A Novel Finite Element Model Updating Application based on Experimental Vibration Data

Quoc Bao Nguyen1 and Duong Huong Nguyen1,\*

1 Department of Bridge and Road Engineering, Hanoi University of Civil Engineering, Vietnam

duongnh2@huce.edu.vn

**Abstract.** This paper proposes a Finite Element (FE) model updating method using structural dynamic characteristics. The method is performed on a slab bridge. The FE bridge is modelled in Matlab, considering the minimum number of support points for boundary conditions and the maximum meshed element size. The prototype of the slab bridge is set up in the laboratory. The experimental vibration data is collected using fifteen accelerometers, National Instruments (NI) equipment, and a laptop. The dynamic characteristics of the experimental slab bridge such as natural frequencies and mode shapes are analysed using vibration data and used as an objective goal for updating the FE model. The updating approach is based on novel nature-inspired optimization algorithms optimization techniques such as Particle Swarm Optimization (PSO), Genetic Algorithm (GA), Fireﬂy Algorithm (FA), Cuckoo Search (CS), and Flower Pollination Algorithm (FPA). The modulus of elasticity, support distances, and plate dimensions are considered to be updated parameters. The experimental results indicate that the updated parameters are reasonable and have a clear physical meaning.

**Keywords:** Model updating; Elastic modulus; FEM; Experimental Analysis; Accelerometers, SHM.

1. Introduction

Vibration-based approaches based on experimental collected vibration response data to analyse structural dynamic characteristics [1-3]. Modal information and frequency response functions are extracted and can be used for structural health monitoring or structural updating [4-7]. The objective of modal updating is to correct an inaccurate a-priori model to agree with test results. Some unknown parameters in a finite element model will be estimated based on measurement data.

Optimization is very important in many applications, from engineering design and business planning to data mining and machine learning. The purpose of optimization is to minimize or maximize anything. To solve optimization problems, optimization algorithms and techniques are needed and their objective functions and constraints are defined. To determine some unknown parameters in a structure such as modulus of elasticity, support distances, and plate dimensions, some nature-inspired optimization algorithms are used. These are genetic algorithms, particle swarm optimization, fireﬂy algorithm, cuckoo search, flower pollination algorithm. The implementation process consists of the following steps:

**Step 1:** Construct an experimental model, and conduct experiments to determine the steel plate's natural frequencies.

**Step 2:** Build a numerical model similar to the experimental model.

**Step 3:** Update the numerical model (in step 2) with the experimental natural frequencies measured in step 1. After updating, the numerical model will have characteristics equivalent to the experimental model, and the parameters of the numerical model are then the ones that need to be determined for the experimental model.

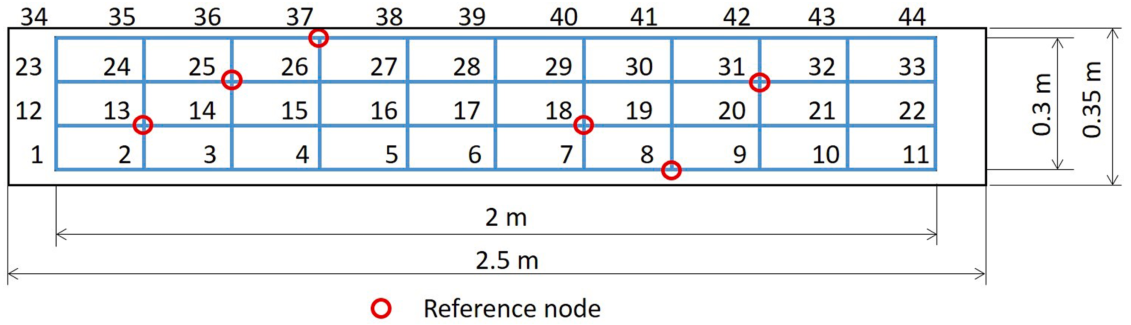
1. Experiment and determination of natural frequencies
   1. Experimental model

