COBEM 2023 27TH INTERNATIONAL CONGRESS OF MECHANICAL ENGENEERING

ABCM Brazilian Society of Mechanical Sciences and Engineering
27th ABCM International Congress of Mechanical Engineering
December 04-08, 2023, Florianópolis-SC, Brazil

COB-2023-1473
DETECTING AND LOCALIZING DAMAGE IN AN ACTIVE AUTOMOTIVE SUSPENSION USING MACHINE LEARNING METHODS AND A FILTER BANK

Bruno da Silva Santos
Murilo Bellezoni Loiola
Helói Francisco Gentil Genari
Engineering, Modeling, and Applied Social Sciences Center, Federal University of ABC, Santo André, São Paulo 09210-580, Brazil
s.bruno@aluno.ufabc.edu.br, murilo.loiola@ufabc.edu.br, and heloi.genari@ufabc.edu.br

Fabíola Martins Campos de Oliveira
Center of Mathematics, Computing, and Cognition, Federal University of ABC, Santo André, São Paulo 09210-580, Brazil
fabiola.oliveira@ufabc.edu.br

Abstract. *The structural health monitoring area has received increasing attention from the industry and the scientific community in recent decades. Several techniques applied to damage detection and localization have been proposed, including recent developments based on artificial intelligence. In this context, this paper investigates a framework that combines a machine learning method with a residue generator to detect and localize multiple damaged regions. The residue signal is the output of a filter bank, built with Luenberger observers. This type of estimation allows a direct relation between the residue signals and the changes in dynamics caused by damage. However, the correlation between the residue signal and the damage localization is not straight. For this purpose, this paper analyzes the performance of three classifiers: k-nearest neighbors, a decision tree, and a support vector machine to identify damage occurrence and its respective localization using the temporal residue data. An active automotive suspension model is used as a case study structure, generating the temporal data to build the classifiers. Simulated results show that the support vector machine method is the best tested classifier to identify changes in the suspension components, achieving a classification accuracy between 79.6% and 100%.*

Keywords: *structural health monitoring, artificial intelligence, filter bank, active suspension.*

1. INTRODUCTION

Repetitive dynamic forces, robust events, or both may lead to damage occurrence in mechanical structures, compromising operational safety. Consequently, the structural health monitoring (SHM) area has received increasing attention from the industry and the scientific community in the last decades (Genari and Nóbrega, 2012; Meher and Sunny, 2022), aiming to satisfy safety requirements, prevent catastrophic failures, reduce maintenance costs, and extend the structure lifetime (Genari and Oliveira, 2019). Several techniques applied to damage detection and localization have been proposed, including recent developments based on artificial intelligence (Sarmadi and Entezami, 2021; Das *et al.*, 2021; Wang and Cha, 2022).

Damage alters the structural dynamics, changing the structure vibration characteristics such as natural frequencies, vibration modes, and damping (Genari *et al.*, 2013). Vibration-based SHM methods capture this dynamic change by comparing the structure behavior before and after damage occurrence. These techniques are attractive because they are non-destructive and global, *i.e.*, the damage existence may be detected remotely (Kullaa, 2010). Several vibration-based strategies have been proposed in the last few years. For instance, Pavelko *et al.* (2017) applied a correlation coefficient deviation to extract the damage features. Li *et al.* (2020) proposed an algorithm based on the unscented Kalman filter for damage detection and localization. Mousavi *et al.* (2021) presented an SHM approach based on the wavelet transform and an artificial neural network.

Machine learning techniques can process many vibration signals. Despite the promising results, the machine learning strategy is yet an under-explored theme for SHM purposes. In this context, this paper investigates a vibration-based SHM framework to detect and localize damage in an active automotive suspension. This framework combines a filter bank and machine learning algorithms to provide suspension system diagnosis. The filter bank is built with Luenberger observers and is used to extract damage features using a residue generator. The temporal residue signals are the input of the machine-learning methods: k-nearest neighbors (k-NN), a support vector machine (SVM), and a decision tree. These classifiers have the performance evaluated concerning damage diagnosis, including the identification and localization of multiple damaged components. An active quarter-car model with the tire, wheel, suspension, and quarter-car mass is used to create the temporal data, in which the characteristic component alterations represent the damage occurrence.

The remainder of this paper is organized as follows: in Section 2, the state-space model of an automotive suspension is described. The proposed method for damage classification is presented in Section 3, while the simulation results are presented and analyzed in Section 4. Finally, Section 5 concludes the paper.

2. STATE-SPACE MODEL OF AN AUTOMOTIVE SUSPENSION

Figure 1 presents a quarter-car model, considering the vertical dynamics and the vehicle symmetry. The sprung mass m_s represents a quarter of the vehicle body weight. The spring k_s, damper c_s, and the actuator, which produces the control force $f_c(t)$, compose the suspension system. The wheel is described as the unsprung mass m_u, connecting the suspension and the tire, which is modeled by the spring with stiffness k_t associated with a damping coefficient c_t. As the car moves along the road, the displacement $z_r(t)$ at the tire excites the system.

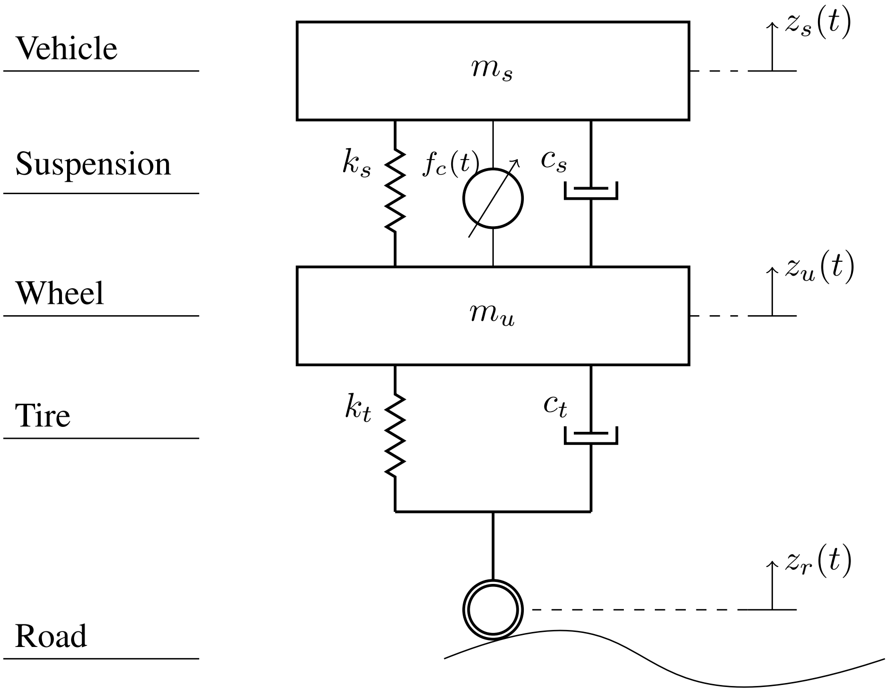


Figure 1: A quarter-car model.

