Available online at www.CivileJournal.org

Civil Engineering Journal

(E-ISSN: 2476-3055; ISSN: 2676-6957)

Vol. 11, No. 06, June, 2025

Effect of Signal Filtering on Metaheuristic-Based Structural Parameter Identification in Shear Building Models

Carlos A. González-Pérez [1*], Jaime De-la-Colina [2], Jesús Valdés-González [2]

[1] *Facultad de Ingeniería, Universidad Panamericana, Álvaro del Portillo 49, Zapopan, Jalisco, 45010, Mexico.*

[2] *Facultad de Ingeniería, Universidad Autónoma del Estado de México, Cerro de Coatepec S/N, Ciudad Universitaria, Toluca, 50110, México.*

Received 21 February 2025; Revised 22 May 2025; Accepted 29 May 2025; Published 01 June 2025

Abstract

This study evaluates the effectiveness of three metaheuristic algorithms—Genetic Algorithm (GA), Differential Evolution (DE), and Particle Swarm Optimization (PSO)—for identifying lateral interstory stiffness and the modal damping ratio in two-dimensional shear building models. The main objective is to estimate these parameters using time-domain displacement, velocity, and acceleration data, assuming known floor masses and unknown input excitation that primarily excites translational vibration modes. Three structural configurations with 2, 3, and 5 stories are analyzed to assess the scalability and robustness of each algorithm. To assess the effect of signal filtering on the performance of the algorithms, white noise is added to the synthetic response data at six levels ranging from 0% to 5% of the root mean square (RMS) amplitude. A sixth-order Butterworth filter is applied to evaluate the effect of signal preprocessing, and results obtained with and without filtering are compared. The results show that all three algorithms achieve acceptable levels of accuracy, even under noisy conditions. Filtering consistently improves identification accuracy, especially in high-noise conditions. In the most challenging case (5% noise, 5-story model), the average identification errors were 5.042% for GA, 5.106% for DE, and 5.035% for PSO. The findings underscore the practical value of integrating signal filtering with metaheuristic optimization for robust structural system identification in noise-contaminated environments. To account for the random nature of the algorithms, all results reported correspond to the average of 10 independent runs per identification scenario to ensure reliable performance evaluation.

Keywords: Metaheuristic Algorithms; Genetic Algorithm; Differential Evolution; Particle Swarm Optimization; Structural Parameter Identification; Shear Buildings; Butterworth Filter; Lateral Interstory Stiffness; Damping Identification.

1. Introduction

The accurate identification of the mechanical properties of structures such as buildings, bridges, and transmission towers is essential for evaluating their dynamic behavior under seismic loads, wind, and other loads. In particular, stiffness and damping are important parameters for assessing the safety and durability of these structures. Traditionally, the determination of these properties has been carried out using classical identification methods. These techniques, widely used in research, often rely on simplified assumptions about structural behavior and may require significant prior knowledge of the system or structure [1–8]. Moreover, their applicability can be limited in complex optimization problems, where the objective functions are nonlinear or multiple local minima exist.

To overcome some of the limitations of classical methods, a promising alternative approach has emerged in recent years: system identification using metaheuristic algorithms. These methods, inspired by natural, social, or physicochemical processes, explore multiple potential solutions, making them particularly useful when information

* Corresponding author: cgonzalezp@up.edu.mx

http://dx.doi.org/10.28991/CEJ-2025-011-06-04

about the system is limited or uncertain, a common situation in the analysis of complex structures. Significant advantages of metaheuristic algorithms include: (1) a high probability of finding optimal solutions; (2) increased robustness against incomplete or noisy data; (3) adaptability to complex problems, i.e., the ability to handle multidimensional problems with a large number of parameters, making them more applicable to real-world environments; and (4) efficient handling of large search spaces. In summary, metaheuristic algorithms provide a promising approach, offering robustness to imprecise data and flexibility in tackling complex problems, making them a valuable tool for identifying structural parameters. The field continues to evolve rapidly, with recent developments proposing hybrid algorithms tailored for structural optimization in diverse civil engineering problems [9], reinforcing their adaptability and growing relevance across structural modeling tasks.

Recent studies have shown the potential of metaheuristic approaches in structural identification tasks [10–27], particularly under conditions where traditional techniques struggle due to model uncertainty or data limitations. Although individual studies exist, few have systematically benchmarked multiple metaheuristic methods under realistic noisy conditions. In addition, despite the practical relevance of signal filtering in experimental environments, its role in improving or altering the accuracy of metaheuristic-based identification has received limited attention. This study addresses these gaps through a controlled simulation framework that incorporates filtering strategies and seeks to benchmark three well-known metaheuristic methods.

This study evaluates three metaheuristic algorithms widely used in science and engineering: Genetic Algorithm (GA), Differential Evolution (DE), and Particle Swarm Optimization (PSO). The algorithms are used for the identification of key structural parameters, such as lateral interstory stiffness and modal damping, corresponding to the first two modes of vibration, in two-dimensional shear buildings. The research focuses on the ability of these methods to determine these parameters from the dynamic response of the system, considering the impact of signal filtering and the presence of noise in the measurements, which represent a common scenario in real applications.

To carry out this analysis, a methodology involving the acquisition of the dynamic response of structures in the time domain is used. Although the applied excitation is assumed to be unknown, it is considered capable of exciting the vibration modes of the system. In this study, an impulsive excitation is employed, applied to the roof of each building model. The dynamic lateral response is measured at each floor. To evaluate the precision of the metaheuristic algorithms, the response signals are contaminated with white noise at different levels (1%, 2%, ..., 5% of the RMS value). Three scenarios are analyzed to evaluate the performance of the algorithms: (1) in the absence of noise; (2) without filtering the contaminated response signals; and (3) filtering the contaminated response signals using a sixth-order Butterworth filter. An interesting aspect of this study is the evaluation of the performance of the metaheuristic algorithms when filtering is applied. In experimental settings, measurements are often contaminated by various types of noise. Filtering, depending on the type of filter applied, does not completely eliminate the original disturbances, but attenuates the unwanted components, particularly those of high frequency that are irrelevant to the structural response. This filtering process simulates a context in which initially "contaminated" signals are processed to highlight useful information. In this manner, the identification process is evaluated under conditions that more closely resemble real-world practices, where the information obtained from sensors must be processed to extract relevant parameters. This ensures that the employed methods are robust in the presence of noise and interference.

This study aims to evaluate the effectiveness of metaheuristic algorithms for identifying structural parameters in shear buildings and to explore the use of filtering techniques under noisy measurement conditions. Although comparisons with classical identification techniques are not included, this study focuses on evaluating the robustness and relative performance of metaheuristics under realistic noise and filtering conditions.

2. Literature Review

This literature review analyzes the state of the art in the application of metaheuristic algorithms for the identification of structural parameters in two-dimensional shear buildings, including approaches related to damage detection.

Wang et al. [10] explored the application of a GA to identify dynamic parameters in structural systems subjected to seismic activity. Parameters were optimized by minimizing the error between recorded and predicted responses. The methodology was validated using simulated data in single- and multi-degree-of-freedom systems (SDOF and MDOF), considering both linear and nonlinear dynamics. The tests included noise levels up to 30%, demonstrating that the GA accurately identifies the system parameters even with noisy data, highlighting its robustness and precision in all evaluated scenarios.

Casciati [11] formulated an objective function to minimize the differences between the measured and theoretical modal characteristics of a structure. The stiffness of each finite element was used as an optimization variable, and a DE algorithm generated populations to improve the fit with the measured responses until a specified tolerance was reached. Modal parameters were recalculated at each iteration using updated stiffness matrices. As a numerical example, a cantilever beam was discretized into 16 quadrilateral four-node finite elements under plane stress conditions, analyzing six damage scenarios. The comparison between the identified and initial stiffness matrices enabled effective damage detection and localization.

Xue et al. [12] examined the use of PSO to estimate parameters in linear and nonlinear structural systems. The effectiveness of PSO was validated by simulations on a nonlinear 2-DOF system with hysteresis and a 10-DOF shear building model. The PSO demonstrated good performance even with partial measurements and noisy data. For the 2-DOF system, errors ranged from 0.4% to 7.8% with noise. The PSO required fewer evaluations than GAs to converge. The analysis was conducted in the time domain with sampling every 0.01 seconds over 10 seconds. Begambre & Laier [13] introduced the Particle Swarm Optimization algorithm combined with the Simplex method (PSOS) for structural damage detection through frequency domain analysis. The PSOS integrates PSO with the Simplex method, thereby enhancing accuracy and reliability over traditional PSO. In their study, they evaluated three models: a 10-bar truss, a free-free beam with cracks, and a Duffing oscillator. The results demonstrated that PSOS accurately identifies both the location and extent of the damage, even with noisy or incomplete data, surpassing Simulated Annealing in both precision and efficiency.

Xiang & Liang [14] proposed a two-step method to detect and assess damage in thin plates. First, damage locations were identified by applying the 2D wavelet transform to the modal shape, where singularities revealed the affected areas. Second, the severity of the damage was assessed using PSO, exploring a database that related natural frequencies to damage levels. Simulations on a plate with multiple damages demonstrated the method's effectiveness. The authors recommend obtaining precise mode shapes and updating numerical models to improve accuracy in real structures. Li et al. [15] employed the Symbolization-based Differential Evolution Strategy (SDES) to identify structural parameters in two-dimensional shear building models. SDES combines symbolic analysis of time series and DE, converting numerical values into symbols to facilitate the analysis. Simulations showed that SDES outperformed PSO, providing better estimates when data were contaminated by noise. The approach uses the Euclidean distance of state frequency vectors of the symbols transformed from raw acceleration data and the DE method. SDES is reliable and effective for identifying structural systems in free and forced vibrations.

Ravanfar et al. [16] presented a method to detect damage in beams using a GA without the need for reference data. The method relies on analyzing vibration signals and uses the Relative Wavelet Packet Entropy (RWPE) to determine the location and extent of the damage. The GA optimized the selection of the mother wavelet function and the decomposition level, achieving high precision. Tests on a simply supported beam modeled with 15 finite elements demonstrated its effectiveness. The main advantage is that it does not require data from an undamaged structure, making it more practical. Ghannadi & Kourehli [17] proposed a new damage identification method using Moth-Flame Optimization (MFO), inspired by the navigation of moths. A vibration-based approach was introduced to detect damage in trusses and shear buildings, combining MFO with a new objective function that considers natural frequency and the Modal Assurance Criterion (MAC) values. This hybrid approach allowed for effective damage assessment. The finite element model for the experimental and numerical examples was created using MATLAB to extract the structure's modal properties. The mode shapes and natural frequencies of the analyzed models were contaminated with noise levels of 1% and 10%, respectively. The results showed that the method is efficient in damage identification.

Ferreira-Gomes & Alves-de-Almeida [18] presented a metaheuristic optimization approach based on the SunFlower Optimization (SFO) algorithm to identify damage in plates. The identification was performed by minimizing an objective function based on modal parameters of composite laminated structures (CFRP). To evaluate noise sensitivity, noise levels of 1%, 5%, and 10% were added to the responses. The process consisted of two steps. First, the problem was modeled using the Finite Element Method (FEM). Then, an improved algorithm called Sunflower Optimization (SFO) was applied to minimize an objective function based on modal parameters. Two numerical examples demonstrated that the method successfully identifies the location and severity of small damages.

Miao et al. [19] presented a method to identify beam damage using wavelet transform and Neural Networks (NNs). The method integrates a BP network with a GA to improve speed and accuracy. The authors analyzed modal rotation parameters using the Continuous Wavelet Transform (CWT) with the Mexican hat wavelet. Damage location was determined by peaks in wavelet coefficients, which were used as inputs in the BP network optimized by the GA to estimate damage severity. The results showed the method's effectiveness for both single and multiple damages, achieving an average error of 0.18% in the latter case. This method is precise and applicable to various structures, particularly in monitoring railway infrastructure. Guo et al. [20] explored a method that combines Wavelet transform and an Improved Particle Swarm Algorithm (WIPSO) to detect and quantify micro-damages in a beam and a single-story single-bay frame. The method identifies damage locations by detecting abrupt changes (singularities) in wavelet coefficients. Subsequently, WIPSO is employed to calculate damage severity by optimizing an objective function based on experimental data and numerical simulations. The effectiveness of WIPSO was evaluated under different scenarios, and its performance was compared with PSO, GA, and the Bat Algorithm (BA). WIPSO showed high precision in both damage localization and quantification, even in the presence of noise.

Minh et al. [21] formulated a variant of PSO, called Enhancing Particle Swarm Optimization Algorithm (EHVPSO), to locate and assess damage in a 44.05 m transmission tower modeled with approximately 900 finite elements. Four simulated damage scenarios were analyzed, involving stiffness reductions between 15% and 60%, affecting between 3 and 6 elements. Noise levels of 3%, 6%, and 12% were introduced to evaluate the algorithm's robustness. To describe

changes in frequencies and modal shapes, the Multiple Damage Location Assurance Criteria (MDLAC) and the Modal Assurance Criterion (MAC) were incorporated into the proposed objective function. The results showed that the algorithm can detect damage with high precision and reliability. Minh et al. [22] used a bio-inspired metaheuristic optimization algorithm called Termite Life Cycle Optimizer (TLCO) to estimate the location and severity of damage in high-rise concrete structures. TLCO simulates the behavior and life cycle of termites to address complex optimization problems. The authors formulated an inverse problem to estimate damage, minimizing the difference between the measured values and those predicted by the updated finite element model.

Zar et al. [23] present a 2024 review on vibration-based Structural Damage Detection (SDD) and its integration with Machine Learning (ML) and Deep Learning (DL) methods. The authors emphasize persistent challenges such as structural nonlinearities, environmental noise, and the ill-posed nature of inverse problems, issues also relevant to optimization-based identification. The review underscores the need for robust, noise-resilient strategies and highlights hybrid approaches and unsupervised learning as promising directions for continuous monitoring and early damage detection. Li et al. [24] propose an innovative Structural Damage Identification (SDI) method integrating a weighted average-based surrogate model combining three popular approaches with an improved bio-inspired optimizer, the Improved Termite Life Cycle Optimizer (ITLCO). The integrated surrogate model enhances predictive accuracy and generalization, while ITLCO incorporates a novel autonomous movement strategy that strengthens robustness and the ability to escape local optima, addressing challenges typical in nonlinear, multimodal structural optimization. Applied to a laboratory-scale dam model with simulated damage, the method achieves over a computational efficiency improvement of over 100 times compared to traditional FE model updating techniques without compromising accuracy. The study reveals that individual surrogate model accuracy does not directly correlate with SDI effectiveness, underscoring the importance of model integration and algorithm design. Recognized limitations include the computational complexity of probabilistic finite element analysis and reliance on limited modal parameters for precise damage localization. Overall, this work significantly advances metaheuristic optimization in structural dynamics by providing an efficient, accurate, and scalable framework for damage detection in large-scale structures.