This study concerns a steel plate with a length of 2500 mm, a width of 350 mm, and a thickness of 10 mm. The two ends of the plate (at 250 mm from the end) are positioned on two steel supports, one of which is movable (in the form of a steel cylinder with a diameter of 50 mm and a length of 400mm), while the other is fixed (in the form of a steel cylinder with a diameter of 50 mm and a length of 400mm; a notch was cut along the bottom plate of fixed support to eliminate the movement of the cylinder). Fig. 1 illustrates these supports. The distance between the two supports is 2000mm. The vibration characteristics are determined by using 16 accelerometers, National Instruments (NI) equipment and a laptop. Each accelerometer has a sensitivity in the range of 10.13-10.50 mV/m/s2. The accelerometers are installed on the upper surface of the plate. Four setups are used to measure vibrations at 44 points on the plate, with six reference nodes used in each setup. The accelerometer layout diagram is shown in Fig. 2 and the number of accelerometers for each setup is detailed in Fig. 3.

To excite the plate’s vibration, a hammer was used and struck at different locations on the upper surface of the plate. For each experiment, the measurement time was fixed at 300 seconds and the sampling frequency was 2560 Hz. In order to analyze the vibration data, the reference-based SSI method was used [9]. The stabilization diagram was illustrated in Fig. 3 by using a stabilization criterion of 1% error in frequency, 5% error in damping, and 98% confidence in mode shape vectors. This diagram shows four alignments that are present at a minimum model order of 100. The first four bending mode shapes identified from experimental tests are plotted in Fig. 4. The four natural frequencies obtained from the experiment are 5.74 Hz, 21.89 Hz, 45.95 Hz and 72.35 Hz, corresponding to Mode 1, Mode 2, Mode 3, and Mode 4, respectively.

|  |  |
| --- | --- |
|  |  |
| (a) | (b) |

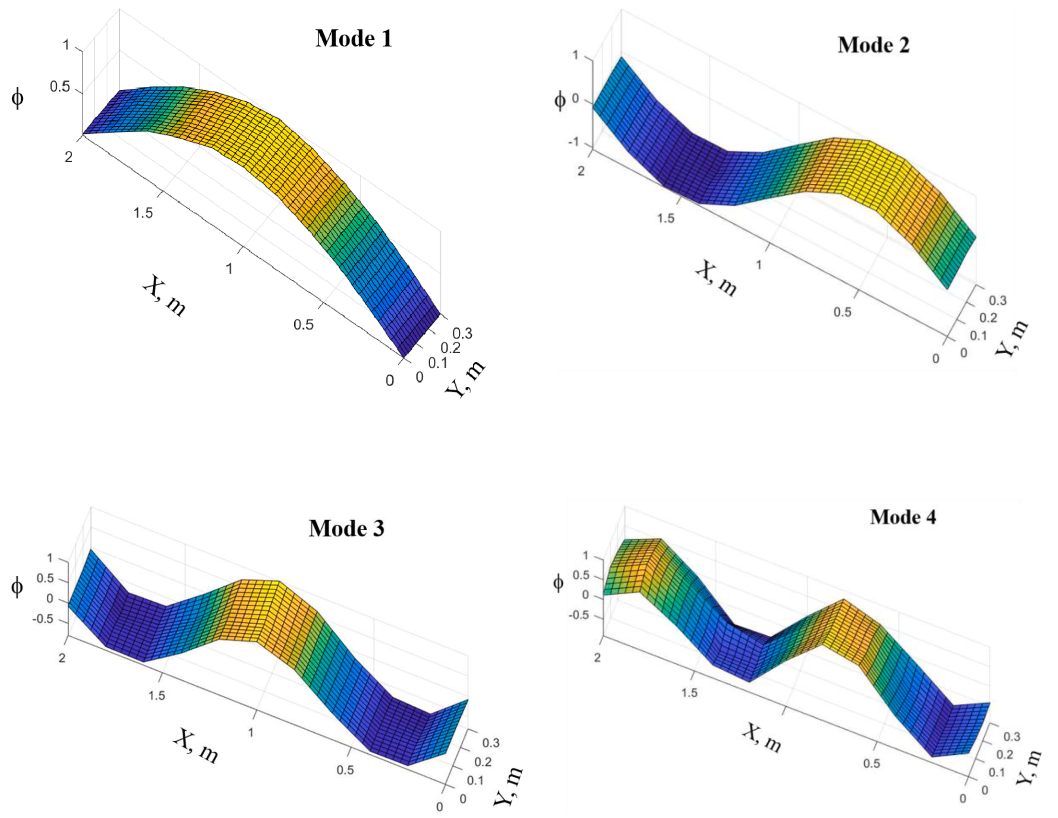
**Fig. 1.** Supports: (a) Moveable support; (b) Fixed support.