Applying Newton's second law to the sprung and unsprung masses, the following motion equations result:

$$\begin{aligned} m_s\ddot{z}_s(t) &= -k_s\left[z_s(t)-z_u(t)\right]-c_s\left[\dot{z}_s(t)-\dot{z}_u(t)\right]+f_c(t) \\ &= -k_s z_s(t)+k_s z_u(t)-c_s\dot{z}_s(t)+c_s\dot{z}_u(t)+f_c(t) \\ \ddot{z}_s(t) &= -\frac{k_s}{m_s}z_s(t)+\frac{k_s}{m_s}z_u(t)-\frac{c_s}{m_s}\dot{z}_s(t)+\frac{c_s}{m_s}\dot{z}_u(t)+\frac{1}{m_s}f_c(t) \qquad (1) \\ m_u\ddot{z}_u(t) &= k_s\left[z_s(t)-z_u(t)\right]-k_t\left[z_u(t)-z_r(t)\right]+c_s\left[\dot{z}_s(t)-\dot{z}_u(t)\right]-c_t\left[\dot{z}_u(t)-\dot{z}_r(t)\right]-f_c(t) \\ &= k_s z_s(t)-\left(k_s+k_t\right)z_u(t)+c_s\dot{z}_s(t)-\left(c_s+c_t\right)\dot{z}_u(t)+k_t z_r(t)+c_t\dot{z}_r(t)-f_c(t) \\ \ddot{z}_u(t) &= \frac{k_s}{m_u}z_s(t)-\frac{(k_s+k_t)}{m_u}z_u(t)+\frac{c_s}{m_u}\dot{z}_s(t)-\frac{(c_s+c_t)}{m_u}\dot{z}_u(t)+\frac{k_t}{m_u}z_r(t)+\frac{c_t}{m_u}\dot{z}_r(t)-\frac{1}{m_u}f_c(t). \qquad (2) \end{aligned}$$

Defining the disturbance input as the road profile and the control input as the actuator force, *i.e.*,

$$\begin{aligned} w(t) &= z_r(t) \\ u(t) &= f_c(t), \end{aligned}$$

and the state vector $\mathbf{x}(t)$ as a composition between the position and velocity variables, *i.e.*,

$$\begin{aligned} \mathbf{x}(t) &= \begin{bmatrix} z_s(t) & z_u(t) & \dot{z}_s(t) & \dot{z}_u(t)-\frac{c_t}{m_u}z_r(t) \end{bmatrix}^T \\ &= \begin{bmatrix} x_1(t) & x_2(t) & x_3(t) & x_4(t) \end{bmatrix}^T, \end{aligned}$$

Eq. (1) and Eq. (2) can be combined into a single equation in the following state-space representation:

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}_1 w(t) + \mathbf{B}_2 u(t)\\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t) + \mathbf{D}_1 w(t) + \mathbf{D}_2 u(t),\end{aligned} \tag{3}$$

in which $\mathbf{A}$ is the state matrix, $\mathbf{B}_1$ and $\mathbf{B}_2$ are the input matrices, and the output matrices are $\mathbf{C}$, $\mathbf{D}_1$ and $\mathbf{D}_2$.

The first system output signal is the suspension deflection. Considering the vehicle body has an accelerometer, the second measured output is the sprung mass acceleration. These two output signals are usually used to design, test an automotive suspension, or both (Sun *et al.*, 2012). Thus, the output vector $\mathbf{y}(t)$ is written as

$$\begin{aligned}\mathbf{y}(t) &= \begin{bmatrix} y_1(t) & y_2(t)\end{bmatrix}^T\\ &= \begin{bmatrix} z_s(t) - z_u(t) & \ddot{z}_s(t)\end{bmatrix}^T,\end{aligned}$$

leading to the following state-space matrices of the model in Eq. (3):

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\\ -\frac{k_s}{m_s} & \frac{k_s}{m_s} & -\frac{c_s}{m_s} & \frac{c_s}{m_s}\\ \frac{k_s}{m_u} & -\frac{k_s+k_t}{m_u} & \frac{c_s}{m_u} & -\frac{c_s+c_t}{m_u}\end{bmatrix},\ \mathbf{B}_1 = \begin{bmatrix} 0\\ \frac{c_t}{m_u}\\ \frac{c_s c_t}{m_s m_u}\\ \frac{k_t}{m_u} + \frac{c_t(c_s+c_t)}{m_u^2}\end{bmatrix},\ \mathbf{B}_2 = \begin{bmatrix} 0\\ 0\\ \frac{1}{m_s}\\ -\frac{1}{m_u}\end{bmatrix},$$

$$\mathbf{C} = \begin{bmatrix} 1 & -1 & 0 & 0\\ -\frac{k_s}{m_s} & \frac{k_s}{m_s} & -\frac{c_s}{m_s} & \frac{c_s}{m_s}\end{bmatrix},\ \mathbf{D}_1 = \begin{bmatrix} 0\\ \frac{c_s c_t}{m_s m_u}\end{bmatrix},\text{ and } \mathbf{D}_2 = \begin{bmatrix} 0\\ \frac{1}{m_s}\end{bmatrix}.$$

3. OVERVIEW OF THE PROPOSED APPROACH FOR DAMAGE CLASSIFICATION

Figure 2 shows the detailed block diagram of the proposed framework, containing the suspension system and an SHM module, which encompasses a residue generator and a classifier system. The suspension output signals are defined as the relative displacement between the wheel and vehicle masses ($y_1(t)$) and the vehicle acceleration ($y_2(t)$). The residue generator module uses a filter bank composed of two observers to estimate the suspension output signals, designed with the healthy model and considering that the input $w(t)$ is unknown. The residue generator module compares the outputs of the operational and estimated structure, generating the residue signals $r_1(t)$ and $r_2(t)$. The classifier system processes these residue signals to detect and localize damage using machine learning strategies. Thus, when the suspension is damaged, this alters the structure dynamics, leading to a residue signal increase, which is classified to provide the diagnosis of the suspension system. Next, each component of the block diagram shown in Fig. 2 is described.