Hernández-González & García-Macías [25] proposed a supervised damage identification strategy based on populations of competing FE models, each parameterized to represent a specific failure mechanism. To enable computational efficiency suitable for continuous Structural Health Monitoring (SHM), Kriging-based surrogate models were used to approximate the modal properties (natural frequencies and mode shapes) of each FE model. The inverse problem was solved using PSO, and model selection was performed via the Bayesian Information Criterion (BIC), balancing prediction accuracy and model complexity. The approach was validated on a 25-bar planar truss with synthetic noisy data and on the Muhammad Tower, located in Granada, Spain, using real vibration measurements combined with damage scenarios generated via nonlinear analyses. Results confirmed that multi-model formulations with BIC enhance damage localization and the identification of activated failure mechanisms under limited modal observability.

Alemu et al. [26] proposed a data-driven damage detection method using 680 vibration signals recorded from a three-story steel frame. The study introduces the Most Damage-Sensitive Segment (MDSS), a novel spectral feature extraction technique that enhances classification accuracy under environmental variability. Their method achieves over 80% accuracy in detecting, localizing, and grading damage, emphasizing the importance of robust preprocessing in high-variability field conditions.

Farhadi et al. [27] develop a two-stage framework for damage identification in truss structures using modal data. The first stage employs the Residual Force Vector (RFV) and Total Element Discrepancy Index (TEDI) for damage localization; the second uses PSO, DE, and ECBO for severity quantification via model updating. To reduce computational cost, a PSO-trained GMDH surrogate model is employed, retaining high accuracy even with incomplete modal data, and demonstrating the viability of metaheuristic optimization in large-scale structural health monitoring. Numerical experiments on three large-scale benchmarks, a 47-bar power-line tower, a 200-bar double-layer grid, and a 200-bar planar truss, demonstrate that the framework locates and quantifies damage accurately, and that the surrogate retains virtually the same precision even with only twelve sensors, confirming robustness under incomplete modal information.

The precise identification of lateral interstory stiffnesses and damping in two-dimensional shear building models is fundamental for structural analysis and design. Although metaheuristic algorithms such as GA, PSO, and DE have demonstrated effectiveness in structural parameter identification and damage detection, there is limited research in the literature regarding the impact of signal filtering on the accuracy of these algorithms. Filtering is a common practice to reduce noise in measurements, but it can alter the characteristics of the signals and affect the identification of critical parameters. This work addresses the effect of signal filtering on metaheuristic algorithms for the identification of stiffnesses and damping in two-dimensional shear buildings. Understanding this impact is important because it allows for improving the robustness and reliability of these algorithms in real applications. The expected findings may lead to more precise methods for structural health monitoring and early damage detection, contributing to the advancement of the state of the art in structural engineering and improving safety and efficiency in infrastructure management.

3. Problem Formulation

The problem formulation is detailed in the following subsections, which describe the approach for identifying key parameters of two-dimensional shear buildings. The methodology followed in the present work is briefly summarized in Figure 1, which illustrates the main steps of the analysis.

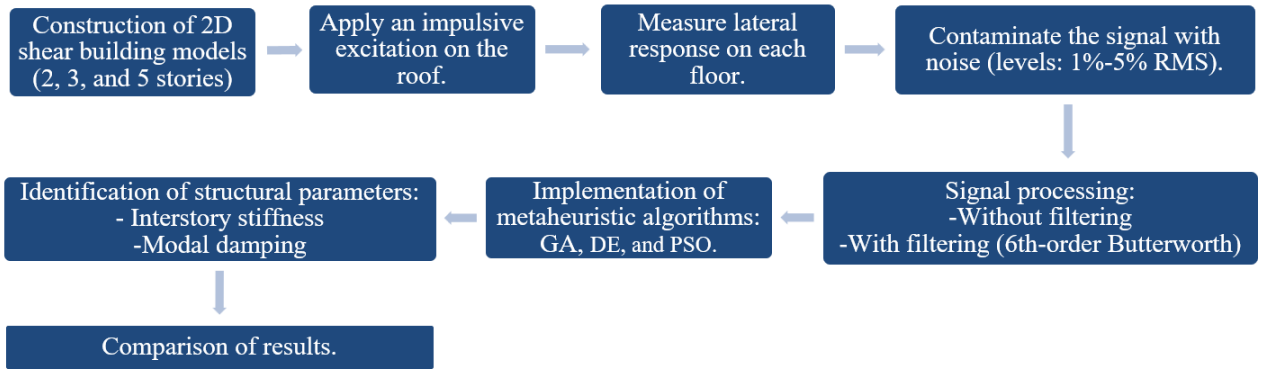


Figure 1. Flowchart of the proposed methodology

3.1. Problem Definition

This study aims to identify the lateral interstory stiffnesses ($\boldsymbol{k} = [k_1, k_2, \cdots, k_N]^T$) and the modal damping ratio (ζ) of two-dimensional shear buildings using a Rayleigh-type damping model. The system is modeled as a Multi-Degree-Of-Freedom (MDOF) system, minimizing the discrepancy between the simulated and measured responses under impulsive excitation. Three shear buildings with 2, 3, and 5 stories are analyzed. To identify the values of $\boldsymbol{k}$ and ζ, time-measured displacements are used. It is assumed that the mass matrix ($\boldsymbol{M}$) and the first two translational modal frequencies (ω_1 and ω_2) are known. However, due to inherent uncertainties in the experimental or numerical determination of modal parameters, a ±7% variation is considered in the estimation of ω_1 and ω_2, as achieving absolute accuracy in the identification of the building's fundamental frequencies is practically unachievable. The impulsive excitation is applied at the top level of the building, assuming that it excites the translational modes necessary for the accurate identification of the parameters.

3.2. Equation of Motion

The structural system is modeled using the following equation of motion for a MDOF system:

$$\boldsymbol{M}\ddot{\boldsymbol{u}}(t) + \boldsymbol{C}(\zeta)\dot{\boldsymbol{u}}(t) + \boldsymbol{K}(k_i)\boldsymbol{u}(t) = \boldsymbol{f}(t) \tag{1}$$

Such that:

$$\boldsymbol{M} = \begin{bmatrix} m_1 & 0 & 0 & \cdots & 0 \\ 0 & m_2 & 0 & \cdots & 0 \\ 0 & 0 & m_3 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & m_N \end{bmatrix}; \boldsymbol{K} = \begin{bmatrix} k_1 + k_2 & -k_2 & 0 & \cdots & 0 \\ -k_2 & k_2 + k_3 & -k_3 & \cdots & 0 \\ 0 & -k_3 & k_3 + k_4 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & k_{N-1} + k_N \end{bmatrix}; \boldsymbol{f}(t) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ f_N(t) \end{bmatrix} \tag{2}$$

where $\boldsymbol{M}$ is the diagonal mass matrix, which is known and constant for each level of the building; $\boldsymbol{C}(\zeta)$ is the Rayleigh-type proportional damping matrix, which depends on the modal damping ratio ζ, and is expressed as $\boldsymbol{C}(\zeta) = \alpha\boldsymbol{M} + \beta\boldsymbol{K}(k_i)$, where α and β are damping coefficients defined as $\alpha = \zeta\frac{2\omega_1\omega_2}{\omega_1+\omega_2}$ and $\beta = \zeta\frac{2}{\omega_1+\omega_2}$. Here, ω_1 and ω_2 are the first and second modal frequencies of vibration of the system, respectively; $\boldsymbol{K}(k_i)$ is the stiffness matrix that depends on the individual story stiffness values $k_1, k_2, \cdots, k_N$; $\boldsymbol{f}(t)$ is the vector of external forces applied to the system, in this case, an impulsive excitation applied at the top level; $\boldsymbol{u}(t)$, $\dot{\boldsymbol{u}}(t)$, and $\ddot{\boldsymbol{u}}(t)$ correspond to the displacement, velocity, and acceleration vectors of each of the building levels, respectively.

3.3. Objective Function

The objective is to identify the lateral interstory stiffness values $(k_1, k_2, \cdots, k_N)$, where N is the number of building stories, and the modal damping ratio (ζ), which, when used in the simulation of the system's response, accurately reproduces the measured responses. This formulation allows for a direct comparison between observed (measured) and simulated dynamic responses, enabling parameter optimization.

The optimization of the structural parameter vector $\hat{\boldsymbol{p}} = [\zeta \quad k_1 \quad k_2 \quad \cdots \quad k_N]^T$ is carried out using a metaheuristic algorithm. This vector is extended to facilitate the optimization process as $\boldsymbol{p} = [\boldsymbol{p}_0 \quad \boldsymbol{p}_1 \quad \boldsymbol{p}_2]^T$ such that $\boldsymbol{p}_0 = \alpha$, $\boldsymbol{p}_1 = \beta[k_1 \quad k_2 \quad \cdots \quad k_N]^T$, and $\boldsymbol{p}_2 = [k_1 \quad k_2 \quad \cdots \quad k_N]^T$. With this, the equation of motion (1) is rewritten as:

$$(\boldsymbol{A}_0 \quad \boldsymbol{A}_1 \quad \boldsymbol{A}_2)\begin{Bmatrix}\boldsymbol{p}_0\\ \boldsymbol{p}_1\\ \boldsymbol{p}_2\end{Bmatrix} = \hat{\boldsymbol{f}} \tag{3}$$

such that $\hat{\boldsymbol{f}} = \boldsymbol{f} - \boldsymbol{M}\ddot{\boldsymbol{u}}$, and $\boldsymbol{A}_0$, $\boldsymbol{A}_1$, and $\boldsymbol{A}_2$ are functions of time defined from $t = t_0$ to $t = t_m$, as follows:

$$\boldsymbol{A}_0 = [m_1\ddot{u}_1 \quad m_2\ddot{u}_2 \quad \cdots \quad m_N\ddot{u}_N]^T \tag{4}$$

$$\boldsymbol{A}_1 = \begin{bmatrix} \dot{u}_1 & \dot{u}_1 - \dot{u}_2 & 0 & \cdots & 0 \\ 0 & \dot{u}_2 - \dot{u}_1 & \dot{u}_2 - \dot{u}_3 & \cdots & 0 \\ 0 & 0 & \dot{u}_3 - \dot{u}_2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \dot{u}_{N-1} - \dot{u}_N \\ 0 & 0 & 0 & \cdots & \dot{u}_N - \dot{u}_{N-1} \end{bmatrix} \tag{5}$$

$$\boldsymbol{A}_2 = \begin{bmatrix} u_1 & u_1 - u_2 & 0 & \cdots & 0 \\ 0 & u_2 - u_1 & u_2 - u_3 & \cdots & 0 \\ 0 & 0 & u_3 - u_2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & u_{N-1} - u_N \\ 0 & 0 & 0 & \cdots & u_N - u_{N-1} \end{bmatrix} \tag{6}$$

It is easy to see that $\boldsymbol{A}_0\boldsymbol{p}_0$ represents $\boldsymbol{M}\ddot{\boldsymbol{u}}(t)$, $\boldsymbol{A}_1\boldsymbol{p}_1$ represents $\boldsymbol{C}(\zeta)\dot{\boldsymbol{u}}(t)$, and $\boldsymbol{A}_2\boldsymbol{p}_2$ represents $\boldsymbol{K}(k_i)\boldsymbol{u}(t)$. To minimize Equation 3, we define a cost function (or objective function) that measures the squared error between the left-hand side of the equation and $\hat{\boldsymbol{f}}$ for each time t. Thus, the cost function J is defined by:

$$J(\hat{\boldsymbol{p}}) = \sum_{t=t_0}^{t_m} \left\| [\boldsymbol{A}_0(t) \quad \boldsymbol{A}_1(t) \quad \boldsymbol{A}_2(t)]\hat{\boldsymbol{p}} - \hat{\boldsymbol{f}}(t) \right\|^2 \tag{7}$$

where $\|\cdot\|$ represents the L^2 norm (or Euclidean norm), and the sum extends from $t = t_1$ to $t = t_m$.

By minimizing the function $J(\hat{\boldsymbol{p}})$ with respect to the vector of structural parameters $\hat{\boldsymbol{p}}$, the discrepancy between the simulated dynamic responses and the measured responses is minimized. This minimization allows the identification of key parameters, such as lateral interstory stiffnesses and modal damping ratio.

To evaluate the accuracy of the metaheuristic algorithms in the estimation of the structural parameters, the Mean Absolute Percentage Error (MAPE) is used. This metric measures the average percentage error between the actual values of the parameters and the values estimated by each metaheuristic algorithm. The MAPE is defined as:

$$\text{MAPE} = \frac{1}{N_p}\sum_{i=1}^{N_p}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100 \tag{8}$$

where y_i represents the actual value of the parameter, $\hat{y}_i$ is the value estimated by the metaheuristic algorithm, and N_p is the total number of parameters to be identified. A lower MAPE value indicates a higher accuracy in the estimation of the parameters.

4. Description of Metaheuristic Algorithms

The identification of structural parameters in two-dimensional shear building models often involves solving optimization problems with large search spaces and data that may be incomplete or noisy. To address these challenges, this study focuses on the evaluation of three widely used metaheuristic algorithms: the Genetic Algorithm (GA), Differential Evolution (DE) and Particle Swarm Optimization (PSO). The following sections briefly describe these algorithms.

4.1. Genetic Algorithm (GA)

GA is probably the most widely known of the evolutionary algorithms (EAs). In their original formulation, GAs used bit strings. However, GAs are not constrained to bit strings. Indeed, Davis [28], Janikow & Michalewicz [29], as well as Wright [30] were the pioneers who developed GAs with real parameters, which is the formulation used in this paper. A Genetic Algorithm (GA) consist of six fundamental steps.

The first step, *Initialization*, consists of randomly generating a population, where each individual represents a possible solution to the optimization problem. Each individual (chromosome) is a vector of real values, and each element of the vector corresponds to a variable or parameter to be optimized (gene). The initial values for the population are calculated as follows:

$$x_{i,j}^{1} = x_j^{min} + r_{i,j}(x_j^{max} - x_j^{min}) \quad (9)$$

For $i = 1 \cdots N$, and $j = 1 \cdots n$, where N is the number of individuals (chromosomes); n is the number of parameters to identify (genes); $r_{i,j}$ is a uniformly distributed random number $U[0,1]$ generated independently for each individual and each variable; $x_{i,j}^{1}$ represents the initial value of the j-th variable of the i-th individual of the population, the superscript "1" indicates the first (initial) generation; x_j^{min} is the lower limit of the valid range for the j-th variable; x_j^{max} is the upper limit of the valid range for the j-th variable. During this stage, the algorithm parameters are set: population size, crossover rate, mutation rate, and stopping criterion.