**Fig. 2.** Accelerometer layout



**Fig. 3.** Stabilization diagram



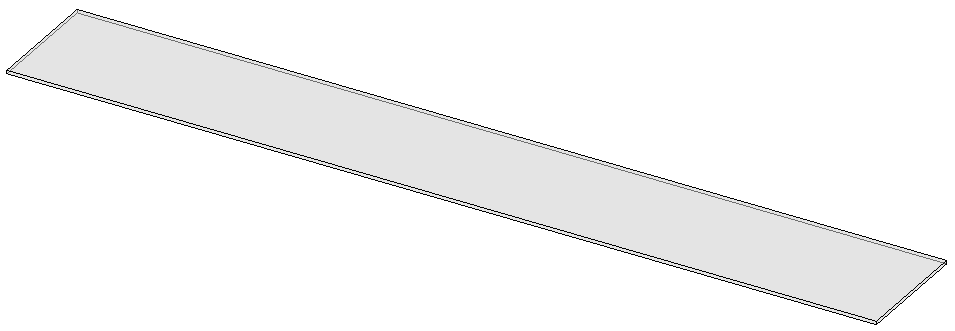
**Fig. 4.** First four bending mode shapes of the steel plate obtained from the experiment

**Table 1.** Measurement setup

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Setup | Number of accelerometers | | | | | | | | | | | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| 1 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 13 | 18 | 25 | 31 | 37 |
| 2 | 8 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 25 | 31 | 37 |  |
| 3 | 8 | 13 | 18 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 37 |  |
| 4 | 8 | 13 | 18 | 25 | 31 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 |

1. Numerical model

A numerical model of the steel plate was created in Matlab using the Partial Differential Equation toolbox [8]. This plate has a length of 2500 mm, a width of 350 mm, and a thickness of 10 mm, as shown in Fig. 5. The plate material is steel with an elastic modulus of 200 GPa, a Poisson's ratio of 0.3, and a density of 7820 kg/m3. This one was modelled using three-dimensional elements (solid elements). In order to describe the boundary conditions similar to the experimental conditions, support points were added horizontally along the plate, 250 mm apart from each end of the plate. When the number of support points is sufficiently large, they will accurately describe the boundary conditions in the experiment. A small study on the minimum number of support points will be presented below. Additionally, the maximum size of the solid elements also greatly affects the results: the smaller the maximum size, the more accurate the results will be, but the computation time will increase. Therefore, determining the maximum size of the solid elements also needs to be considered.



**Fig. 5.** Geometry of of the steel plate

As mentioned above, determining the minimum number of support points is an important research topic. The numerical model was executed multiple times with the number of support points ranging from 2 to 50. For each run of the numerical model, the natural frequencies of the first four bending modes were synthesized and plotted in Figure 1. As shown in the results, the natural frequencies of the four modes begin to converge when the number of support points is 20 or greater. Therefore, the minimum number of support points used in this paper is 20.



**Fig. 6.** Frequencies vs. number of support points

Similar to the determination of the minimum number of support points, the numerical model is performed with the maximum element size varying from 500 mm to 10 mm. The natural frequencies of the first four bending modes are also synthesized in Fig. 7. The results show that the frequencies begin to converge when the element size approaches 60 mm. In this paper, the maximum element size is fixed at 60 mm.



