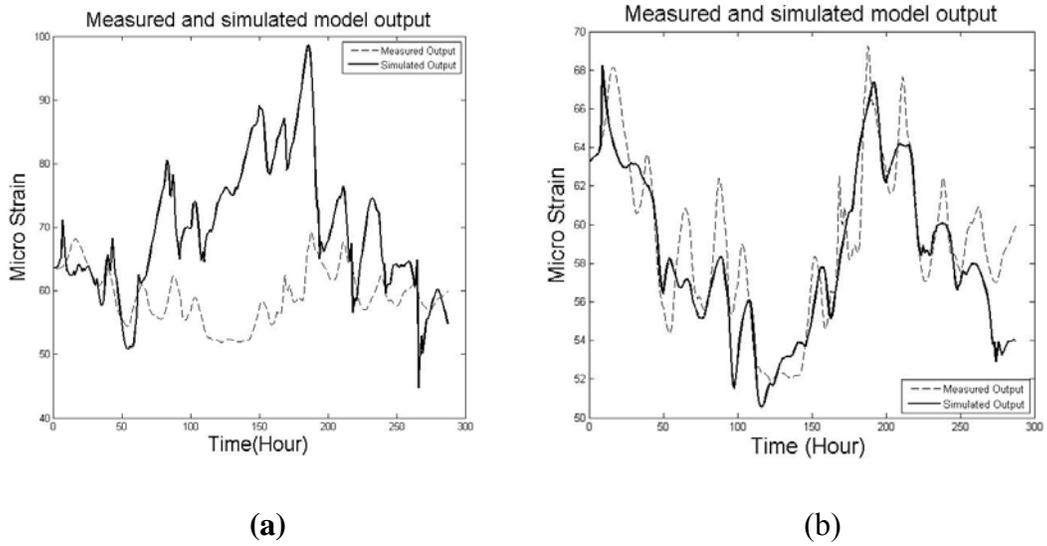
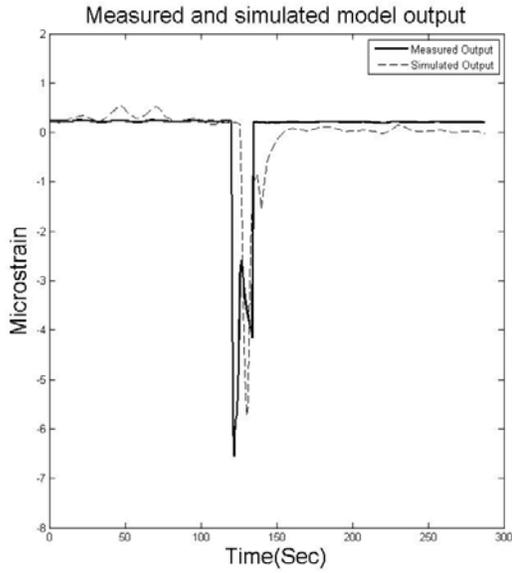
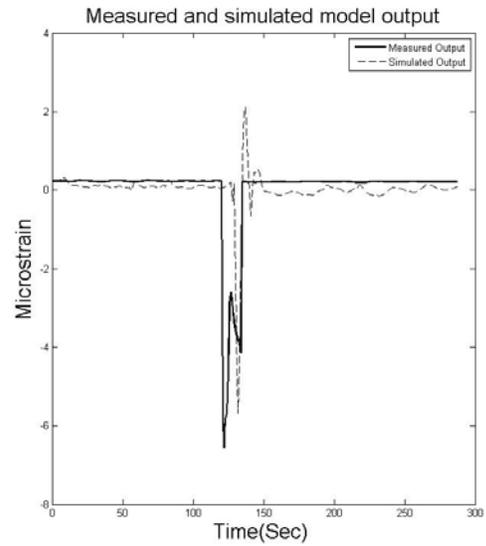


Figure 8: Measured and simulated output from March 2006 with (a) Sensor 19 removed and (b) Sensor 27 removed.

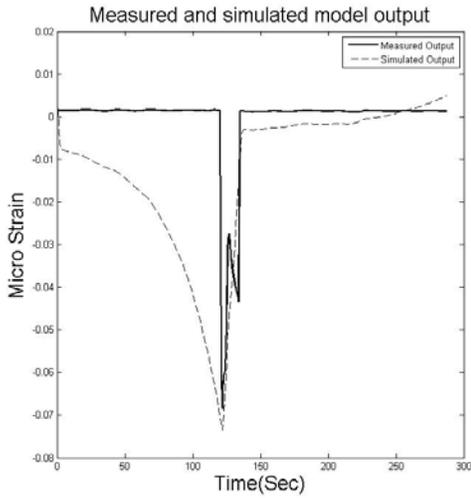


(a)

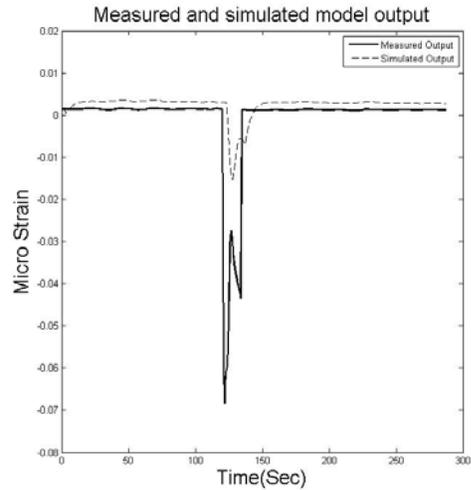


(b)

Figure 9: Measured and simulated output from March 2006 with (a) Sensor 18 removed and (b) Sensor 24 removed.



(a)



(b)

Figure 10: Measured and simulated output from March 2006 for (a) the first group of 8 sensors and (b) the second group of 8 sensors.