In the second stage, *Evaluation*, the fitness of each individual in the population is evaluated using the objective function of the problem.

In the third phase, *Parent Selection*, individuals are selected from the population to participate in breeding, using an appropriate selection method. A wide variety of *parent selection* methods exist, such as *Tournament Selection*; *Rank Selection*; *Uniform Parent Selection*; *Roulette Wheel Selection*, and *Steady State Selection*. In this study, tournament selection is used due to its simplicity and the fact that selection pressure is easy to control by varying the tournament size [31].

In the fourth stage, *Crossover* (or *Recombination*), new solutions (*offspring*) are generated by randomly combining the genetic information of the *parents*. It is analogous to the crossover process that occurs during sexual reproduction. The most common crossover operators for continuous variables are: *Arithmetic Crossover*; *Intermediate Crossover*; *Exponential Crossover*; *Blend Alpha Crossover*; *Simulated Binary Crossover* (SBX); *Blend Crossover* (BLX-α), among others. This operator requires a high crossover probability ($p_c \geq 0.90$). In this case, a crossover probability of $p_c = 0.90$ is used, in conjunction with the SBX crossover method. In SBX, two parent solutions $x_{i,j}^{(1,\bar{k})}$ and $x_{k,j}^{(2,\bar{k})}$ are selected, where i and k denote the indices of the parent individuals such that $i \neq k$ and j refers to the parameter index. These parents generate two new offspring solutions $x_{i,j}^{(1,\bar{k}+1)}$ and $x_{k,j}^{(2,\bar{k}+1)}$ in generation $\bar{k}+1$, where $\bar{k}$ indicates the current iteration number. The offspring are computed as:

$$\begin{aligned} x_{i,j}^{(1,\bar{k}+1)} &= 0.5\left[\left(1+\beta_q\right)x_{i,j}^{(1,\bar{k})} + \left(1-\beta_q\right)x_{k,j}^{(2,\bar{k})}\right] \\ x_{k,j}^{(2,\bar{k}+1)} &= 0.5\left[\left(1-\beta_q\right)x_{i,j}^{(1,\bar{k})} + \left(1+\beta_q\right)x_{k,j}^{(2,\bar{k})}\right] \end{aligned} \quad (10)$$

where β_q is the spread factor that controls the distribution of the offspring with respect to the parents. Values of $\beta_q > 1$ generate offspring outside the range defined by the parents, while $\beta_q < 1$ results in offspring closer to each other. The value of β_q is computed as shown below:

$$\beta_q = \begin{cases} \left(2\bar{r}_{i,k,j}\right)^{\frac{1}{\eta_c+1}} & \text{if } \bar{r}_{i,k,j} \leq 0.50 \\ \left(\frac{1}{2\left(1-\bar{r}_{i,k,j}\right)}\right)^{\frac{1}{\eta_c+1}} & \text{otherwise} \end{cases} \quad (11)$$

In these equations, $\bar{r}_{i,k,j}$ is a uniformly distributed random variable $U[0,1]$; η_c is a distribution parameter that controls the distribution of the children around the parents. Here $\eta_c = 2$ is used, which is a value suggested by Deb & Agrawal [32] for single-objective optimization.

Mutation: In the fifth stage of GA, genetic variability is introduced into the population by randomly modifying one or more genes (real values) of the newly created individuals. This process avoids premature convergence to suboptimal solutions and ensures a complete exploration of the search space, including regions that could be omitted by the crossover operator. The most commonly utilized mutation schemes include the following: *Random Mutation*, *Uniform Mutation*, *Non-Uniform mutation*, *Gaussian Mutation*, and *Polynomial Mutation*. For this operator to be executed, a relatively low probability is used, such that $p_m = \frac{1}{n}$ where n is the number of variables to be optimized [33]. In this paper, the polynomial mutation scheme proposed by Deb & Agrawal [34] is used, which has been successfully applied in the resolution of single-objective and multi-objective optimization problems [35]. In polynomial mutation, a polynomial probability distribution is employed to perturb a solution in the vicinity of a parent, ensuring that the mutation does not result in any values outside the specified interval $[x_j^{min}, x_j^{max}]$, for a parent solution [36]. The result of applying this operator is given by:

$$y_{i,j}^{(1,\bar{k}+1)} = x_{i,j}^{(1,\bar{k}+1)} + \left((x_j^{max} - x_j^{min})\bar{\delta}_m\right) \quad (12)$$

Such that:

$$\bar{\delta}_m = \begin{cases} \left[(2r_i)^{\frac{1}{n_m+1}}\right] - 1 & \text{if } r_i < 0.50 \\ 1 - (2-2r_i)^{\frac{1}{n_m+1}} & \text{if } r_i \geq 0.50 \end{cases} \quad (13)$$

where $y_{i,j}^{(1,\bar{k}+1)}$ is the mutation result at iteration $\bar{k}+1$; $\bar{\delta}_m$ is a parameter calculated from the polynomial probability distribution $P(\delta) = 0.5(n_m+1)(1-|\delta|)^{\eta_m}$; r_i is a uniformly distributed random variable $U[0,1]$, and η_m is the polynomial mutation rate, which is set to $\eta_m = 20$.

Replacement: The sixth stage consists of selecting the individuals that will form the new population in each generation of a genetic algorithm. Although the average fitness of the genetic algorithm population usually increases with the number of generations (iterations), it is always possible that at some point the best individuals of each generation may be lost. This is due to the fact that selection, crossover, and mutation operators modify the individuals in the process of creating the subsequent generation [37]. This stage is crucial for the success of the algorithm, since it influences the diversity of the population, the speed of convergence, and the quality of the final solution. The most commonly used replacement schemes in real parameter GAs include: *Generational Replacement*; *Steady-State Replacement*; *Ranked Replacement*; *Elitism*, and the *Survival Strategy* $(\mu+\lambda)$, among others. In this case, the *Survival Strategy* $(\mu+\lambda)$ is used. In this strategy, the populations of parents (μ) and offspring (λ) are combined, and the best *N* individuals are selected.

The algorithm proceeds iteratively until either a satisfactory solution is found or the maximum number of iterations is reached.

4.2. Differential Evolution (DE)

DE is a population-based optimization algorithm, which means that it works with a set of candidate solutions (the population) that evolves through iterations. This algorithm consists of four fundamental steps: *Initialization*, *Differential Mutation*, *Recombination*, and *Selection*. The first step is performed at the beginning of the search execution and consists of generating a first generation with an initial population of *N* random individuals using the same expression as in the GAs (Equation 9), with $i = 1 \cdots N$, $j = 1 \cdots n$, where *N* is the number of individuals and *n* is the number of parameters to identify.

The second stage, *Differential Mutation*, consists of the creation of a new candidate solution $\boldsymbol{\eta}_i^{\bar{k}}$, also called mutated individual, corresponding to the $\bar{k}$-th iteration. This new solution is formed from the combination of three randomly chosen individuals, called target vectors, $\boldsymbol{x}_{r1}^{\bar{k}}$, $\boldsymbol{x}_{r2}^{\bar{k}}$, $\boldsymbol{x}_{r3}^{\bar{k}}$ from the current population, such that:

$$\boldsymbol{\eta}_i^{\bar{k}} = \boldsymbol{x}_{r1}^{\bar{k}} + F \cdot \left(\boldsymbol{x}_{r2}^{\bar{k}} - \boldsymbol{x}_{r3}^{\bar{k}}\right) \tag{14}$$

with i, r_1, r_2, and r_3 all distinct, and F is a scaling factor that controls the mutation intensity. Wang et al. [38] and Ho-Huu et al. [39] consider that an acceptable value of F should be between [0.4, 1.0]. In this work, we use $F = 0.50$, a value suggested by Das & Suganthan [40]. In the third stage, *Recombination* (*Crossover*), a new test individual $t_{i,j}^{\bar{k}}$ is generated by combining the mutated individual $\eta_{i,j}^{\bar{k}}$with a parent individual $x_{i,j}^{\bar{k}}$ from the current population, such that:

$$t_{i,j}^{\bar{k}} = \begin{cases} \eta_{i,j}^{\bar{k}} & r_{i,j} \le p_c \text{ or } j = j_{\text{rand}} \\ x_{i,j}^{\bar{k}} & \text{Otherwise} \end{cases} \tag{15}$$

where $r_{i,j} \sim U[0,1]$, j_{rand} is a randomly chosen index in the set $\{1,2\ldots,n\}$, and p_c is the crossover probability. Storn & Price [41] suggest $0.10 \le p_c \le 0.90$; here, we use $p_c = 0.5$.

Finally, in the *Selection* stage, the fitness of the test individual $\boldsymbol{t}_i^{\bar{k}}$ is compared with the fitness of the parent individual $\boldsymbol{x}_i^{\bar{k}}$. The individual with the highest fitness—i.e., the best solution—$\boldsymbol{x}_i^{\bar{k}+1}$, is selected to be part of the new population:

$$\boldsymbol{x}_i^{\bar{k}+1} = \begin{cases} \boldsymbol{t}_i^{\bar{k}} & f\left(\boldsymbol{t}_i^{\bar{k}}\right) < f\left(\boldsymbol{x}_i^{\bar{k}}\right) \\ \boldsymbol{x}_i^{\bar{k}} & \text{Otherwise} \end{cases} \tag{16}$$

The algorithm proceeds iteratively until a satisfactory solution is found or the maximum number of iterations is reached.

4.3. Particle Swarm Optimization (PSO)

PSO was developed in 1995 by James Kennedy & Eberhart [42]. This stochastic algorithm, belonging to the family of evolutionary algorithms, has proven to be efficient and effective in a wide range of applications, including function optimization, system identification, process control, and machine learning problem solving [42-44]. The PSO algorithm uses a swarm of particles that explore the search space looking for optimal solutions. Each particle represents a possible solution, with a position and velocity that are updated at each iteration. This update is based on their own experience (local behavior) and on the shared information of the swarm (global behavior), allowing them to converge towards more favorable solutions. A summary of the four steps involved in this method is presented below.

PSO *Initialization* represents the first step in the process by which the fundamental basis of the search is established. At this stage, particles are randomly distributed in the search space, with velocities also randomly assigned, resulting in an initial swarm configuration. In the initial iteration $(\bar{k} = 1)$, a swarm $\boldsymbol{X}^1$ composed of N particles is created, forming a population represented by $\boldsymbol{X}^1 = \{\boldsymbol{x}_1^1, \boldsymbol{x}_2^1, \ldots, \boldsymbol{x}_N^1\}$; as defined in Equation 9. Simultaneously, an initial velocity vector is defined for each particle, $\boldsymbol{v}_i^{\bar{k}=1}$. One practice is to initially set the velocity of the particles as $\boldsymbol{v}_i^{\bar{k}=1} = \boldsymbol{0}$. However, a uniform distribution can also be used to generate random velocities within a specific range. In this study, the latter approach is adopted, allowing a more diversified initial exploration of the search space. The best individual $\boldsymbol{p}_i^1$ and global $\boldsymbol{g}^1$ positions are initialized as: $\boldsymbol{p}_i^1 = \boldsymbol{x}_i^1$ and $\boldsymbol{g}^1 = \underset{i=1,\ldots,N}{\operatorname{argmin}} f(\boldsymbol{x}_i^1)$, respectively.

The second stage of the PSO algorithm consists of updating both the velocity and the position of each particle. Thus, the velocity of the i -th particle is defined as:

$$\boldsymbol{v}_i^{\bar{k}+1} = \omega \boldsymbol{v}_i^{\bar{k}} + c_1 r_1 \left(\boldsymbol{p}_i^{\bar{k}} - \boldsymbol{x}_i^{\bar{k}}\right) + c_2 r_2 \left(\boldsymbol{g}^{\bar{k}} - \boldsymbol{x}_i^{\bar{k}}\right) \tag{17}$$

The first term $(\omega \boldsymbol{v}_i^{\bar{k}})$ of Equation 17 represents the inertial component of the motion. According to Shi & Eberhart [45], a suitable choice of ω should be in the range between 0.80 and 1.40. Here, a mean value is taken, i.e., ω = 1.10. The second term, $c_1 r_1 \left(\boldsymbol{p}_i^{\bar{k}} - \boldsymbol{x}_i^{\bar{k}}\right)$, represents the cognitive component, while the third term $c_2 r_2 \left(\boldsymbol{g}^{\bar{k}} - \boldsymbol{x}_i^{\bar{k}}\right)$ represents the influence of the overall population on the individual. c_1 and c_2 are the cognitive and social acceleration coefficients, respectively. According to Carlisle & Dozier [44], $c_1 + c_2 \leq 4$. Here, $c_1 = c_2 = 2.0$ is assumed, which are values suggested by Kennedy & Eberhart [42]. r_1 and r_2 are uniformly distributed random numbers in the range [0,1]; $\boldsymbol{x}_i^{\bar{k}}$ and $\boldsymbol{v}_i^{\bar{k}}$ are the position and velocity vectors of the i -th particle, corresponding to the n -th variable to be optimized, at the $\bar{k}$-th iteration. Subsequently, the position of each particle in the search space is updated using:

$$\boldsymbol{x}_i^{\bar{k}+1} = \boldsymbol{x}_i^{\bar{k}} + \boldsymbol{v}_i^{\bar{k}+1} \tag{18}$$

Finally, in the third stage (*Evaluation* of $\boldsymbol{x}_i^{\bar{k}+1}$), the performance of the population in the $\bar{k}+1$-th iteration is evaluated, i.e., $f\left(\boldsymbol{x}_i^{\bar{k}+1}\right)$ is computed, and the best individual position of particle i, $\boldsymbol{p}_i^{\bar{k}+1}$, and the best global position, $\boldsymbol{g}^{\bar{k}+1}$, are updated using Equations 19 and 20, respectively.

$$\boldsymbol{p}_i^{\bar{k}+1} = \begin{cases} \boldsymbol{x}_i^{\bar{k}+1}, \text{if } f\left(\boldsymbol{x}_i^{\bar{k}+1}\right) < f\left(\boldsymbol{p}_i^{\bar{k}}\right) \\ \boldsymbol{p}_i^{\bar{k}}, \text{otherwise} \end{cases} \tag{19}$$

$$\boldsymbol{g}^{\bar{k}+1} = \begin{cases} \boldsymbol{p}_i^{\bar{k}+1}, \text{ if } f\left(\boldsymbol{p}_i^{\bar{k}+1}\right) < f\left(\boldsymbol{g}^{\bar{k}}\right) \text{ for any } i \in \{1, \ldots, N\} \\ \boldsymbol{g}^{\bar{k}}, \text{ otherwise} \end{cases} \tag{20}$$

Iterations continue until a satisfactory solution is found or the predefined maximum number of iterations is reached.