**Fig. 7.** Frequencies vs. Element size

The numerical model was performed in order to determine the natural frequencies of the first four bending modes with a minimum number of support nodes equal to 20 and a maximum element size of 60 mm. The four natural frequencies are 5.685 Hz, 22.045 Hz, 45.950 Hz and 70.409 Hz, corresponding to Mode 1, Mode 2, Mode 3, and Mode 4, respectively. The mode shapes are shown in Fig. 8. The the numerical, experimental natural frequencies as well as the differences between them are presented in Table 2.

**Table 2.** The numerical and experimental natural frequencies

|  |  |  |  |
| --- | --- | --- | --- |
| Mode | Frequencies | | |
| Experimental | Numerical | Error (%) |
| 1 | 5.74 | 5.685 | -0.958% |
| 2 | 21.89 | 22.045 | 0.708% |
| 3 | 45.95 | 45.950 | 0.000% |
| 4 | 72.35 | 70.409 | -2.683% |

|  |  |
| --- | --- |
|  |  |
|  |  |

**Fig. 8.** First four frequencies and mode shapes of steel plate

1. Updating the numerical model
   1. Objective function

The aim of this article is to determine the elastic modulus, the distance between two supports, and the dimensions of the steel plate through the vibration characteristics determined by experiments. Therefore, updating the numerical model to match the experimental model is necessary. This update is based on the experimental natural frequencies with the following objective function:

 (1)

Where:

* ,  are the natural frequencies of the updating numerical model and experimental frequencies, respectively.
* n is number of considered modes.
  1. Nature-inspired optimization algorithm

Nature-inspired optimization algorithms (NIOA) are a group of algorithms that draw inspiration from natural phenomena, such as animal behavior, biology, and chemical reactions, to provide solutions for engineering and medical problems. Due to the complexity of natural processes, each algorithm is unique and powerful. To improve the effectiveness of NIOAs, researchers study issues such as algorithm selection, parameter tuning, and updating old algorithms. As natural occurrences keep evolving, algorithms must also adapt and change over time, necessitating the development of new ones and the updating of existing ones. Examples of NIOAs include the genetic algorithms, particle swarm optimization, fireﬂy algorithm, cuckoo search, flower pollination algorithm, among others.

### Genetic algorithms (GAs).

GAs use a population of potential solutions and apply genetic operators, such as mutation, crossover, and selection, to simulate the evolution of the population towards better solutions. The fitness of each solution is evaluated based on a fitness function, which measures how well the solution satisfies the objective. GAs have been successfully applied to a wide range of optimization problems, including engineering design, financial modeling, and machine learning. They are particularly useful when the search space is large and complex, and traditional optimization methods are not effective [10-13].

### Particle Swarm Optimization (PSO)

PSO simulates the social behavior of swarms or flocks, introduced in 1995 by Kennedy and Eberhart. In PSO, a group of particles move through the search space, each particle represents a candidate solution to the optimization problem. The position and velocity of each particle is updated at each iteration, based on its own best solution and the best solution found by the entire swarm. The algorithm seeks to find the optimal solution by repeatedly adjusting the positions of particles based on the performance of their best known positions. PSO has been successfully applied to various optimization problems in engineering, economics, and other fields [14,15].

### Fireﬂy Algorithm (FA).

FA bases on the behavior of fireflies, proposed by Xin-She Yang in 2008 and has since been applied to solve various optimization problems in engineering, science, and other fields. The algorithm simulates the flashing behavior of fireflies and uses their attraction towards each other to search for the optimal solution. In FA, each firefly represents a candidate solution, and the intensity of its flashing represents the quality of the solution. The algorithm updates the positions of fireflies using their brightness and attraction towards other fireflies, leading to a gradual convergence towards the optimal solution [16,17].