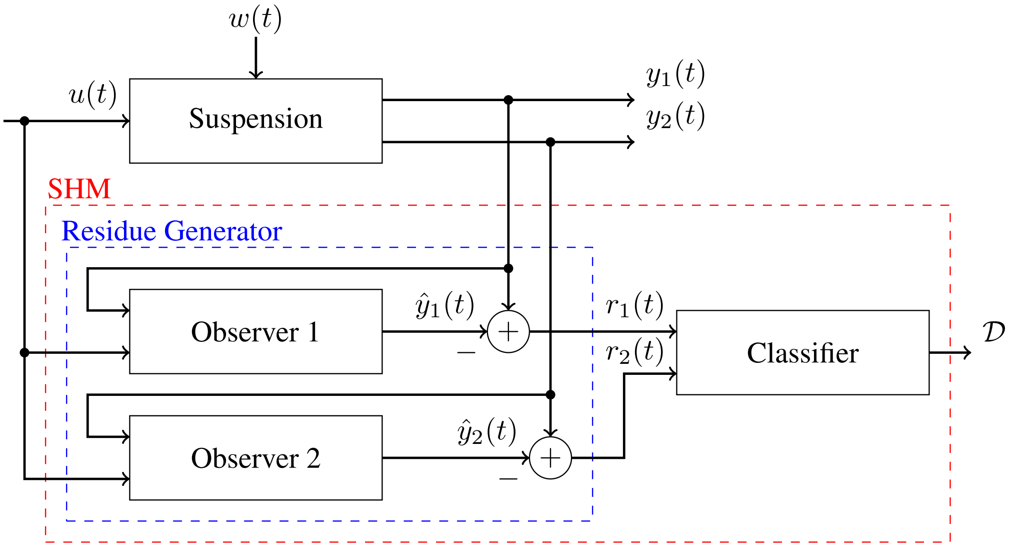


Figure 2: Block diagram of the SHM framework.

3.1 Residue generator

The dynamics of the Luenberger observer is defined as (Franklin *et al.*, 2010):

$$\begin{aligned}\dot{\hat{\mathbf{x}}}(t) &= \mathbf{A}\hat{\mathbf{x}}(t) + \mathbf{B}_2 u(t) + \mathbf{L}\left[\mathbf{y}(t) - \hat{\mathbf{y}}(t)\right]\\ \hat{\mathbf{y}}(t) &= \mathbf{C}\hat{\mathbf{x}}(t) + \mathbf{D}_2 u(t),\end{aligned}$$

in which $\hat{\mathbf{x}}(t)$ is the estimated state vector, $\hat{\mathbf{y}}(t)$ represents the estimated output vector, and $\mathbf{L}$ is the observer gain matrix.

The state estimation error is defined as $\mathbf{e}(t) = \mathbf{x}(t) - \hat{\mathbf{x}}(t)$. Thus, considering the active suspension model represented by Eq. (3), the estimation error dynamics is given by:

$$\begin{aligned}\dot{\mathbf{e}}(t) &= \dot{\mathbf{x}}(t) - \dot{\hat{\mathbf{x}}}(t)\\ &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}_1 w(t) + \mathbf{B}_2 u(t) - \mathbf{A}\hat{\mathbf{x}}(t) - \mathbf{B}_2 u(t) - \mathbf{L}\left[\mathbf{y}(t) - \hat{\mathbf{y}}(t)\right]\\ &= \mathbf{A}\left[\mathbf{x}(t) - \hat{\mathbf{x}}(t)\right] + \mathbf{B}_1 w(t) - \mathbf{L}\left[\mathbf{C}\mathbf{x}(t) + \mathbf{D}_1 w(t) + \mathbf{D}_2 u(t) - \mathbf{C}\hat{\mathbf{x}}(t) - \mathbf{D}_2 u(t)\right]\\ &= (\mathbf{A} - \mathbf{L}\mathbf{C})\,\mathbf{e}(t) + (\mathbf{B}_1 - \mathbf{L}\mathbf{D}_1)\,w(t), \end{aligned} \tag{4}$$

where the matrix $\mathbf{L}$ must be chosen to minimize the disturbance effects in the state vector estimation (Franklin *et al.*, 2010). Equation (4) shows that the dynamic behavior of the error vector is determined by the eigenvalues of matrix $(\mathbf{A} - \mathbf{L}\mathbf{C})$. Furthermore, if the pair $(\mathbf{A}, \mathbf{C})$ is observable, matrix $\mathbf{L}$ can be computed by a pole placement routine.

Under the presence of damage, the active suspension model can be rewritten as

$$\begin{aligned}\dot{\mathbf{x}}(t) &= (\mathbf{A} + \Delta\mathbf{A})\,\mathbf{x}(t) + (\mathbf{B}_1 + \Delta\mathbf{B}_1)\,w(t) + (\mathbf{B}_2 + \Delta\mathbf{B}_2)\,u(t)\\ \mathbf{y}(t) &= (\mathbf{C} + \Delta\mathbf{C})\,\mathbf{x}(t) + (\mathbf{D}_1 + \Delta\mathbf{D}_1)\,w(t) + (\mathbf{D}_2 + \Delta\mathbf{D}_2)\,u(t),\end{aligned}$$

in which $\Delta\mathbf{A}$, $\Delta\mathbf{B}_1$, $\Delta\mathbf{B}_2$, $\Delta\mathbf{C}$, $\Delta\mathbf{D}_1$, and $\Delta\mathbf{D}_2$ represent the perturbations in the model caused by damage. The designed filter bank minimizes the residue signals in the healthy condition. However, when the suspension suffers damage, its dynamics change, increasing the amplitude of the residue signals. Thus, the classifiers can interpret these changes, detecting and localizing damage.