4.4. Configuration of Metaheuristic Algorithms

This section details the configuration of the metaheuristic algorithms used to identify structural parameters in three shear-building models with 2, 3, and 5 stories. These parameters include two or more lateral interstory stiffnesses, along with the modal damping ratio. The configuration for each case was established based on the fact that the number of parameters increases with the number of stories and on the empirical behavior observed in exploratory trials. To promote a balance between global exploration and local exploitation of the search space, a population size of 50 individuals or particles was selected for the 2- and 3-story models, and 100 for the 5-story model. This decision was based on the fact that, as the number of stories increases, so does the number of lateral interstory stiffnesses to be identified, which expands the search space and increases the risk of convergence stagnation. In this context, a larger population improves diversity, enhances space coverage, and reduces premature convergence.

In addition, exploratory trials were conducted to determine appropriate maximum iteration limits, taking into account the convergence behavior and execution variability of each algorithm. For the 3- and 5-story models, generous limits of 10,000 and 15,000 iterations were selected to ensure robust convergence. For the 2-story model, 5,000 iterations proved sufficient to achieve stable and reproducible results, since only three parameters needed to be identified: two lateral interstory stiffnesses and the modal damping ratio, resulting in a simpler objective function topology.

The algorithms employed two stopping criteria: the first based on a fixed number of iterations, and the second on a population stabilization condition defined by the relative convergence index Δ:

$$\Delta = \left| \frac{f_{mean}}{f_{best}} - 1 \right| \tag{21}$$

where f_{mean} is the average value of the objective function in the population, and f_{best} is the best value obtained. Convergence was considered to have been reached when $\Delta \leq 10^{-10}$.

Regarding the variable bounds, for the modal damping ratio, a range of [0.01, 0.10] was established for all the models analyzed. Although typical values for low-rise structures are usually between 1% and 6%, a wider interval was adopted to account for model uncertainties, absorb possible noise effects, and allow for flexibility during the optimization process. As for the lateral interstory stiffnesses, the lower and upper bounds were set at 20% and 300% of the nominal value for the 2- and 3-story models, and at 40% and 180% for the 5-story model, respectively. This choice was made to reduce parametric dispersion, avoid non-physical configurations, and promote faster convergence in the higher-dimensional optimization problem. For the 2- and 3-story models, these issues were less relevant, making broader intervals acceptable without detriment to performance. To account for the stochastic nature of the algorithms, each identification scenario was executed 10 times independently. All reported results correspond to the average values computed over these independent runs.

5. Building Models and Input Forces

This study analyzes three shear building models with 2, 3 and 5 stories, modeled in two dimensions. Rayleigh-type damping is assumed for all models. The masses of each level (m_i), and the first two translational vibration frequencies (ω_1 and ω_2) are considered known, acknowledging an inherent uncertainty of ±7% in their estimation due to practical limitations in the field measurement of vibration frequencies. In all cases analyzed, a modal damping ratio of $\zeta = 0.05$ is adopted for the first two modes. The structural properties (masses, stiffnesses, and vibration frequencies) of the analyzed models are shown in Figure 2.

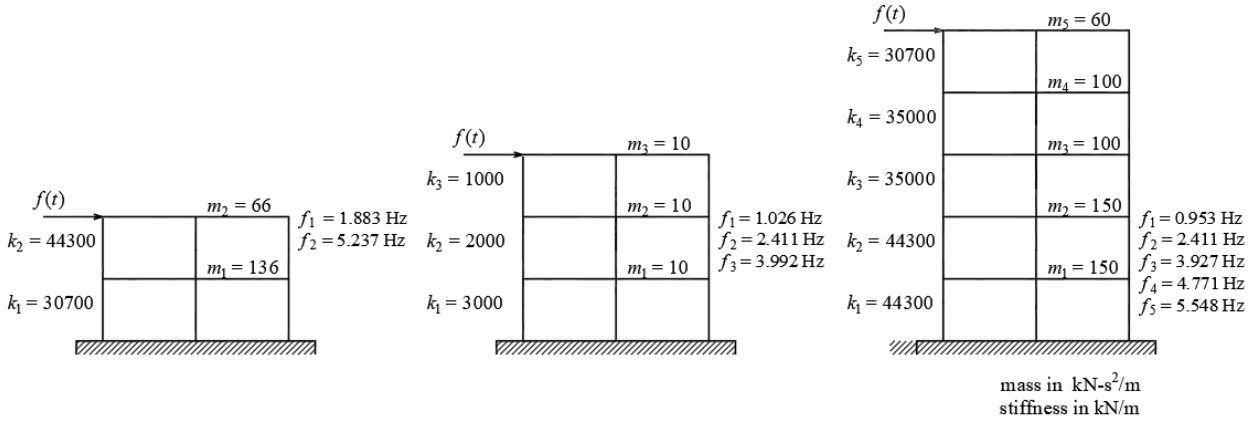


Figure 2. Structural properties of the studied building models

For the models under investigation, the lateral dynamic excitation applied at the upper level is considered unknown. However, for simulation purposes, an impulsive excitation is used to excite all modes of vibration. The Fourier amplitude spectra of the excitations are shown in Figures 3 to 5. These figures also show, with gray lines, the modal vibration frequencies of the models. It is assumed that the system responses (displacements, velocities, and accelerations) are measured, and noise is then added to these signals to simulate realistic recordings. Six noise levels are considered: 0%, 1%, ..., 5% of the Root Mean Square (RMS) value, which represents the average intensity of the signal.

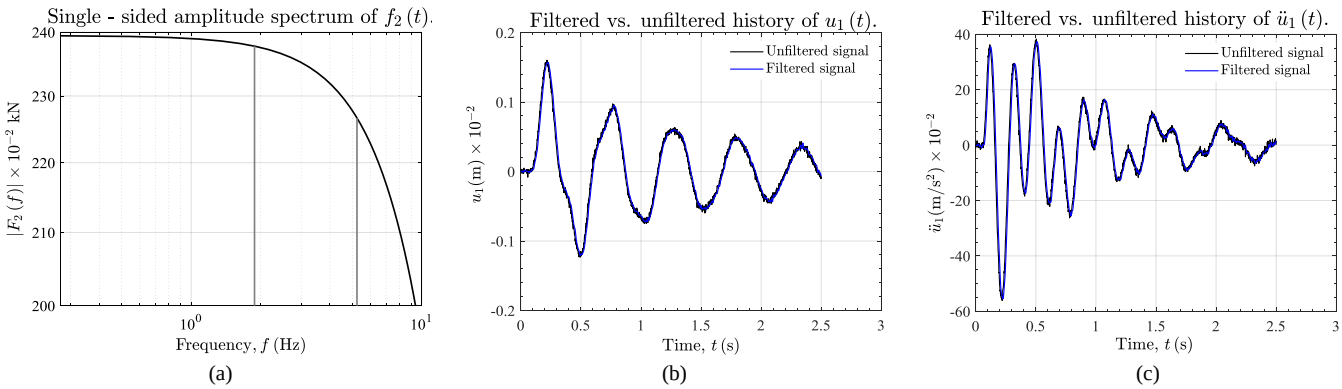
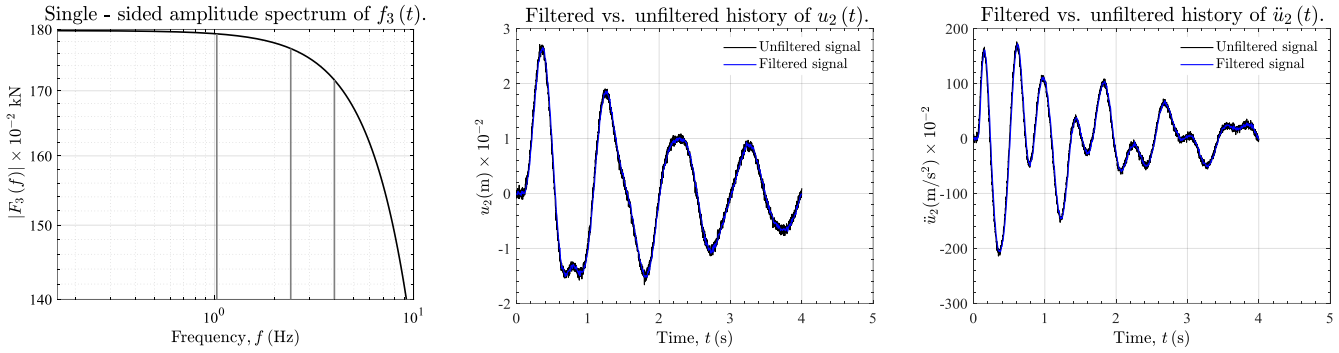
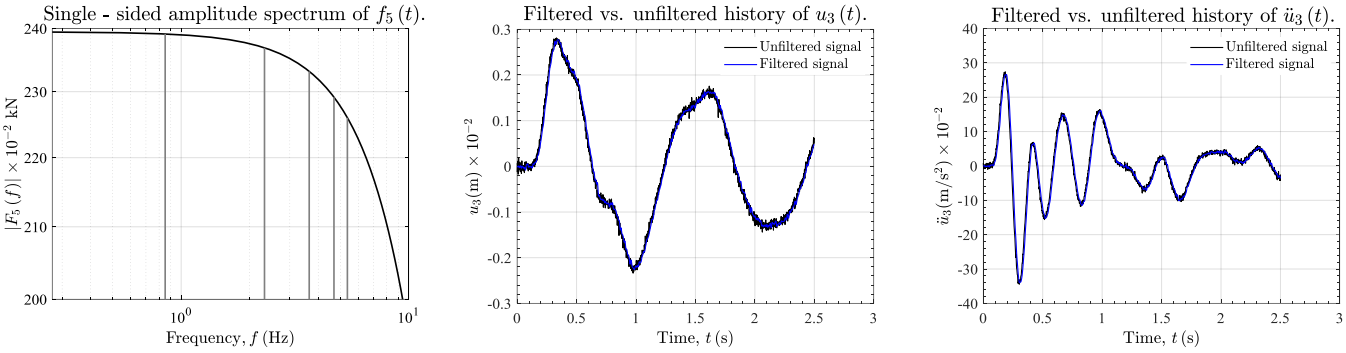


Figure 3. (a) Single-sided amplitude spectrum of the excitation; (b) displacement; and (c) acceleration histories for a noise level of 5% (filtered and unfiltered signals) for the 2-story model



(a) (b) (c)

Figure 4. (a) Single-sided amplitude spectrum of the excitation; (b) displacement; and (c) acceleration histories for a noise level of 5% (filtered and unfiltered signals) for the 3-story model



(a) (b) (c)

Figure 5. (a) Single-sided amplitude spectrum of the excitation; (b) displacement; and (c) acceleration histories for a noise level of 5% (filtered and unfiltered signals) for the 5-story model

An important aspect of this study is the evaluation of the metaheuristic methods' performance both under noisy conditions and after applying a filtering process. Additionally, the study examines how filtering the "measured" signals affects the algorithms' performance. Although displacement, velocity, and acceleration histories are typically filtered in practice, the algorithms' performance is first analyzed without filtering and subsequently with filtering. A low-pass Butterworth filter is employed, and its transfer function is defined as follows:

$$|H(j\omega)| = \frac{1}{\sqrt{1+\left(\frac{\omega}{\omega_c}\right)^{2\tilde{N}}}} \tag{22}$$

where $\tilde{N}$ is the filter order, ω_c is the cutoff frequency, and ω is the angular frequency. In this study, a sixth-order Butterworth low-pass filter with a normalized cutoff frequency defined as $\omega_c = 0.25\omega_{max}$ was used. The quantity ω_{max} corresponds to the maximum angular frequency of the excitation signal, which in this study is approximately 1745.33 rad/s (equivalent to 278 Hz). The resulting cutoff frequency is therefore $\omega_c = 436.33$ rad/s. This value was selected based on preliminary analyses to attenuate high-frequency noise without distorting the dominant low-frequency components.

A sixth-order configuration was selected to ensure a steeper roll-off beyond the cutoff, enabling effective suppression of high-frequency components without compromising the fidelity of the low-frequency signal content. While lower-order filters offer computational advantages, they exhibit more gradual attenuation and are therefore less effective at isolating the response from residual noise. The chosen order represents a balanced trade-off between frequency selectivity and signal fidelity. Alternative filters such as Chebyshev or elliptic designs were not considered due to their non-monotonic magnitude response in the passband or stopband, which may introduce undesirable amplitude distortions that alter modal characteristics relevant to structural identification.

Figure 6 illustrates the frequency responses of second-, fourth-, and sixth-order Butterworth filters, with vertical lines indicating the minimum and maximum frequencies of the studied models. This figure demonstrates how each filter attenuates specific frequency ranges, highlighting their effectiveness in reducing high-frequency noise.

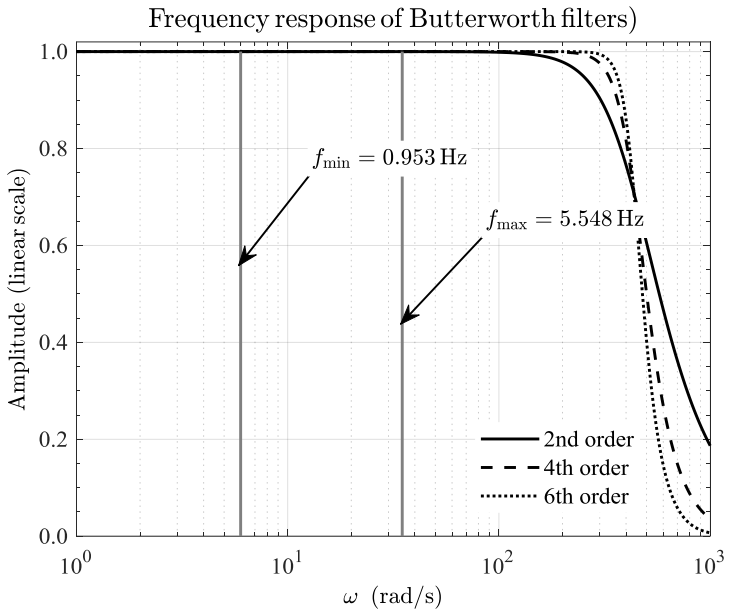


Figure 6. Frequency responses of Butterworth filters (2nd-, 4th-, and 6th-order)

The Butterworth filter was selected due to its flat frequency response in the passband, which preserves low-frequency modal content with minimal amplitude distortion. Alternative filters such as Chebyshev or band-pass designs could compromise the integrity of the signals of interest. For instance, Chebyshev filters introduce ripples in the passband, potentially distorting low-frequency characteristics. Band-pass filters, in turn, are unsuitable when the primary goal is to eliminate high-frequencies content while preserving the full low-response. It should be noted that the Butterworth filter does not eliminate all noise, but it selectively attenuates frequencies above the cutoff threshold, effectively reducing high-frequency white noise without completely removing it.