### Cuckoo Search (CS).

CS is a population-based metaheuristic optimization algorithm inspired by the brood parasitic behavior of some cuckoo species. A set of nests represents the candidate solutions to an optimization problem, and each nest is associated with a fitness value. The algorithm iteratively generates new nests by taking inspiration from the behavior of the cuckoo bird, which lays its eggs in other birds' nests and occasionally replaces their eggs with its own. This process involves two main steps: laying eggs and removing some of the existing eggs from other nests. The eggs represent potential solutions to the optimization problem, and the process of laying eggs and removing eggs from other nests allows the algorithm to explore the search space and converge to better solutions [18,19].

### Flower Pollination Algorithm (FPA).

The Flower Pollination Algorithm (FPA) is a nature-inspired optimization algorithm that mimics the pollination behavior of flowering plants. It was proposed by Xin-She Yang in 2012 and has been applied to solve various optimization problems. In the FPA, each flower represents a solution to the optimization problem, and the pollen represents the quality of the solution. The algorithm uses the concept of pollen transfer between flowers to search for the optimal solution. The FPA has been shown to be effective in solving optimization problems in various fields, including engineering, finance, and machine learning [20,21].

* 1. Numerical model updating

Updating the numerical model to match the experimental model through the first four bending frequencies is performed using the optimization algorithms mentioned above. After updating, the values of the modulus of elasticity, support distance, and the plate dimensions (thickness) of the numerical model are the values to be determined. However, it should be noted that updating all three parameters at the same time often results in many illogical results. Therefore, updating each parameter is proposed in this article, and the updating process is as follows:

**Step 1:** Fix the support distance and the plate thickness, update the modulus of elasticity.

**Step 2:** Fix the modulus of elasticity (taken from the updated value in step 1) and the plate thickness, update the support distance.

**Step 3:** Fix the modulus of elasticity (taken from the updated value in step 1), the support distance (taken from the updated value in step 2), update the plate thickness.

Each optimization algorithm above was executed five times, and the best value of these five runs is shown in Table 3. Where:

* Best: the best value of objective functions among thousand iterations,
* Mean: the mean value of thousand objective functions,
* Worst: the worst value of thousand objective functions,
* Standard deviation: the standard deviation value of thousand objective functions,
* The number of iterations: the minimum number of iterations to achieve the best value of objective functions,
* The elastic modulus, the support distance and the plate thickness are the best values after updating.

**Table 3.** Best results of the objective functions using different methods

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Algorithm | Genetic algorithms (GA) | | | Particle Swarm Optimization (PSO) | | |
|  | Objective function when updating | | | Objective function when updating | | |
| E | ltt | h | E | ltt | h |
| Best | 37.0778 | 35.5348 | 4.9972 | 6.5564 | 3.3579 | 3.3213 |
| Mean | 43.6484 | 48.7208 | 46.9623 | 19.1113 | 10.0455 | 41.5005 |
| Worst | 508.2728 | 1148.1798 | 2088.2792 | 508.2727 | 943.5555 | 1531.2873 |
| Standard deviation | 44.9512 | 67.8261 | 161.7528 | 56.0942 | 46.0212 | 179.6536 |
| Number of iteration | 2 | 191 | 218 | 163 | 893 | 236 |
| Elastic modulus E (Pa) | 214,273,224,086 | | | 202,892,457,953 | | |
| Support distance ltt (m) | 2.0036 | | | 2.0097 | | |
| Plate thickness h (m) | 0.0096985 | | | 0.00999 | | |
| Algorithm | Fireﬂy Algorithm (FA) | | | Cuckoo Search (CS) | | |
|  | Objective function when updating | | | Objective function when updating | | |
| E | ltt | h | E | ltt | h |
| Best | 6.5564 | 3.3592 | 3.3197 | 6.5564 | 3.3602 | 3.3216 |
| Mean | 10.9657 | 19.5085 | 23.6012 | 11.3944 | 8.1239 | 13.9486 |
| Worst | 416.4423 | 1097.5680 | 1531.5051 | 244.1318 | 381.3886 | 1372.3689 |
| Standard deviation | 28.8316 | 74.5082 | 110.8081 | 24.5284 | 24.4168 | 92.7672 |
| Number of iteration | 315 | 847 | 574 | 278 | 507 | 247 |
| Elastic modulus E (Pa) | 202,914,204,708 | | | 202,894,229,229 | | |
| Support distance ltt (m) | 2.0098 | | | 2.0098 | | |
| Plate thickness h (m) | 0.0099893 | | | 0.0099901 | | |
| Algorithm | Flower Pollination Algorithm (FPA) | | |  | | |
|  | Objective function when updating | | |  | | |
| E | ltt | h |  |  |  |
| Best | 6.5564 | 3.3652 | 3.3252 |  |  |  |
| Mean | 82.1830 | 155.5868 | 209.5903 |  |  |  |
| Worst | 508.2728 | 1097.5892 | 1531.9457 |  |  |  |
| Standard deviation | 123.1223 | 250.4156 | 382.9468 |  |  |  |
| Number of iteration | 685 | 477 | 988 |  |  |  |
| Elastic modulus E (Pa) | 202,920,951,889 | | |  | | |
| Support distance ltt (m) | 2.0097 | | |  | | |
| Plate thickness h (m) | 0.0099895 | | |  | | |