3.2 Damage classification

Three classification algorithms are analyzed to evaluate the SHM framework performance: K-nearest neighbors (k-NN), Support Vector Machine (SVM), and decision tree. Consider that a set of correctly classified data points Ω is given. This set contains n pairs $(\mathbf{x}_1, \theta_1)$, $(\mathbf{x}_2, \theta_2)$, ..., $(\mathbf{x}_n, \theta_n)$, in which $\mathbf{x}_i$ represents the measurement and θ_i is the class previously attributed to the i-th measurement. The classification problem is stated as follows: given a new pair $(\mathbf{x}, \theta)$, where only the measurement $\mathbf{x}$ is known, classify θ using the information contained in Ω (Brunton and Kutz, 2022). In this paper, the damage classification problem is a binary classification problem with multiple outputs, *i.e.*, each component (output) can be classified as healthy (h) or damaged (d).

3.2.1 K-nearest neighbors

The K-NN is a non-parametric supervised learning algorithm. This method classifies a new object based on its similarities to a previously available data set. In other words, given a new data point $\mathbf{x}$ which does not have a label, find the k-nearest neighbors $\mathbf{x}_i$ with labels θ_i. Then, the label of $\mathbf{x}$ corresponds to the label of the highest occurrence. A data point $\mathbf{x}' \in \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n\}$ is defined as a nearest neighbor to $\mathbf{x}$ if (Brunton and Kutz, 2022)

$$\min d(\mathbf{x}_i, \mathbf{x}) = d(\mathbf{x}', \mathbf{x}),$$

in which $d(\mathbf{x}_i, \mathbf{x}_j)$ is a defined metric of distance. The most common choice is the Minkowski distance

$$d(\mathbf{x}_i, \mathbf{x}_j) = \left(\sum |\mathbf{x}_i - \mathbf{x}_j|^p\right)^{\frac{1}{p}}.$$

This generalized distance definition contains many well-known distances as special cases, such as the Euclidian and the Manhattan distance. In this paper, p is chosen so that the Minkowski distance corresponds to the Euclidean distance, *i.e.*, $p = 2$.

A set of nearest neighbors Ω_k can be constructed by simply finding the k-nearest neighbors $\mathbf{x}_k$ to a new data point $\mathbf{x}$. Let $\theta_k = f(\mathbf{x}_k)$ be a function that returns the class associated with the observation $\mathbf{x}_k$. Then, the k-NN classifier can be defined as follows:

$$h(\mathbf{x}) = mode(f(\mathbf{x}_1), f(\mathbf{x}_2), ..., f(\mathbf{x}_k)),$$

where $mode(\cdot)$ represents the choice of the label of the highest occurrence (Brunton and Kutz, 2022).

3.2.2 Support vector machines

The idea of the linear SVM method is to construct a hyperplane (Brunton and Kutz, 2022):

$$\mathbf{w} \cdot \mathbf{x} + b = 0, \tag{5}$$

in which the vector $\mathbf{w}$ and the constant b parameterize it. Given the hyperplane in Eq. (5), a new data point $\mathbf{x}_k$ can be classified by:

$$h(\mathbf{x}_k) = \operatorname{sign}\left(\mathbf{w} \cdot \mathbf{x}_k + b\right). \tag{6}$$

It is critical to the algorithm accuracy that $\mathbf{w}$ and b are properly determined. In that manner, an appropriate optimization must be formulated. Although easily implemented, linear classifiers are too restrictive for data embedded into a high-dimensional space. The feature space for SVM must be enriched to overcome this limitation and build more sophisticated classification curves. Thus, the data is mapped into a nonlinear and higher dimensional space as

$$\mathbf{x} \mapsto \Phi\left(\mathbf{x}\right),$$

where $\Phi(\mathbf{x})$ is a nonlinear function that projects the observation into a higher dimensional space. The labeling function is now

$$h(\mathbf{x}) = \operatorname{sign}(\mathbf{w} \cdot \Phi(\mathbf{x}) + b).$$

The SVM method of creating nonlinear classifiers by enriching data in higher dimensions may lead to a computationally intractable optimization problem, making the computation of the vectors $\mathbf{w}$ too expensive. To solve this problem, the kernel trick is introduced, and the vector $\mathbf{w}$ is represented as follows (Brunton and Kutz, 2022):

$$\mathbf{w} = \sum_i \alpha_i \Phi(\mathbf{x}_i),$$

in which α_i weights $\Phi(\mathbf{x}_i)$. Eq. (6) can be generalized to

$$h(\mathbf{x}) = \operatorname{sign}\left[\sum_i \alpha_i \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}) + b\right].$$

The kernel function is defined as

$$K(\mathbf{x}_i, \mathbf{x}) = \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}),$$

and the optimization problem is stated as

$$\underset{\boldsymbol{\alpha}, b}{\arg\min} \sum_i l(\theta_i, \bar{\theta}_i) + \frac{1}{2} || \sum_i \alpha_i \Phi(\mathbf{x}_i) ||^2 \quad \text{subject to} \quad \min_i |\mathbf{x}_i \cdot \mathbf{w}| = 1,$$

where $\boldsymbol{\alpha}$ is the vector of α_i coefficients that must be determined in the minimization process and $l(\theta_i, \bar{\theta}_i)$ represents a loss function (Brunton and Kutz, 2022).

3.2.3 Decision trees

Decision tree classifiers establish an algorithmic flowchart for making decisions using rules concerning a desired outcome. This algorithm provides a principled data-driven method for creating a predictive model for classification and regression problems. The technique looks for optimal ways to split the data to provide a robust classification network. Generally, the decision tree method produces interpretable results that can be graphically displayed and easily interpreted (Brunton and Kutz, 2022).