6. Results and Discussion

This section analyzes and discusses the performance of the three metaheuristic algorithms used to identify the lateral interstory stiffnesses and the modal damping ratio corresponding to the first two modes in two-dimensional shear-building models. The analysis is conducted based on the number of stories of the building and the level of noise in the recorded signals, considering two scenarios: one with the unfiltered response signals, and another using a sixth-order Butterworth filter. Six levels of white noise (0%, 1%, 2%, 3%, 4%, and 5% of the RMS value) were added to the response signals to evaluate the robustness of the algorithms.

6.1. Noise-Free Performance of the Algorithms

According to the results shown in Tables 1 and 2, corresponding to the scenario without noise, the three algorithms demonstrated high accuracy in the identification of structural parameters, with small differences among them. Table 1 presents the error in the estimation of the modal damping ratio, as well as the individual errors associated with each lateral interstory stiffness. Table 2, in turn, reports the same damping error (since a single value is identified per model) and the average of the lateral interstory stiffness errors previously reported in Table 1. The DE and PSO algorithms stood out as the most accurate, with an average error of 0.00% for both the modal damping ratio and the lateral interstory stiffnesses, and for the three building models analyzed.

With regard to GA, the estimation of the modal damping ratio and the average error in the lateral interstory stiffnesses for the 2-story building showed an error of 0.00% and 0.01%, respectively, while for the 3-story building, these errors reached 0.76% and 0.04%, respectively. Finally, for the 5-story building, these values were equal to 0.16% and 0.12%, respectively. This indicates that the selected metaheuristic method has little influence on the determination of the structural parameters when there is no noise in the measured signals.

Table 1. Values obtained from the identification process and errors in the estimation of lateral interstory stiffnesses and damping ratio under noise-free measurement conditions

Model	Actual parameter value	GA		DE		PSO	
		Identified value	Percentage error	Identified value	Percentage error	Identified value	Percentage error
2-Story building	ζ = 0.05	0.0500	0.00%	0.0500	0.00%	0.0500	0.00%
	k_1 = 30700.0 kN	30704.31	0.01%	30700.00	0.00%	30700.00	0.00%
	k_2 = 44300.0 kN	44300.28	0.00%	44300.00	0.00%	44300.00	0.00%
3-Story building	ζ = 0.05	0.0496	0.76%	0.0500	0.00%	0.0500	0.00%
	k_1 = 3000.0 kN	2997.79	0.07%	3000.00	0.00%	3000.00	0.00%
	k_2 = 2000.0 kN	1999.50	0.02%	2000.00	0.00%	2000.00	0.00%
	k_3 = 1000.0 kN	1000.12	0.01%	1000.00	0.00%	1000.00	0.00%
5-Story building	ζ = 0.05	0.0501	0.16%	0.0500	0.00%	0.0500	0.00%
	k_1 = 44300 kN	44290.62	0.02%	44300.00	0.00%	44300.00	0.00%
	k_2 = 44300 kN	44319.23	0.04%	44300.00	0.00%	44300.00	0.00%
	k_3 = 35000 kN	35032.10	0.09%	35000.00	0.00%	35000.00	0.00%
	k_4 = 35000 kN	35055.93	0.16%	35000.00	0.00%	35000.00	0.00%
	k_5 = 30700 kN	30785.72	0.28%	30700.00	0.00%	30700.00	0.00%

Table 2. Average errors in the identification of parameters (lateral interstory stiffness and modal damping) under noise-free conditions

Algorithm	2-Story building		3-Story building		5-Story building	
	ζ_n	Lateral stiffness	ζ_n	Lateral stiffness	ζ_n	Lateral stiffness
GA	0.00%	0.01%	0.76%	0.04%	0.16%	0.12%
DE	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
PSO	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%

6.2. Performance of Algorithms in the Presence of Noise in the Response Signals (Without Filtering)

As expected, the introduction of noise into the calculated displacement, velocity, and acceleration histories resulted in a decrease in the accuracy of parameter identification. This is evidenced by analyzing the results shown in Table 3. The average error increased with the noise level and with the complexity of the building model analyzed. Regarding the GA, there is an average error of 0.253% for a noise level of 1%, and it reaches a value of 2.806% for a noise level of 5% for the 2-story model. This error grows to 0.597% and 3.705% for 1% and 5% noise levels, respectively, for the 3-story model. When the model complexity increases, the errors grow to 0.717% and 7.882% for noise levels of 1% and 5%, respectively.

Table 3. Average errors in the identification of parameters under noisy conditions (unfiltered signals)

Algorithm	Model height	Noise level				
		1.00%	2.00%	3.00%	4.00%	5.00%
GA	2-story	0.253	0.701	1.056	1.694	2.806
	3-story	0.597	0.705	2.012	2.546	3.705
	5-story	0.717	3.131	5.962	6.296	7.882
DE	2-story	0.326	0.467	0.946	1.378	1.918
	3-story	0.181	0.688	1.504	2.867	4.080
	5-story	0.887	3.357	6.439	6.354	7.060
PSO	2-story	0.177	0.615	1.113	1.405	2.223
	3-story	0.171	0.702	1.412	2.561	4.190
	5-story	0.861	3.256	6.440	6.475	7.217

As can be seen in Table 3, in the case of DE, the errors obtained are similar in magnitude to those obtained with GA in the 2-story building; thus, for the 2-story model, the error in the estimation of the parameters is 0.326% for a noise level of 1% and 1.918% for a noise level of 5%. For the 3-story model, the error is 0.181% for a noise level of 1% and 4.080% for a noise level of 5%. While in the 5-story model, these errors are 0.887% and 7.060% for 1% and 5% noise,

respectively. Regarding the PSO algorithm, it presents average errors for the 2-story building of 0.177% and 2.223% for the 1% and 5% noise levels, respectively; values that increase to 0.171%–4.190% and 0.861%–7.217% for the 3- and 5-story models, respectively.

For the models with 2 and 3 stories, the choice of the metaheuristic method (GA, DE, and PSO) has a relatively limited influence on the accuracy of the identification of the structural parameters. In the scenario with the maximum noise level, errors smaller than 2.9% are observed for the 2-story model and values smaller than 4.2% for the 3-story model, regardless of the algorithm used.

Regarding the 5-story model, it is observed that the average errors obtained at the maximum noise level increase considerably compared to the simpler models. In this case, the average errors for GA, DE, and PSO are 7.882%, 7.060%, and 7.217%, respectively, which represents a significant increase in relation to those of the 2- and 3-story models. Although the errors obtained for the 5-story model are relatively high, the maximum difference among the metaheuristic algorithms analyzed is less than 1%, indicating that the three algorithms exhibit similar performance in this complex scenario.

Figure 7 illustrates the variation of the Mean Absolute Percentage Error (MAPE) as a function of the noise level in unfiltered response signals (Table 3).

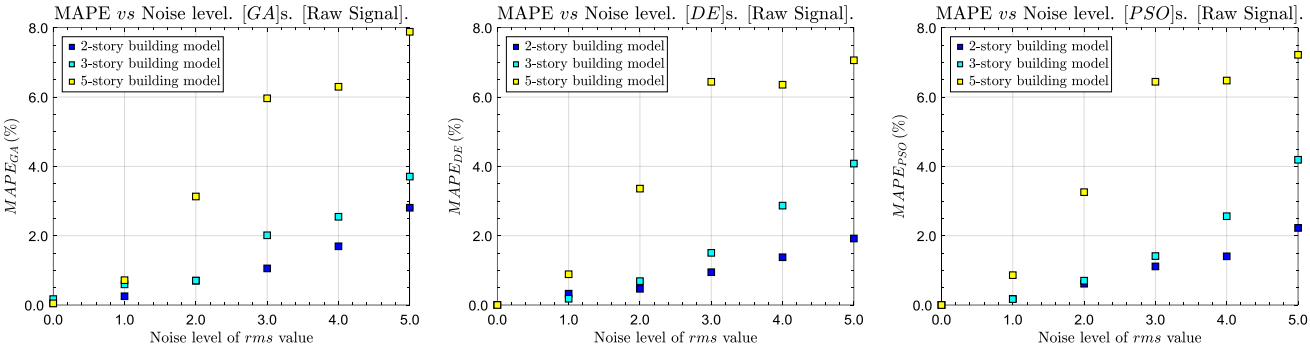


Figure 7. Variation of the Mean Absolute Percentage Error (MAPE) as a function of noise level in the unfiltered response signals

6.3. Performance of Algorithms in the Presence of Noise in the Response Signals (With Filtering)

The application of the sixth-order Butterworth filter to noise-contaminated signals has the effect of improving the accuracy of structural parameter identification. This improvement is particularly evident in scenarios with elevated noise levels and in 5-story building models. As shown in Table 4, under the most unfavorable noise level, the GA metaheuristic method accurately determines the damping ratios of the first two vibration modes, with an error of no greater than 1.74% for the 2-story model, 1.18% for the 3-story model, and 2.28% for the 5-story model.

Table 4. Values obtained from the identification process and errors in the estimation of lateral interstory stiffnesses and damping ratio under noisy measurement conditions. Metaheuristic Algorithm: GA

Model	Actual parameter value	Noise level				
		1.00%	2.00%	3.00%	4.00%	5.00%
2-Story building	ζ = 0.05	0.0504	0.0497	0.0498	0.0496	0.0491
		(0.77%)	(0.68%)	(0.34%)	(0.87%)	(1.74%)
	k_1 = 30700.0 kN	30711.36	30697.14	30593.25	30685.84	30461.91
		(0.04%)	(0.01%)	(0.35%)	(0.05%)	(0.78%)
	k_2 = 44300.0 kN	44265.92	44193.44	44139.52	43881.74	43814.46
		(0.08%)	(0.24%)	(0.36%)	(0.94%)	(1.10%)
3-Story building	ζ = 0.05	0.0495	0.0505	0.0494	0.0502	0.0504
		(1.00%)	(0.99%)	(1.18%)	(0.39%)	(0.89%)
	k_1 = 3000.0 kN	2998.24	2989.74	2984.47	2982.42	2980.85
		(0.06%)	(0.34%)	(0.52%)	(0.59%)	(0.64%)
	k_2 = 2000.0 kN	1997.53	1990.50	1990.06	1984.60	1980.57
		(0.12%)	(0.48%)	(0.50%)	(0.77%)	(0.97%)
	k_3 = 1000.0 kN	999.12	995.90	996.30	992.38	991.57
		(0.09%)	(0.41%)	(0.37%)	(0.76%)	(0.84%)

5-Story building	ζ = 0.05	0.0504	0.0496	0.0496	0.0505	0.0511
		(0.70%)	(0.85%)	(0.78%)	(1.07%)	(2.28%)
	k_1 = 44300 kN	44206.42	44128.18	43876.60	43708.90	43480.18
		(0.21%)	(0.39%)	(0.96%)	(1.33%)	(1.85%)
	k_2 = 44300 kN	44219.16	44053.68	43836.94	43422.77	43116.38
		(0.18%)	(0.56%)	(1.05%)	(1.98%)	(2.67%)
	k_3 = 35000 kN	34859.26	34738.72	34332.58	33764.65	33207.13
		(0.40%)	(0.75%)	(1.91%)	(3.53%)	(5.12%)
	k_4 = 35000 kN	34821.58	34566.60	34071.08	33287.77	32579.29
		(0.51%)	(1.24%)	(2.65%)	(4.89%)	(6.92%)
	k_5 = 30700 kN	30494.31	30219.06	29474.19	28228.22	27196.56
		(0.67%)	(1.57%)	(3.99%)	(8.05%)	(11.41%)

With respect to the estimation of lateral interstory stiffnesses, the GA algorithm exhibits very low errors for the 2- and 3-story buildings, with maximum deviations of 1.10% and 0.97%, respectively. For the 5-story model, although the errors increase due to the system's complexity and higher sensitivity to noise, the maximum recorded deviation is 11.41%, specifically in the identification of the fifth interstory stiffness.

The performance of the DE algorithm, presented in Table 5, also confirms the benefits of filtering. For the most unfavorable case (5% noise), damping ratio errors are limited to 1.79% for the 2-story model, 0.47% for the 3-story model, and 2.15% for the 5-story model. These values are comparable to those obtained with GA, and the error trends are similarly stable across noise levels.

In terms of lateral interstory stiffness identification, DE achieves high accuracy for 2- and 3-story models, with maximum errors of 1.19% and 1.22%, respectively. In the 5-story model, although the complexity leads to higher deviations, the maximum error reaches 11.39%, which is slightly lower than the maximum error observed with GA. Table 6 shows the results obtained with the PSO algorithm. For the 5% noise case, the damping ratio errors are 1.90% for the 2-story model, 0.91% for the 3-story model, and 1.89% for the 5-story model—figures that are comparable to those from GA and DE, confirming the robustness of the filtering technique across metaheuristics.