Among the optimization algorithms used, GA gives the worst result when the objective functions (during the update process of elastic modulus E, support distance ltt, plate thickness h) have larger values compared to other algorithms. PSO, FA, CS, and FPA algorithms provide similar results with objective function values of 3.3213, 3.3197, 3.3216, and 3.3252, respectively. The FA algorithm has the smallest objective function value. Excluding GA, the CS algorithm converges the fastest with a total of 1032 iterations, while the PSO, FA, and FPA algorithms converge after 1292, 1736, and 2150 iterations, respectively. When considering the average value, worst value, or standard deviation, it is also evident that the Cuckoo algorithm performs better than the other algorithms. The optimized values of the modulus of elasticity, support distance, and plate thickness are different from the theoretical results (corresponding to 200 GPa, 2 m, and 0.01 m) indicating the necessity of the update process for the numerical model.

Table 4 summarizes the results of the natural frequencies of the first four vertical bending modes in the following situations: (1) before model updating; (2) after updating by GA; (3) after updating by PSO; (4) after updating by FA; (5) after updating by CS; (6) after updating by FPA; and (7) experimental measurement.

**Table 4.** Natural frequencies based on numerical and experimental models

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No | Mode | Frequency (Hz) | | | | | | |
| Before updating | Model updating - GA | Model updating - PSO | Model updating - FA | Model updating - CS | Model updating - FPA | Experimental data |
| 1 | 1st vertical bending | 5.685 | 5.69087 | 5.67331 | 5.6727 | 5.67312 | 5.67309 | 5.74 |
| 2 | 2nd vertical bending | 22.045 | 22.1058 | 22.0767 | 22.0753 | 22.0766 | 22.0762 | 21.89 |
| 3 | 3rd vertical bending | 45.95 | 46.2113 | 46.3712 | 46.3702 | 46.3742 | 46.3719 | 45.95 |
| 4 | 4th vertical bending | 70.409 | 71.1033 | 71.8904 | 71.8967 | 71.9012 | 71.8924 | 72.35 |
| No | Mode | Differences to experimental data (%) | | | | | | |
| Before updating | Model updating - GA | Model updating - PSO | Model updating - FA | Model updating - CS | Model updating - FPA | Experimental data |
| 1 | 1st vertical bending | -0.958% | -0.856% | -1.162% | -1.172% | -1.165% | -1.166% | 0.000% |
| 2 | 2nd vertical bending | 0.708% | 0.986% | 0.853% | 0.847% | 0.852% | 0.851% | 0.000% |
| 3 | 3rd vertical bending | 0.000% | 0.569% | 0.917% | 0.914% | 0.923% | 0.918% | 0.000% |
| 4 | 4th vertical bending | -2.683% | -1.723% | -0.635% | -0.627% | -0.620% | -0.632% | 0.000% |
| Sum | | 8.6169 | 4.9972 | 3.3213 | 3.3197 | 3.3216 | 3.3252 |  |

Overall, the frequencies of the first four bending modes after updating the numerical model are closer to the experimental results than before updating. The GA gave larger differences in natural frequencies compared to experimental data than the other algorithms. The remaining algorithms PSO, FA, CS, and FPA, all gave fairly similar and very close natural frequency results to the experimental data.