A decision tree method is composed of internal decision nodes and terminal leaves. Each node m implements a test function $h_m(\mathbf{x})$ with discrete outcomes labeling the branches. Given an input, at each node, a test is applied and one of the branches is chosen depending on the outcome. This process is recursively repeated until a leaf node is reached. At the leaf node, the value written by the leaf constitutes the output (Alpaydin, 2004).

4. RESULTS

This section presents the simulation analysis of the SHM methodology. Initially, a reference active suspension model is depicted, including the different damage possibilities. Then, the machine learning methods are trained to diagnose the state of each suspension component. Finally, the classification network performance is evaluated in terms of classification accuracy and confusion matrices. All simulations and designs are performed using the MATLAB$^{®}$ software, including the classification process.

4.1 Simulation setup

The main idea of the framework is to use the system actuator to excite the suspension with a known signal. The generated residue signal provides the damage signature, which is processed to determine if the suspension components are healthy or damaged. Figure 3 shows the reference signal $u(t)$ used to excite the active suspension. The actuator force is a chirp signal with a frequency band between 0 and 10 Hz, amplitude of 10 N, 10 s of period, and sampled at 100 Hz.

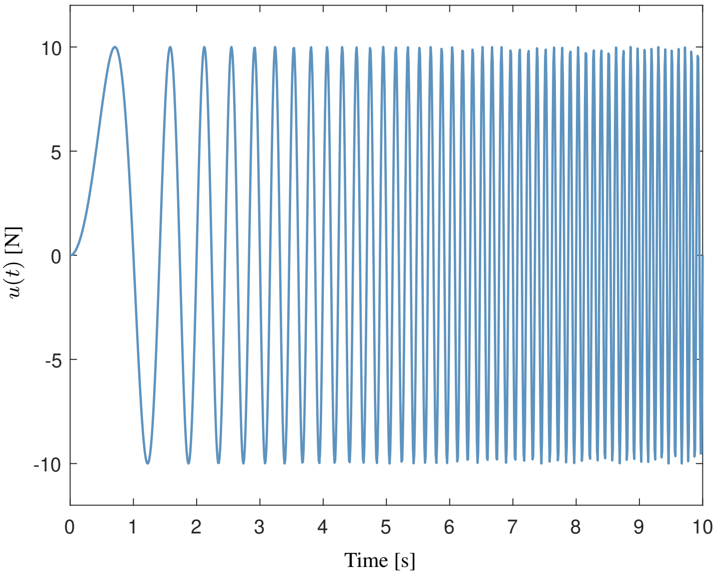


Figure 3: Excitation signal.

To simulate the active suspension structure presented in Fig. 1, the healthy system parameters are considered to be $m_s = 1.45$ kg, $m_u = 1$ kg, $k_s = 900$ N/m, $k_t = 1250$ N/m, $c_s = 7.5$ Ns/m, and $c_t = 5$ Ns/m. These parameters are based on the active suspension manufactured by Quanser (2013). Thus, the structure model is given by:

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -620.7 & 620.7 & -5.172 & 5.172 \\ 900 & -2150 & 7.5 & -12.5 \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0 \\ 5 \\ 25.86 \\ -1313 \end{bmatrix} w(t) + \begin{bmatrix} 0 \\ 0 \\ 0.6897 \\ -1 \end{bmatrix} u(t) \\ \mathbf{y}(t) &= \begin{bmatrix} 1 & -1 & 0 & 0 \\ -620.7 & 620.7 & -5.172 & 5.172 \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0 \\ 25.86 \end{bmatrix} w(t) + \begin{bmatrix} 0 \\ 0.6897 \end{bmatrix} u(t). \end{aligned} \tag{7}$$

4.2 Residue generator design

The information used to detect and localize damage is the residue signal. It is fundamental for the classifier system that the residue generator provides adequate information about the damage characteristics. For this purpose, the Luenberger observer is applied to estimate the outputs, which are compared to the measured outputs, generating the residue signals. Two observers are designed using Eq. (4), and the healthy suspension model is given by Eq. (7). Moreover, the observers are designed to be four times faster than the healthy suspension dynamics. Thus, the first observer matrix is given by

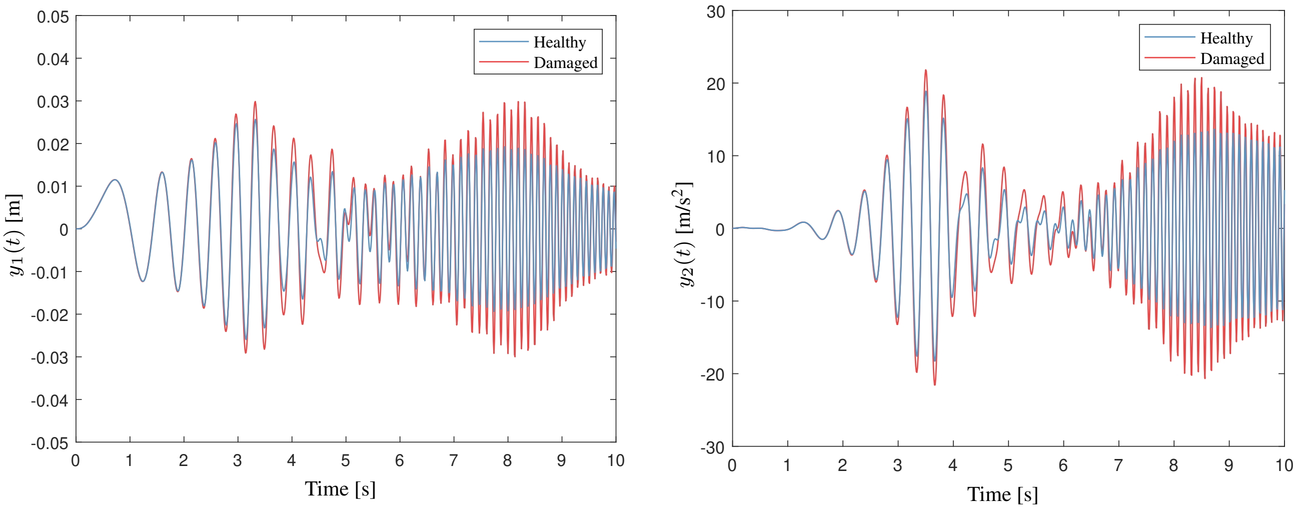
$$\mathbf{L}_1 = \begin{bmatrix} -150.8 & 203.8 & 158 \times 10^3 & 117 \times 10^3 \end{bmatrix}^T.$$