Table 5. Values obtained from the identification process and errors in the estimation of lateral interstory stiffnesses and damping ratio under noisy measurement conditions. Metaheuristic Algorithm: DE

Model	Actual parameter value	Noise level				
		1.00%	**2.00%**	**3.00%**	**4.00%**	**5.00%**
2-Story building	ζ = 0.05	0.0503	0.0502	0.0504	0.0491	0.0509
		(0.55%)	(0.42%)	(0.84%)	(1.78%)	(1.79%)
	k_1 = 30700.0 kN	30694.14	30695.10	30636.18	30554.45	30644.19
		(0.02%)	(0.02%)	(0.21%)	(0.47%)	(0.18%)
	k_2 = 44300.0 kN	44238.86	44299.06	44120.31	43807.32	43772.91
		(0.14%)	(0.00%)	(0.41%)	(1.11%)	(1.19%)
3-Story building	ζ = 0.05	0.0501	0.0501	0.0502	0.0502	0.0500
		(0.11%)	(0.18%)	(0.32%)	(0.47%)	(0.08%)
	k_1 = 3000.0 kN	2998.54	2995.38	2987.30	2980.29	2968.39
		(0.05%)	(0.15%)	(0.42%)	(0.66%)	(1.05%)
	k_2 = 2000.0 kN	1999.44	1996.32	1990.47	1985.94	1975.68
		(0.03%)	(0.18%)	(0.48%)	(0.70%)	(1.22%)
	k_3 = 1000.0 kN	999.32	998.14	994.63	993.50	990.17
		(0.07%)	(0.19%)	(0.54%)	(0.65%)	(0.98%)
5-Story building	ζ = 0.05	0.0500	0.0507	0.0506	0.0506	0.0511
		(0.03%)	(1.43%)	(1.15%)	(1.13%)	(2.15%)
	k_1 = 44300 kN	44267.50	44110.87	43921.21	43657.25	43419.03
		(0.07%)	(0.43%)	(0.86%)	(1.45%)	(1.99%)
	k_2 = 44300 kN	44246.36	44085.27	43782.52	43408.83	43004.99
		(0.12%)	(0.48%)	(1.17%)	(2.01%)	(2.92%)
	k_3 = 35000 kN	34920.61	34697.48	34323.11	33839.22	33349.02
		(0.23%)	(0.86%)	(1.93%)	(3.32%)	(4.72%)
	k_4 = 35000 kN	34898.71	34560.66	33934.97	33400.76	32383.48
		(0.29%)	(1.26%)	(3.04%)	(4.57%)	(7.48%)
	k_5 = 30700 kN	30553.99	30064.40	29321.45	28262.04	27204.77
		(0.48%)	(2.07%)	(4.49%)	(7.94%)	(11.39%)

In the estimation of lateral interstory stiffnesses, PSO also demonstrates competitive performance. Maximum errors recorded are 0.97% for the 2-story model, 1.23% for the 3-story model, and 11.09% for the 5-story model, the latter again corresponding to the most complex stiffness component (k_5).

In summary, the use of a sixth-order Butterworth filter significantly improves the accuracy of structural parameter identification under noisy conditions. This improvement becomes particularly relevant when identifying parameters in more complex building models. For the 2-story model, damping ratio errors across all algorithms and noise levels range from 0.34% to 1.90%, resulting in a maximum variation of 1.56%. Regarding lateral interstory stiffnesses, the errors range from 0.01% to 1.19%, yielding a total spread of 1.17%. These narrow ranges confirm that the filter effectively stabilizes the identification process in relatively simple systems, allowing all three algorithms (GA, DE, and PSO) to achieve comparable levels of accuracy.

In the 3-story model, damping ratio estimation errors lie between 0.39% and 1.18%, producing a total variation of 0.79%. For lateral interstory stiffnesses, the identification errors span from 0.03% to 1.23%, with a maximum variation of 1.20%. Despite the increased complexity compared to the 2-story model, the filter maintains strong error control, and no algorithm shows a clear advantage over the others in this scenario.

In the 5-story model, two distinct behaviors are observed regarding the identification of structural parameters. For the damping ratio, all three algorithms maintain consistent accuracy across noise levels, with errors ranging from 0.42% to 2.28%, showing no significant algorithmic advantage. This indicates that the Butterworth filter is effective at preserving low-frequency modal components critical to damping estimation, as evidenced by the consistently low errors across all noise levels. In contrast, for the lateral interstory stiffnesses, identification accuracy degrades progressively towards the upper stories. For stories 1 through 4, errors remain within the range of 0.07% to 7.48%, but in story 5, the deviations increase notably, reaching 11.09% to 11.41%, depending on the algorithm. This trend may reflect the intrinsic challenges of capturing the structural response at the top level, possibly due to reduced excitation and modeling sensitivity at higher stories.

Table 6. Values obtained from the identification process and errors in the estimation of lateral interstory stiffnesses and damping ratio under noisy measurement conditions. Metaheuristic Algorithm: PSO

Model	Actual parameter value	Noise level				
		1.00%	**2.00%**	**3.00%**	**4.00%**	**5.00%**
2-Story building	ζ = 0.05	0.0503	0.0494	0.0496	0.0497	0.0490
		(0.66%)	(1.12%)	(0.72%)	(0.58%)	(1.90%)
	k_1 = 30700.0 kN	30688.65	30678.01	30668.97	30596.66	30663.15
		(0.04%)	(0.07%)	(0.10%)	(0.34%)	(0.12%)
	k_2 = 44300.0 kN	44268.27	44113.58	44105.76	44224.01	43868.10
		(0.07%)	(0.42%)	(0.44%)	(0.17%)	(0.97%)
3-Story building	ζ = 0.05	0.0500	0.0500	0.0501	0.0505	0.0495
		(0.01%)	(0.09%)	(0.14%)	(0.94%)	(0.91%)
	k_1 = 3000.0 kN	2998.72	2995.21	2990.43	2980.44	2969.52
		(0.04%)	(0.16%)	(0.32%)	(0.65%)	(1.02%)
	k_2 = 2000.0 kN	1998.98	1997.14	1992.53	1986.75	1975.47
		(0.05%)	(0.14%)	(0.37%)	(0.66%)	(1.23%)
	k_3 = 1000.0 kN	999.73	998.80	995.59	994.15	988.81
		(0.03%)	(0.12%)	(0.44%)	(0.59%)	(1.12%)
5-Story building	ζ = 0.05	0.0500	0.0502	0.0510	0.0505	0.0509
		(0.10%)	(0.45%)	(1.93%)	(1.00%)	(1.89%)
	k_1 = 44300 kN	44267.19	44134.59	44011.75	43855.79	43375.69
		(0.07%)	(0.37%)	(0.65%)	(1.00%)	(2.09%)
	k_2 = 44300 kN	44254.63	44095.58	43845.88	43508.62	43099.85
		(0.10%)	(0.46%)	(1.03%)	(1.79%)	(2.71%)
	k_3 = 35000 kN	34903.84	34739.72	34326.39	33876.92	33241.37
		(0.27%)	(0.74%)	(1.92%)	(3.21%)	(5.02%)
	k_4 = 35000 kN	34895.90	34547.52	34047.35	33514.87	32405.77
		(0.30%)	(1.29%)	(2.72%)	(4.24%)	(7.41%)
	k_5 = 30700 kN	30502.33	30026.85	29341.05	28275.28	27296.69
		(0.64%)	(2.19%)	(4.43%)	(7.90%)	(11.09%)

Despite this localized discrepancy, all three metaheuristic algorithms maintain a globally consistent performance pattern, particularly in the damping ratio estimation, where differences remain bounded and no method shows a significant advantage once the filtering is applied. These results demonstrate that applying a sixth-order Butterworth filter is a highly effective strategy for improving robustness in parameter identification under noise. Furthermore, since none of the metaheuristic algorithms significantly outperforms the others under filtered conditions, their shared ability to deliver stable and accurate results becomes evident, especially in structures of moderate complexity.

Table 7 reports the average parameter identification errors based on filtered displacement, velocity, and acceleration data. Across all noise levels analyzed, the following trends are observed:

2-story model: The DE algorithm achieves the best performance, with a maximum average error of 1.122%. The PSO algorithm follows closely, reaching a maximum of 0.999%, while the GA algorithm shows the least favorable performance, with 1.203%.

3-story model: Both DE and GA algorithms yield the most accurate results, with maximum errors of 0.834% and 0.835%, respectively. The PSO algorithm, in contrast, presents a higher maximum error of 1.067%. Thus, while all three algorithms maintain acceptable performance, DE and GA outperform PSO in this particular case.

5-story model: The PSO algorithm again demonstrates a slight advantage, with a maximum average error of 5.035%, closely followed by GA with 5.042%, and DE with 5.106%. The differences among the three methods are minimal, despite the increased complexity of the system.

Across all cases, the GA, DE, and PSO algorithms exhibit broadly comparable performance in parameter identification under all noise levels analyzed. As expected, the average errors increase with both noise intensity and structural complexity, particularly for the 5-story model.

Table 7. Average errors in the identification of parameters under noisy conditions (filtered response signals)

Algorithm	Model height	Noise level				
		1.00%	2.00%	3.00%	4.00%	5.00%
GA	2-Story	0.294	0.310	0.351	0.621	1.203
	3-Story	0.318	0.554	0.641	0.626	0.835
	5-Story	0.446	0.890	1.890	3.476	5.042
DE	2-Story	0.236	0.145	0.483	1.122	1.054
	3-Story	0.065	0.175	0.440	0.619	0.834
	5-Story	0.202	1.089	2.107	3.403	5.106
PSO	2-Story	0.256	0.538	0.419	0.363	0.999
	3-Story	0.033	0.128	0.319	0.711	1.067
	5-Story	0.248	0.918	2.113	3.190	5.035

Therefore, algorithm selection may be guided by practical considerations, such as ease of implementation, tuning simplicity, or computational cost, rather than accuracy alone. In summary, the combination of signal filtering and metaheuristic optimization yields reliable structural parameter identification in noisy shear building models.

Overall, the GA, DE, and PSO algorithms demonstrate similar performance in structural parameter identification under all noise levels, with average errors increasing with both noise intensity and system complexity. These results confirm that, when combined with signal filtering, GA, DE, and PSO are effective tools for identifying modal damping and lateral interstory stiffness in two-dimensional shear building models.

Figure 8 illustrates the variation in the Mean Absolute Percentage Error (MAPE) as a function of noise level in the filtered response signals (Tables 1 and 7). At the highest noise level, the maximum average errors remain below 5.106% for the 5-story model. In contrast, in the 2- and 3-story models, errors are below 1.203%. These results suggest that the metaheuristic algorithms analyzed (GA, DE, and PSO), combined with signal filtering, are suitable for the identification of structural parameters, particularly modal damping and lateral interstory stiffness, in two-dimensional shear building models.

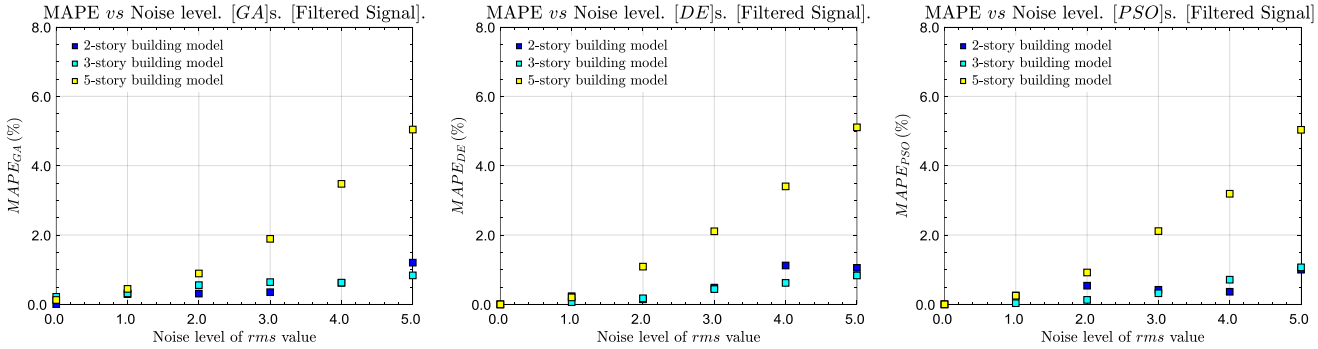


Figure 8. Variation of the Mean Absolute Percentage Error (MAPE) as a function of noise level in the filtered response signals

6.4. Comparative Evaluation with the Element-Level Identification Method by Wang & Haldar [1]

To benchmark the performance of the metaheuristic algorithms evaluated in this study, a comparative analysis was carried out against the time-domain identification methodology introduced by Wang & Haldar [1]. Both approaches estimate structural parameters solely from measured response data, without requiring knowledge of the input excitation. While their method identifies interstory stiffnesses and viscous damping coefficients for each story, our approach estimates the lateral interstory stiffnesses and the modal damping ratios associated with the first two modes of vibration. The main difference between the two methods lies in the optimization strategy. Wang & Haldar update parameter estimates iteratively by solving a sequence of least-squares problems based on measured responses and assumed zero input at selected time instants. In contrast, our approach formulates the identification as a nonlinear optimization problem, solved using metaheuristic algorithms (GA, DE, and PSO), which explore the solution space stochastically and do not rely on gradient-based updates.

Although the present study estimates modal damping ratios instead of story-specific viscous damping coefficients, a valid comparison can still be made by analyzing the average relative errors in damping identification. For lateral interstory stiffnesses, the comparison is performed on a story-by-story basis. This is straightforward in the case of the 2- and 3-story models, as both studies use identical structural properties (masses and stiffnesses), thereby enabling a direct match. Note that the relative errors obtained using the metaheuristic-based approach (GA, DE, and PSO), and reported in Tables 8 to 10, were computed from the filtered response signals described in Section 6.3. For the 5-story building model specifically, both our model and that of Wang & Haldar have varying mass and stiffness properties, but these are not the same. Therefore, to ensure dimensional consistency, the comparison was limited to the first five stories of their six-story model. While this setting is not structurally equivalent, it is included to explore potential performance trends across identification methods as model complexity increases.

Table 8. Relative errors (%) in the estimation of structural parameters using different identification methods for a 2-story shear building model

Structural parameter	Noise-free				5% Noise level			
	Wang & Haldar [1]	GA	DE	PSO	Wang & Haldar [1]	GA*	DE*	PSO*
ζ/c_{prom}	0.08%	0.00%	0.00%	0.00%	7.48%	1.74%	1.79%	1.90%
k_1	0.00%	0.01%	0.00%	0.00%	0.05%	0.79%	0.18%	0.12%
k_2	0.01%	0.00%	0.00%	0.00%	1.81%	1.10%	1.19%	0.97%

∗ These cases correspond to scenarios in which response signals were pre-filtered using a Butterworth filter prior to the structural parameter identification process. Estimations were conducted on two-dimensional shear building models subjected to dynamic excitation.

Table 9. Relative errors (%) in the estimation of structural parameters using different identification methods for a 3-story shear building model

Structural parameter	Noise-free				1% Noise level			
	Wang & Haldar [1]	GA	DE	PSO	Wang & Haldar [1]	GA*	DE*	PSO*
ζ/c_{prom}	0.47%	0.76%	0.00%	0.00%	3.28%	1.00%	0.11%	0.01%
k_1	0.08%	0.07%	0.00%	0.00%	0.38%	0.06%	0.05%	0.04%
k_2	0.05%	0.02%	0.00%	0.00%	1.75%	0.12%	0.03%	0.05%
k_3	0.01%	0.01%	0.00%	0.00%	0.16%	0.09%	0.07%	0.03%

∗ These cases correspond to scenarios in which response signals were pre-filtered using a Butterworth filter prior to the structural parameter identification process. Estimations were conducted on two-dimensional shear building models subjected to dynamic excitation.

Table 10. Relative errors (%) in the estimation of structural parameters using different identification methods for a 5-story shear building model

Structural parameter	Noise-free				5% Noise level			
	Wang & Haldar [1]	GA	DE	PSO	Wang & Haldar [1]	GA*	DE*	PSO*
ζ/c_{prom}	0.02%	0.16%	0.00%	0.00%	1.55%	2.28%	2.15%	1.89%
k_1	0.01%	0.02%	0.00%	0.00%	0.64%	1.85%	1.99%	2.09%
k_2	0.01%	0.04%	0.00%	0.00%	1.23%	2.67%	2.92%	2.71%
k_3	0.01%	0.09%	0.00%	0.00%	0.92%	5.12%	4.72%	5.02%
k_4	0.01%	0.16%	0.00%	0.00%	0.67%	6.92%	7.48%	7.41%
k_5	0.01%	0.28%	0.00%	0.00%	0.58%	11.41%	11.39%	11.09%

∗ These cases correspond to scenarios in which response signals were pre-filtered using a Butterworth filter prior to the structural parameter identification process. Estimations were conducted on two-dimensional shear building models subjected to dynamic excitation

Tables 8 to 10 provide a comparative summary of the relative errors (%) in the estimation of structural parameters using the two identification methodologies—Wang & Haldar's approach and the metaheuristic-based framework—for 2-, 3-, and 5-story shear building models.

In the case of the 2-story shear building model (Table 8), all methods exhibit very high accuracy under noise-free conditions. The DE and PSO algorithms achieve exact matches for all parameters, while GA presents a minor error of 0.01% in k_1. Wang & Haldar's method shows similarly low errors, with a maximum of 0.08% in damping. These small differences confirm that under ideal conditions, the four methods deliver highly accurate and practically equivalent results. Under 5% noise in the signal responses, however, differences in robustness emerge. Wang & Haldar's method shows a notable increase in damping error (7.48%), while the metaheuristic algorithms keep this error below 2%, specifically, 1.74% (GA), 1.79% (DE), and 1.90% (PSO). For stiffness identification, all methods remain under 2% error.

In the 3-story shear building model (Table 9), all methods once again demonstrate excellent performance under noise-free conditions. The DE and PSO algorithms produce exact estimates for all parameters, while GA and Wang & Haldar's method show slightly higher—but still very low—errors, particularly in damping and the first two stiffness coefficients. When 1% noise level is introduced into the response signals, the superior robustness of the metaheuristic framework becomes more evident. Damping estimation errors increase noticeably for Wang & Haldar (3.28%), while the metaheuristic algorithms maintain significantly lower errors: 1.00% (GA), 0.11% (DE), and 0.01% (PSO). Similarly, for stiffness parameters, all metaheuristic methods remain below 0.12% error across all stories, whereas Wang & Haldar's method reaches up to 1.75% in k_2.

For the 5-story shear building model (Table 10), all methods perform well under noise-free conditions. The DE and PSO algorithms produce exact estimates for all parameters, while GA shows small but increasing errors in the upper stories, reaching up to 0.16% in k_4 and 0.28% in k_5. In contrast, Wang & Haldar's method maintains uniformly low errors (0.01%) across all stiffness parameters and 0.02% in damping. Under 5% noise, differences in performance become more evident. Damping identification errors rise to 1.55% for Wang & Haldar, while metaheuristic methods yield values ranging from 1.89% (PSO) to 2.28% (GA). Regarding stiffness estimation, Wang & Haldar's errors remain below 1.3% across all stories, with a maximum of 1.23% in k_2. In contrast, the metaheuristic methods show a progressive increase in error with story height. Errors remain moderate in the lower stories (k_1, k_2), but rise sharply in the upper levels: for k_4, all metaheuristic algorithms exceed 6.9%, and for k_5, they surpass 11%, with GA reaching the highest error at 11.41%.

These results suggest that while both frameworks are capable of accurate identification under ideal conditions, the metaheuristic-based approach demonstrates greater robustness in damping identification under noise. In contrast, Wang & Haldar's method yields more consistent results in lateral interstory stiffness estimation—specifically in the 5-story model and particularly in the upper stories, where the metaheuristic algorithms show reduced accuracy under noisy conditions.

6.5. Computational Efficiency and Convergence Behavior of Metaheuristic Algorithms

This section presents a detailed evaluation of the computational efficiency and convergence behavior of the GA, DE, and PSO algorithms, measured through average CPU time and iteration count required to achieve convergence in each model. These results complement the accuracy analysis discussed in previous sections and provide a broader perspective on the practical efficiency of each metaheuristic algorithm.

Simulations were implemented in MATLAB R2023b and executed across five machines with identical lower-mid-range specifications [Intel® Core™ i7-6700T (4 cores/8 threads, 2.8 GHz), 16 GB DDR4 RAM, and a 250 GB solid-state drive (SSD)], all running Windows 11 as the operating system.

The simulation runs were evenly distributed among the five systems, and the values reported in Table 11 correspond to the average of 10 independent executions per model and algorithm. All metaheuristic algorithms (GA, DE, and PSO) were implemented by the authors in MATLAB using custom code. No proprietary or built-in metaheuristic functions from MATLAB toolboxes were used in this study.

Table 11. Average CPU Time and Iteration Count per Algorithm and Model

Algorithm	2-Story building		3-Story building		5-Story building	
	CPU Time (s)	Average Iterations	CPU Time (s)	Average Iterations	CPU Time (s)	Average Iterations
GA	1044.34 R0	2759.56 R0	2356.41 R0	3117.63 R0	843.83 R0	6441.65 R0
	1146.01 R1	2746.83 R1	1564.83 R1	1920.00 R1	987.23 R1	6680.56 R1
DE	145.56 R0	2719.28 R0	454.49 R0	4333.98 R0	2363.20 R0	9300.60 R0
	170.59 R1	2861.00 R1	485.82 R1	4049.52 R1	1283.69 R1	9804.54 R1
PSO	277.60 R0	3357.00 R0	2281.86 R0	4866.11 R0	2067.54 R0	10364.35 R0
	284.89 R1	3078.94 R1	2218.69 R1	4503.85 R1	977.62 R1	10336.81 R1

R0 refers to the case in which the system response was used without any filtering. R1 corresponds to the case in which the response was filtered prior to the identification process.

Table 11 summarizes the average CPU time and iteration count for the GA, DE, and PSO algorithms under both unfiltered (R0) and filtered (R1) response scenarios. These metrics not only provide a general measure of computational effort and cost but also offer complementary insight into the internal search dynamics of each algorithm. While CPU time quantifies the total computational effort, it does not necessarily reflect the algorithm's internal efficiency. For instance, an algorithm may require a large number of iterations with only marginal improvements per step, yet maintain a relatively low total CPU time if the cost per iteration is minimal, as is the case when the objective function is simple or the population size is small. Conversely, another algorithm may converge in fewer iterations but incur higher total CPU time due to significant per-evaluation costs, arising from a more complex objective function, a more intensive search mechanism, or a larger population size. For this reason, the number of iterations complements CPU time, since it provides insight into the algorithm's capacity to explore and exploit the search space independently of the total runtime, which is influenced by the number of iterations, the complexity of the objective function, and the algorithm-specific population or swarm size. A detailed analysis of these results is presented in the following subsections.

1. Performance Without Filtering (R0 Scenario)

In the unfiltered scenario (R0), all algorithms show an increase in average CPU time when transitioning from the 2-story to the 3-story model, albeit with markedly different growth rates. DE is the most efficient in this context, despite showing a 212.2% increase in time (from 145.56 s to 454.49 s). PSO follows in ranking but displays a much larger relative increase of 722.0% (from 277.60 s to 2281.86 s). GA, despite showing the smallest relative increase (125.6%), remained the slowest algorithm in absolute terms: 1044.34 s for the 2-story model and 2356.41 s for the 3-story model.

However, in the 5-story model, this trend reverses: GA becomes the most efficient algorithm, achieving an average time of 843.83 s, significantly outperforming the others. In comparison, PSO shows a 145.02% increase relative to GA, and DE exhibits the highest computational cost with a 180.1% increase.

Regarding the average number of iterations to convergence, DE requires 2719.28 iterations for the 2-story model, followed by GA with 2759.56 (+1.48%) and PSO with 3357.00 (+23.45%). For the 3-story model, GA requires 3117.63 iterations, while DE and PSO need 4333.98 (+39.02%) and 4866.11 (+56.08%), respectively. For the 5-story model, GA converges in 6441.65 iterations, compared to 9300.60 for DE (+44.38%) and 10364.35 for PSO (+60.90%).

These results suggest that GA demonstrates greater robustness as problem dimensionality increases, as its performance deteriorates less than DE and PSO. However, this does not necessarily imply superior internal efficiency. This resilience may be attributed to GA's lower sensitivity to the combination of bounded search spaces and high dimensionality, whereas DE and PSO tend to stagnate or disperse in more complex domains.

In simpler models with wide parameter bounds, DE and PSO outperform GA. However, as the number of parameters increases, GA becomes clearly more competitive, surpassing the other two algorithms. This reinforces the notion that, in high-dimensional problems, designing a well-defined search space alone is insufficient to ensure efficiency. In such scenarios, DE and PSO require enhanced initial diversity and adaptive exploration strategies to avoid premature convergence, even when the search bounds are well configured. GA's capacity to maintain robust performance under demanding conditions makes it a particularly attractive tool for large-scale optimization problems.

2. Performance With Filtering (R1 Scenario)

In the 2-story model with filtered response, DE achieved the lowest average CPU time (170.59 s), outperforming PSO (284.89 s, +67.00%) and GA (1146.01 s, +571.79%). In the 3-story model, DE again had the best computational performance (485.82 s), followed by GA (1564.83 s, +222.10%) and PSO (2218.69 s, +356.69%). Finally, in the 5-story model, PSO emerged as the most efficient (977.62 s), slightly outperforming GA (987.23 s, +0.98%) and substantially outperforming DE (1283.69 s, +31.31%).

As for the number of iterations to convergence, in the 2-story model GA required 2746.83 iterations, followed by DE (2861.00, +4.16%) and PSO (3078.94, +12.09%). In the 3-story model, GA required only 1920.00 iterations, while DE and PSO needed 4049.52 (+110.91%) and 4503.85 (+134.58%), respectively. In the 5-story case, GA maintained its advantage with 6680.56 iterations, compared to 9804.54 for DE (+46.76%) and 10336.81 for PSO (+54.73%).

These results show that, under filtered conditions, GA consistently achieves the lowest average number of iterations to convergence. DE ranks second, while PSO systematically requires the highest iteration count. This hierarchy is preserved across all three models, suggesting a stable relative behavior of the algorithms as system complexity increases.

3. Impact of Filtering on Algorithmic Performance (R0 vs. R1)

When comparing R0 and R1 scenarios, it becomes evident that filtering the structural response significantly impacts CPU time, while the number of iterations shows minimal variation in most cases. In terms of CPU time, for the 2-story model, DE increased from 145.56 s (R0) to 170.59 s (R1), a 17.2% rise. PSO increased from 277.60 s to 284.89 s (+2.6%), and GA from 1044.34 s to 1146.01 s (+9.7%). For the 3-story model, DE increased from 2719.28 s to 2861.00

s (+5.2%), while PSO decreased from 3357.00 s to 3078.94 s (–8.3%) and GA dropped from 2356.41 s to 1564.83 s (–33.6%). In the 5-story model, DE showed a significant reduction from 2363.20 s to 1283.69 s (–45.7%), PSO from 2067.54 s to 977.62 s (–52.7%), whereas GA exhibited a slight increase from 843.83 s to 987.23 s (+17.0%).

These results indicate that DE and particularly PSO benefit from filtering, especially in the 5-story model. GA, on the other hand, exhibits a mixed response: improved performance in the intermediate case (3 stories), but slight degradation in the other two.

Regarding the average number of iterations, the impact of filtering is more heterogeneous. For the 2-story model, DE increased from 2719.28 to 2861.00 (+5.2%), while PSO decreased from 3357.00 to 3078.94 (–8.3%) and GA remained nearly unchanged, from 2759.56 to 2746.83 (–0.5%). In the 3-story model, all algorithms reduced their iteration count: DE from 4333.98 to 4049.52 (–6.6%), PSO from 4866.11 to 4503.85 (–7.4%), and GA from 3117.63 to 1920.00 (–38.4%). For the 5-story case, DE increased slightly from 9300.60 to 9804.54 (+5.4%), PSO remained stable (from 10364.35 to 10336.81, –0.3%), and GA increased marginally from 6441.65 to 6680.56 (+3.7%).

These findings suggest that, although filtering does not substantially alter the search trajectory or systematically reduce the number of iterations required for convergence, it effectively reduces the per-iteration computational cost. Therefore, its primary benefit lies not in improving the internal efficiency of the search process, but in making each step computationally cheaper—especially in scenarios involving either high-dimensional or computationally expensive objective functions. In this sense, filtering emerges as a complementary mechanism for enhancing overall algorithmic performance and is particularly advisable when evaluation cost dominates total computational effort, without altering the core algorithm.

7. Conclusions

This study compares the performance of three metaheuristic algorithms (DE, GA, and PSO) to identify the lateral interstory stiffnesses and the modal damping ratio of two-dimensional shear buildings. Three building models with 2, 3, and 5 stories were analyzed, subjected to an impulsive excitation at the upper floor. Free vibration responses (displacement histories, velocities, and accelerations at each of the analyzed building levels) were calculated. To evaluate the reliability of the algorithms, the signal responses were contaminated with white noise, considering six noise levels: 0%, 1%..., 5% of the RMS value. Regarding the responses obtained, two scenarios were studied: one with raw response signals and the other with filtered response signals. The results lead to the following conclusions:

1. Unfiltered response signals:

In the model without filtering, the three metaheuristic algorithms (GA, DE, and PSO) are capable of identifying the lateral interstory stiffnesses and the modal damping ratio of the first two modes with reasonable accuracy, especially in low-rise models and moderate noise conditions. The maximum average error in the estimation between actual and identified values was less than 7.882%, observed at the highest noise level and in the most complex model. The results for the identification of lateral interstory stiffness and modal damping ratio at a 5% noise level (RMS value) are summarized below:

- 2-story building: DE achieved the lowest error, with an average of 1.918%, PSO produced an intermediate value (2.223%), while GA had the least accurate results with an error of 2.806%.
- 3-story building: GA showed the lowest average error of 3.705%, followed by DE (4.080%), while PSO recorded the highest, with an average error of 4.190%.
- 5-story building: In this case, DE outperformed the other algorithms with the lowest average error recorded, at 7.060%, followed by PSO (7.217%), while GA showed the highest deviation (7.882%).