Likewise, the second observer gain matrix is computed as

$$\mathbf{L}_2 = \begin{bmatrix} -2.368 & 1.799 & 255 & -176.4 \end{bmatrix}^T.$$

4.3 Damage effects

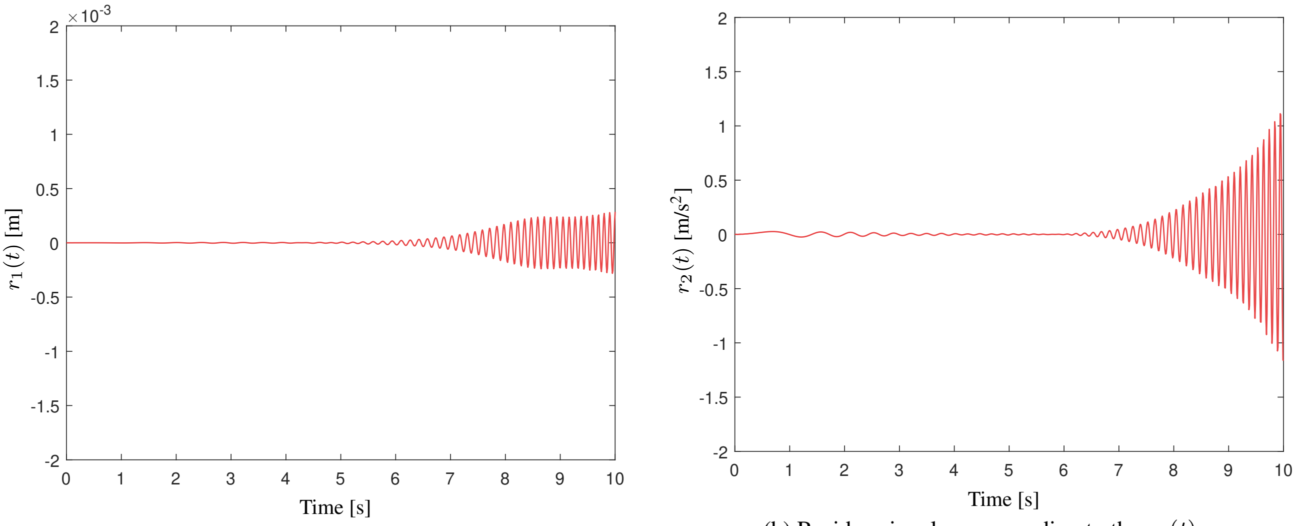
Once structural damage may significantly alter the dynamic response and, consequently, the residue signal, new models are introduced to update the suspension dynamics, *i.e.*, damage effects are created by varying the active suspension parameters. For instance, damage in a damper is common in a suspension. Thus, the damaged system can be created by reducing the damping coefficient c_s by 50%. Figure 4 compares the output signals between the healthy and the damaged suspension, while Fig. 5 shows the respective residue signals, created by comparing the measured and estimated output signals. This damage increases the suspension displacement and the sprung mass acceleration, resulting in an increase in both residue signals, $r_1(t)$ and $r_2(t)$.



(a) Suspension displacement.

(b) Sprung mass acceleration.

Figure 4: Damage effects in the response signals.



(a) Residue signal corresponding to $y_1(t)$.

(b) Residue signal corresponding to the $y_2(t)$.

Figure 5: Damage effects in the residue signals.

4.4 Damage classification

The damage classification problem can be seen as a binary classification problem with multiple outputs, *i.e.*, given $r_1(t)$ and $r_2(t)$, it is necessary to classify each suspension component state as healthy (h) or damaged (d). A supervised learning framework is implemented, consisting of an inductive reasoning process, where a set of rules are learned from training examples, and a classification algorithm that can successfully apply these rules to classify new instances is chosen (Dougherty, 2012). The metrics adopted to evaluate the performance of each classifier are prediction accuracy, precision, and recall. Accuracy is defined as the number of correct predictions divided by the total number of predictions. Precision, or positive predictive value, is calculated as the number of correct positive (damage) classifications divided by the total number of positive classifications, either right or wrong, that is, the number of true positives divided by the number of true positives plus false positives. Precision measures the classification quality of damaged components, *i.e.*, the rate at which damaged components are classified as damaged. A low precision for the classification means that safety is increased since the classifier signals that the component should be verified even if it is healthy. However, maintenance costs are increased. Recall, or sensitivity, on the other hand, is calculated as the number of correct positive classifications divided by the total number of components with damage, that is, the number of true positives divided by the number of true positives plus false negatives. Recall measures the classification quantity, *i.e.*, the rate at which the condition of the components is correctly classified. A low recall means that the classifier is missing damaged components, classifying them as healthy and, thus, decreasing safety. Thus, recall is a more important metric than precision in the damage classification problem.

The data is generated with simulations of the 64 possible damage combinations with different severity degrees for the suspension components. Damage is induced by changes in the values of m_s, m_u, c_s, c_t, k_s, and k_t, including multiple changes to simulate damage occurrences in multiple components. Table 1 presents all possible configurations, in which 10 component setups are simulated in each case, except when the state of all components corresponds to the healthy condition because it is enough to do just one simulation for this setup. When a single component is damaged, the nominal values are attributed to each healthy suspension element, and the damaged component value varies linearly for each simulation, where 10 random combinations are chosen to generate the residue signals. The same logic is applied to create the data for multiple damaged situations. In this case, the properties of the damaged components are also linearly varied from the nominal value and combined to create the residue data, where 10 random combinations are also chosen to generate the residue signals. The sprung and unsprung masses are varied between 50% and 200% of the nominal values. Thus, the SHM strategy can identify damage that leads to mass variation or provide information about the load in the vehicle, which is not a damage occurrence, for instance, a load that the driver is transporting. For the remaining components, damage types are induced by the value decrease in the damping and stiffness coefficients, reaching a reduction of up to 50% of the nominal values. The data generation procedure results in 631 signals for each residue ($r_1(t)$ and $r_2(t)$).