2. Filtered response signals:

Across all noise levels analyzed (with values of 1%, 2%, 3%, 4%, or 5%), the sixth-order Butterworth filter significantly improved the accuracy of structural parameter identification, especially at higher noise levels and for larger buildings. In the 5-story building, considering the maximum errors observed across these noise levels, filtering reduced the identification error by approximately 4% across all algorithms.

- 2-story building: DE performed best with an average error of 1.122%, followed closely by PSO with 0.999%, and GA with the highest error of 1.203%.
- 3-story building: DE and GA yielded nearly identical maximum errors of 0.834% and 0.835%, respectively, while PSO showed a slightly higher error of 1.067%.
- 5-story building: PSO demonstrated a marginal advantage with an error of 5.035%, followed by GA with 5.042%, and DE with 5.106%. Despite the increased complexity, all three algorithms maintained comparable performance.

These results confirm that the application of a Butterworth filter markedly enhances the robustness of metaheuristic algorithms under noisy conditions. In particular, the filter stabilizes the estimation of both damping ratios and lateral interstory stiffnesses, even in high-complexity models. Furthermore, the absence of a consistently superior algorithm reinforces the conclusion that GA, DE, and PSO all remain viable and reliable alternatives when combined with appropriate signal processing.

3. Computational performance:

The comparative evaluation of algorithmic performance under both unfiltered (R0) and filtered (R1) response scenarios revealed distinct strengths for each metaheuristic. Differential Evolution (DE) demonstrated the lowest computational cost in simple, low-dimensional problems, particularly under unfiltered conditions. Particle Swarm Optimization (PSO) showed substantial improvement from pre-filtering, especially in more complex models. However, the Genetic Algorithm (GA) stood out for its consistent convergence behavior and reduced sensitivity to problem dimensionality. GA achieved the lowest iteration counts in most cases, and maintained stable performance regardless of the filtering condition. These attributes make GA a robust and scalable option for large-scale structural identification tasks, where convergence stability and adaptability are critical.

In summary, this study shows that the three metaheuristic algorithms—GA, DE, and PSO—satisfactorily identify the structural parameters of two-dimensional shear buildings with minimal differences. The incorporation of a filtering process, such as the Butterworth filter, substantially enhances the accuracy of the identification process, especially under noisy conditions and in complex models. The reported outcomes correspond to the average values obtained from 10 independent runs for each identification scenario, thereby allowing us to capture the performance trends of the algorithms in response to noise and the complexity associated with the number of stories in the analyzed shear building models. Although these results are based on theoretical models, they demonstrate strong potential for application in real-world structural safety assessments, particularly in dynamic testing scenarios where signal noise is an inherent challenge.

8. Declarations

8.1. Author Contributions

Conceptualization, C.A.G.P., J.D.L.C., and J.V.G.; methodology, C.A.G.P.; validation, C.A.G.P., J.D.L.C., and J.V.G.; investigation, C.A.G.P. and J.D.L.C.; writing—original draft preparation, C.A.G.P., J.D.L.C., and J.V.G.; writing—review and editing, C.A.G.P., J.D.L.C., and J.V.G.; funding acquisition, C.A.G.P. All authors have read and agreed to the published version of the manuscript.

8.2. Data Availability Statement

The data presented in this study are available upon request from the corresponding author.

8.3. Funding

This work was supported by Universidad Panamericana under Grant No. FFI 2024 UP-CI-2024-GDL-13-ING.

8.4. Acknowledgements

The authors would like to thank Universidad Panamericana for its financial support, and the Universidad Autónoma del Estado de México for its academic collaboration

8.5. Conflicts of Interest

The authors declare no conflict of interest.

9. References

[1] Wang, D., & Haldar, A. (1994). Element- Level System Identification with Unknown Input. Journal of Engineering Mechanics, 120(1), 159–176. doi:10.1061/(asce)0733-9399(1994)120:1(159).

[2] Koh, C. G., See, L. M., & Balendra, T. (1995). Determination of storey stiffness of three-dimensional frame buildings. Engineering Structures, 17(3), 179–186. doi:10.1016/0141-0296(95)00055-C.

[3] Ling, X., & Haldar, A. (2004). Element Level System Identification with Unknown Input with Rayleigh Damping. Journal of Engineering Mechanics, 130(8), 877–885. doi:10.1061/(asce)0733-9399(2004)130:8(877).

[4] Yuen, K.-V., Au, S. K., & Beck, J. L. (2004). Two-Stage Structural Health Monitoring Approach for Phase I Benchmark Studies. Journal of Engineering Mechanics, 130(1), 16–33. doi:10.1061/(asce)0733-9399(2004)130:1(16).

[5] Katkhuda, H., Martinez, R., & Haldar, A. (2005). Health Assessment at Local Level with Unknown Input Excitation. Journal of Structural Engineering, 131(6), 956–965. doi:10.1061/(asce)0733-9445(2005)131:6(956).

[6] González-Pérez, C., & Valdés-González, J. (2011). Identification of structural damage in a vehicular bridge using artificial neural networks. Structural Health Monitoring, 10(1), 33–48. doi:10.1177/1475921710365416.

[7] Badaoui, M., Bourahla, N., & Bensaibi, M. (2019). Estimation of accidental eccentricities for multi-storey buildings using artificial neural networks. Asian Journal of Civil Engineering, 20(5), 703–711. doi:10.1007/s42107-019-00137-x.

[8] González-Pérez, C. A., & De-la-Colina, J. (2022). Determination of mass properties in floor slabs from the dynamic response using artificial neural networks. Civil Engineering Journal, 8(8). doi:10.28991/CEJ-2022-08-08-01.

[9] Houssein, E. H., Hossam Abdel Gafar, M., Fawzy, N., & Sayed, A. Y. (2025). Recent metaheuristic algorithms for solving some civil engineering optimization problems. Scientific Reports, 15(1), 7929. doi:10.1038/s41598-025-90000-8.

[10] Wang, G.S., Huang, F.K. & Lin, H.H. (2004) Application of Genetic Algorithm to Structural Dynamic Parameter Identification. 13th World Conference on Earthquake Engineering, 1-6 August , 2004, Vancouver, Canada.

[11] Casciati, S. (2008). Stiffness identification and damage localization via differential evolution algorithms. Structural Control and Health Monitoring, 15(3), 436–449. doi:10.1002/stc.236.

[12] Xue, S., Tang, H., & Zhou, J. (2009). Identification of structural systems using particle swarm optimization. Journal of Asian Architecture and Building Engineering, 8(2), 517–524. doi:10.3130/jaabe.8.517.

[13] Begambre, O., & Laier, J. E. (2009). A hybrid Particle Swarm Optimization - Simplex algorithm (PSOS) for structural damage identification. Advances in Engineering Software, 40(9), 883–891. doi:10.1016/j.advengsoft.2009.01.004.

[14] Xiang, J., & Liang, M. (2012). A two-step approach to multi-damage detection for plate structures. Engineering Fracture Mechanics, 91, 73–86. doi:10.1016/j.engfracmech.2012.04.028.

[15] Li, R., Mita, A., & Zhou, J. (2013). Symbolization-based differential evolution strategy for identification of structural parameters. Structural Control and Health Monitoring, 20(10), 1255–1270. doi:10.1002/stc.1530.

[16] Ravanfar, S. A., Abdul Razak, H., Ismail, Z., & Hakim, S. J. S. (2016). A Hybrid Wavelet Based–Approach and Genetic Algorithm to Detect Damage in Beam-Like Structures without Baseline Data. Experimental Mechanics, 56(8), 1411–1426. doi:10.1007/s11340-016-0181-y.

[17] Ghannadi, P., & Kourehli, S. S. (2019). Structural damage detection based on MAC flexibility and frequency using moth-flame algorithm. Structural Engineering and Mechanics, 70(6), 649–659. doi:10.12989/sem.2019.70.6.649.

[18] Ferreira-Gomes, G., & Alves-de-Almeida, F. (2020). Tuning metaheuristic algorithms using mixture design: Application of sunflower optimization for structural damage identification. Advances in Engineering Software, 149(2020), 102877. doi:10.1016/j.advengsoft.2020.102877.

[19] Miao, B., Wang, M., Yang, S., Luo, Y., & Yang, C. (2020). An Optimized Damage Identification Method of Beam Using Wavelet and Neural Network. Engineering, 12(10), 748–765. doi:10.4236/eng.2020.1210053.

[20] Guo, J., Guan, D., & Zhao, J. (2020). Structural Damage Identification Based on the Wavelet Transform and Improved Particle Swarm Optimization Algorithm. Advances in Civil Engineering, 2020(1), 1–19. doi:10.1155/2020/8869810.

[21] Minh, H. Le, Khatir, S., Abdel Wahab, M., & Cuong-Le, T. (2021). An Enhancing Particle Swarm Optimization Algorithm (EHVPSO) for damage identification in 3D transmission tower. Engineering Structures, 242(September), 112412. doi:10.1016/j.engstruct.2021.112412.

[22] Minh, H. Le, Sang-To, T., Khatir, S., Abdel Wahab, M., & Cuong-Le, T. (2023). Damage identification in high-rise concrete structures using a bio-inspired meta-heuristic optimization algorithm. Advances in Engineering Software, 176(2023), 103399. doi:10.1016/j.advengsoft.2022.103399.

[23] Zar, A., Hussain, Z., Akbar, M., Rabczuk, T., Lin, Z., Li, S., & Ahmed, B. (2024). Towards vibration-based damage detection of civil engineering structures: overview, challenges, and future prospects. International Journal of Mechanics and Materials in Design, 20(3), 591–662. doi:10.1007/s10999-023-09692-3.

[24] Li, Y. F., Minh, H. Le, Cao, M. Sen, Qian, X., & Abdel Wahab, M. (2024). An integrated surrogate model-driven and improved termite life cycle optimizer for damage identification in dams. Mechanical Systems and Signal Processing, 208(2024), 110986. doi:10.1016/j.ymssp.2023.110986.

[25] Hernández-González, I. A., & García-Macías, E. (2024). Towards a comprehensive damage identification of structures through populations of competing models. Engineering with Computers, 40(2024), 3157–3174. doi:10.1007/s00366-024-01972-6.

[26] Alemu, Y. L., Lahmer, T., & Walther, C. (2024). Damage Detection with Data-Driven Machine Learning Models on an Experimental Structure. Eng, 5(2), 629–656. doi:10.3390/eng5020036.

[27] Farhadi, M., Ghiasi, R., & Torkzadeh, P. (2024). Damage detection of truss structures using meta-heuristic algorithms and optimized group method of data handling surrogate model. Structures, 65(2024), 106736. doi:10.1016/j.istruc.2024.106736.

[28] Davis L. (1991). Hybridization and numerical representation. The Handbook of Genetic Algorithms Edited by Davis L., 61–71, Van Nostrand Reinhold, New York, United States.

[29] Janikow, C. Z. & Michalewicz, Z. (1991). An experimental comparison of binary and floating point representations in genetic algorithms. Proceedings of the 4th International Conference on Genetic Algorithms, 13-16 July, 1991, San Diego, United Sates.

[30] Wright, A. H. (1991). Genetic Algorithms for Real Parameter Optimization. Foundations of Genetic Algorithms, 1, 205–218. doi:10.1016/b978-0-08-050684-5.50016-1.

[31] Eiben, A. E., & Smith, J. E. (2015). Introduction to Evolutionary Computing. Natural Computing Series, Springer, Berlin, Germany. doi:10.1007/978-3-662-44874-8.

[32] Deb, K., & Agrawal, R. B. (1995). Simulated binary crossover for continuous search space. Complex systems, 9(2), 115-148.

[33] Schaffer, J. D., Caruana, R., Eshelman, L. J., & Das, R. (1989). A study of control parameters affecting online performance of genetic algorithms for function optimization. Proceedings of the 3rd international conference on genetic algorithms, 4-7 June, 1989, Fairfax, United States.

[34] Deb, K., & Agrawal, S. (1999). A Niched-Penalty Approach for Constraint Handling in Genetic Algorithms. Artificial Neural Nets and Genetic Algorithms. Springer, Vienna, Austria. doi:10.1007/978-3-7091-6384-9_40.

[35] Deb, K. (2001). Multi-Objective Optimization Using Evolutionary Algorithms. John Wiley & Sons, Chichester, United Kingdom.

[36] Deb, K., & Deb, ayan. (2014). Analysing mutation schemes for real-parameter genetic algorithms. International Journal of Artificial Intelligence and Soft Computing, 4(1), 1. doi:10.1504/ijaisc.2014.059280.

[37] Wirsansky, E. (2020). Hands-On Genetic Algorithms with Python: Applying genetic algorithms to solve real-world deep learning and artificial intelligence problems. Packt Publishing, Birmingham, United Kingdom.

[38] Wang, Y., Cai, Z., & Zhang, Q. (2011). Differential evolution with composite trial vector generation strategies and control parameters. IEEE Transactions on Evolutionary Computation, 15(1), 55–66. doi:10.1109/TEVC.2010.2087271.

[39] Ho-Huu, V., Nguyen-Thoi, T., Vo-Duy, T., & Nguyen-Trang, T. (2016). An adaptive elitist differential evolution for optimization of truss structures with discrete design variables. Computers and Structures, 165(2016), 59–75. doi:10.1016/j.compstruc.2015.11.014.

[40] Das, S., & Suganthan, P. N. (2011). Differential evolution: A survey of the state-of-the-art. IEEE Transactions on Evolutionary Computation, 15(1), 4–31. doi:10.1109/TEVC.2010.2059031.

[41] Storn, R., & Price, K. (1997). Differential Evolution - A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. Journal of Global Optimization, 11(4), 341–359. doi:10.1023/A:1008202821328.

[42] Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. Proceedings of ICNN'95 - International Conference on Neural Networks, 4, 1942–1948. doi:10.1109/icnn.1995.488968.

[43] Eberhart, R.C., & Shi, Y. (1998). Comparison between genetic algorithms and particle swarm optimization. Evolutionary Programming VII. EP 1998. Lecture Notes in Computer Science, 1447, Springer, Berlin, Germany. doi:10.1007/BFb0040812.

[44] Carlisle, A., & Dozier, G. (2001). An off-the-shelf PSO. Proceedings of the Workshop on Particle Swarm Optimization. Purdue school of engineering and technology, Indianapolis, Indiana.

[45] Shi, Y., & Eberhart, R. C. (1998). Parameter selection in particle swarm optimization. Evolutionary Programming VII. EP 1998. Lecture Notes in Computer Science, 1447. Springer, Berlin, Germany. doi:10.1007/BFb0040810.