Table 1: Possible combinations of damage.

m_s	m_u	c_s	c_t	k_s	k_t
h	h	h	h	h	h
h	h	h	h	h	d
h	h	h	h	d	h
h	h	h	h	d	d
⋮	⋮	⋮	⋮	⋮	⋮
d	d	d	d	h	h
d	d	d	d	h	d
d	d	d	d	d	h
d	d	d	d	d	d

The complete simulated data is used to train the classifiers, where the hyperparameters are in Tab. 2. The classification performance is obtained by using the 10-fold cross-validation technique. The SHM framework accuracy is shown in Fig. 6, including the damage in multiple suspension components. In order to further evaluate the performance of each classifier, the confusion matrices are organized in Tab. 3. This layout allows the assessment of the classifier performance. Each row of these matrices represents the actual class of an instance, while each column represents the predicted classes. With the confusion matrices, it is possible to calculate the precision and recall metrics presented in Tab. 4. Once the consequences of misclassifying a damaged component as healthy are severe, the recall metric should be considered when choosing an algorithm. One can note that the three classification algorithms can identify the changes in the components associated with mass and stiffness more easily than the ones associated with dampers. Also, the SVM has the highest performance for the system, achieving an accuracy between 79.6% and 100% and the best average precision and recall. Furthermore, the decision tree achieves intermediate results compared to the other methods but is independent of data normalization. However, this method presents a lower performance in identifying tire damping changes, achieving an accuracy of 66.6%. In general, the decision tree classification accuracy is between 66.6% and 100%. Finally, the k-NN presents the worst performance in identifying the modifications in the tire damping and suspension damping coefficients, achieving an accuracy of 51.7% and 66.4%, respectively, with the worst average precision and recall. The classification accuracy of 100% indicates that the residue signals provide reasonable data to be classified. In fact, the residue signal is already a damage indicator.

Table 2: Hyperparameters.

Decision Tree	**K-NN**	**SVM**
`MinLeafSize` (1)	`NumNeighbors` (1)	`BoxConstraint` (1)
`MaxNumSplits` (630)	`Distance` (euclidean)	`KernelScale` (1)
`SplitCriterion` (gdi)	`DistanceWeight` (equal)	`KernelFunction` (linear)
`NumVariablesToSample` (all)	`Exponent` (2)	`PolynomialOrder` (3)

Table 3: Confusion matrices.

	Decision Tree	**K-NN**	**SVM**
m_s	$\begin{bmatrix}311 & 0\\ 0 & 320\end{bmatrix}$	$\begin{bmatrix}286 & 58\\ 25 & 262\end{bmatrix}$	$\begin{bmatrix}299 & 93\\ 12 & 227\end{bmatrix}$
m_u	$\begin{bmatrix}273 & 35\\ 38 & 285\end{bmatrix}$	$\begin{bmatrix}283 & 60\\ 28 & 260\end{bmatrix}$	$\begin{bmatrix}299 & 72\\ 12 & 248\end{bmatrix}$
c_s	$\begin{bmatrix}273 & 54\\ 47 & 257\end{bmatrix}$	$\begin{bmatrix}196 & 88\\ 124 & 223\end{bmatrix}$	$\begin{bmatrix}286 & 14\\ 34 & 297\end{bmatrix}$
c_t	$\begin{bmatrix}215 & 106\\ 105 & 205\end{bmatrix}$	$\begin{bmatrix}67 & 52\\ 253 & 259\end{bmatrix}$	$\begin{bmatrix}221 & 30\\ 99 & 281\end{bmatrix}$
k_s	$\begin{bmatrix}311 & 1\\ 0 & 319\end{bmatrix}$	$\begin{bmatrix}277 & 45\\ 34 & 275\end{bmatrix}$	$\begin{bmatrix}311 & 0\\ 0 & 320\end{bmatrix}$
k_t	$\begin{bmatrix}262 & 67\\ 58 & 244\end{bmatrix}$	$\begin{bmatrix}266 & 40\\ 54 & 271\end{bmatrix}$	$\begin{bmatrix}308 & 12\\ 12 & 299\end{bmatrix}$

Table 4: Precision and recall performance metrics.

	Decision Tree		**K-NN**		**SVM**	
	Precision	Recall	Precision	Recall	Precision	Recall
m_s	100%	100%	91.96%	83.14%	96.14%	76.28%
m_u	87.78%	88.64%	91%	82.51%	96.14%	80.59%
c_s	85.31%	83.49%	61.25%	69.01%	89.37%	95.33%
c_t	67.19%	66.98%	20.94%	56.3%	69.06%	88.05%
k_s	100%	99.68%	89.07%	86.02%	100%	100%
k_t	81.88%	79.64%	83.12%	86.93%	96.25%	96.25%

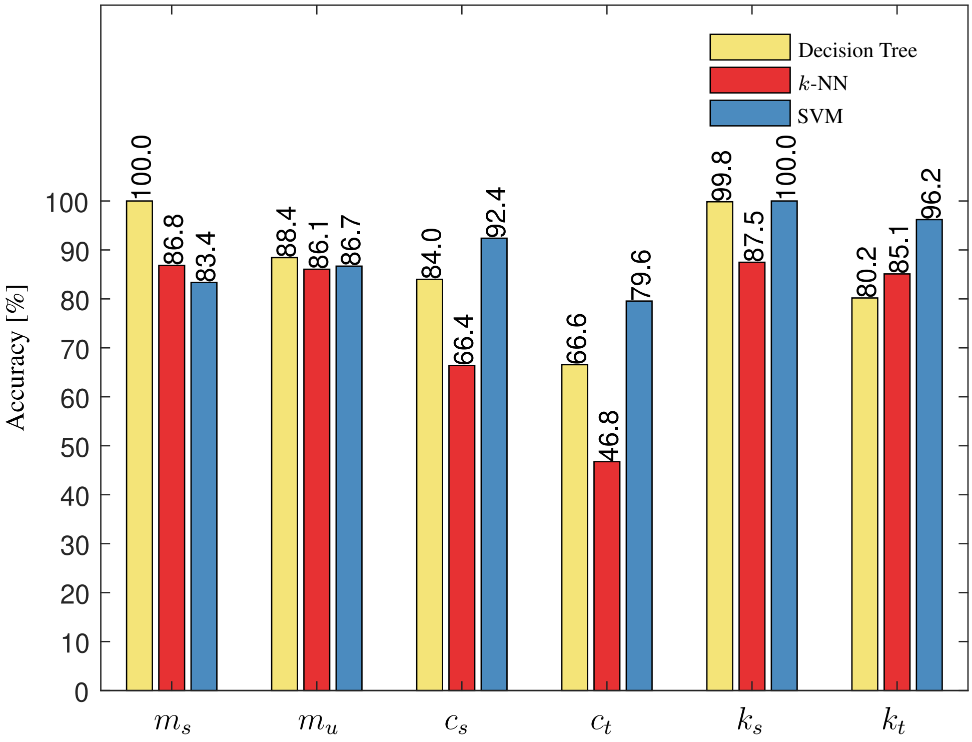


Figure 6: Classification accuracy.

5. CONCLUSIONS

An SHM framework is presented in this paper for damage detection and localization, based on a filter bank and machine learning concepts. The filter bank is built with the Luenberger method. The difference between the structure outputs and the estimated outputs signals the damage occurrence, being used for the classifier system to assess the damage. An active automotive suspension model is used as a case study structure to examine the proposed framework effectiveness. The proposed SHM methodology can detect and localize damage in the active suspension components. Simulated results show that the support vector machine method is the best tested classifier to identify changes in the suspension system, achieving a classification accuracy between 79.6% and 100%. The decision tree technique obtains intermediate performance concerning the other methods, with an accuracy between 66.6% and 100%. Finally, the k-nearest neighbor classifier achieves an accuracy between 46.8% and 86.8%. As future objectives, the framework will be experimentally validated to detect and localize damage, even in road disturbance presence.

6. REFERENCES

Alpaydin, E., 2004. *Introduction to Machine Learning*. Adaptive Computation and Machine Learning. MIT Press. ISBN 9780262012119.

Brunton, S. and Kutz, J., 2022. *Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control*. Cambridge University Press, Cambridge, 2nd edition.

Das, M., Sahu, S. and Parhi, D., 2021. "Composite materials and their damage detection using IA techniques for aerospace application: A brief review". *Materials Today: Proceedings*, Vol. 44, pp. 955–960. ISSN 2214-7853. doi:10.1016/j.matpr.2020.11.005. International Conference on Materials, Processing & Characterization.

Dougherty, G., 2012. *Pattern Recognition and Classification: An Introduction*. Springer-Verlag New York Incorporated. ISBN 9781461453222.

Franklin, G., Powell, J.D. and Emami-Naeini, A., 2010. *Feedback control of dynamic systems*. Pearson, 6th edition.

Genari, H.F.G. and Nóbrega, E.G.O., 2012. "A damage detection technique based on ARMA models distance estimation". In *Proceedings of the 1st International Symposium on Uncertainty Quantification and Stochastic Modeling - Uncertainties 2012*. São Sebastião, Brazil, pp. 140–148.

Genari, H.F.G., Nóbrega, E.G.O. and Mechbal, N., 2013. "Structural damage diagnosis method based on subspace identification metric". In *22nd International Congress of Mechanical Engineering - COBEM2013*. Ribeirão Preto, Brazil, pp. 4200–4208.

Genari, H.F.G. and Oliveira, F.M.C., 2019. "A damage-tolerant modal observer applied to the structural health monitoring of a closed-loop structure subjected to seismic events". In *Proceedings of the 25th International Congress of Mechanical Engineering - COBEM2019*. Uberlândia, Brazil.

Kullaa, J., 2010. *Vibration-Based Structural Health Monitoring Under Variable Environmental or Operational Conditions*, Springer Vienna, Vienna, pp. 107–181. ISBN 978-3-7091-0399-9. doi:10.1007/978-3-7091-0399-9_4.

Li, X., Shi, D. and Yu, Z., 2020. "Nondestructive damage testing of beam structure based on vibration response signal analysis". *Materials*, Vol. 13, No. 15. ISSN 1996-1944. doi:10.3390/ma13153301.

Meher, U. and Sunny, M.R., 2022. "Detection of multiple structural damages from drive point and cross electro-mechanical impedance signatures". *Mechanics of Advanced Materials and Structures*, Vol. 29, No. 26, pp. 4738–4758. doi:10.1080/15376494.2021.1937757.

Mousavi, A.A., Zhang, C., Masri, S.F. and Gholipour, G., 2021. "Damage detection and localization of a steel truss bridge model subjected to impact and white noise excitations using empirical wavelet transform neural network approach". *Measurement*, Vol. 185, p. 110060. ISSN 0263-2241. doi:10.1016/j.measurement.2021.110060.

Pavelko, V., Kuznetsov, S., Nevsky, A. and Marinbah, M., 2017. "Vibration-based structural health monitoring of the aircraft large component". *IOP Conference Series: Materials Science and Engineering*, Vol. 251, No. 1, p. 012093. doi:10.1088/1757-899X/251/1/012093.

Quanser, 2013. *Datasheet Active Suspension*. Quanser Inc. URL www.quanser.com/products/active-suspension.

Sarmadi, H. and Entezami, A., 2021. "Application of supervised learning to validation of damage detection". *Archive of Applied Mechanics*, Vol. 91, No. 1, pp. 393–410. ISSN 1432-0681. doi:10.1007/s00419-020-01779-z.

Sun, W., Zhao, Y., Li, J., Zhang, L. and Gao, H., 2012. "Active suspension control with frequency band constraints and actuator input delay". *IEEE Transactions on Industrial Electronics*, Vol. 59, No. 1, pp. 530–537. doi: 10.1109/TIE.2011.2134057.

Wang, Z. and Cha, Y.J., 2022. "Unsupervised machine and deep learning methods for structural damage detection: A comparative study". *Engineering Reports*, Vol. n/a, No. n/a, p. e12551. doi:10.1002/eng2.12551.

7. RESPONSIBILITY NOTICE

The authors are solely responsible for the printed material included in this paper.