1. Conclusions

This article presents a method for determining the modulus of elasticity, support distance, and plate thickness of a steel plate. A similar approach can be applied to different structures. By updating the numerical model based on the vibration characteristics of the steel plate, specifically the natural frequencies, the modulus of elasticity, support distance, and plate thickness can be estimated. The process involves the following main steps: (1) building an experimental model and determining the vibration characteristics of this structure; (2) building a numerical model equivalent to the experimental model; and (3) updating the numerical model based on the vibration characteristics determined through experimentation. The optimization algorithms used include genetic algorithms, particle swarm optimization, firefly algorithm, cuckoo search, and flower pollination algorithm. The plate characteristics are updated through numerous iterations using these algorithms, with each iteration updating the modulus of elasticity, support distance, and plate thickness values. The results after updating the numerical model show that the natural frequency values are closer to the experimental values than before the update. The GA algorithm gives the worst results compared to the other optimization algorithms. Except for GA, the other algorithms provide very similar results, although the execution time and convergence rate may differ.

Acknowledgements

Duong Huong Nguyen was funded by the Postdoctoral Scholarship Programme of Vingroup Innovation Foundation (VINIF), code VINIF.2022.STS.54

References

1. Jafarkhani, R. and Masri S.F.: Finite element model updating using evolutionary strategy for damage detection. Computer‐Aided Civil and Infrastructure Engineering. 26(3): p. 207-224 (2011).
2. Meruane, V. and Heylen W.: An hybrid real genetic algorithm to detect structural damage using modal properties. Mechanical systems and signal processing. 25(5): p. 1559-1573 (2011).
3. Lin, R. and Zhu J.: Model updating of damped structures using FRF data. Mechanical systems and signal processing. 20(8): p. 2200-2218 (2006).
4. Nguyen, D.H., Nguyen Q.B., Bui-Tien T., De Roeck G., and Wahab M.A.: Damage detection in girder bridges using modal curvatures gapped smoothing method and Convolutional Neural Network: Application to Bo Nghi bridge. Theoretical and Applied Fracture Mechanics. 109: p. 102728 (2020).
5. Nguyen, D.H., Tran-Ngoc H., Bui-Tien T., De Roeck G., and Wahab M.A.: Damage detection in truss bridges using transmissibility and machine learning algorithm: Application to Nam O bridge. Smart Structures and Systems. 26(1): p. 35-47 (2020).
6. Tran-Ngoc H., Khatir S., De Roeck G., Bui-Tien T., Nguyen-Ngoc L., and Abdel Wahab M.: Model Updating for Nam O Bridge Using Particle Swarm Optimization Algorithm and Genetic Algorithm. Sensors (14248220). 18(12) (2018).
7. Nguyen, D.H. and Wahab, M.A.: Damage detection in slab structures based on two-dimensional curvature mode shape method and Faster R-CNN. Advances in Engineering Software. 176: p. 103371 (2023).
8. MATLAB: Natick, Massachusetts: The MathWorks Inc (2020).
9. Peeters B, De Roeck G.: Reference-based stochastic subspace identification for output-only modal analysis. Mechanical systems and signal processing;13(6):855–78 (1999).
10. Holland, J.: Adaptation in Nature and Artiﬁcial Systems. University of Michigan Press, Ann Arbor, MI, USA (1975).
11. De Jong, K.: Analysis of the Behaviour of a Class of Genetic Adaptive Systems. Ph.D. Thesis. University of Michigan, Ann Arbor, MI, USA (1975).
12. Goldberg, D.E.: Genetic Algorithms in Search, Optimization and Machine Learning. AddisonWesley, Reading, MA, USA (1989).
13. Yang, X.S.: Nature-Inspired Optimization Algorithms. Elsevier Insight, London (2014).
14. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proceedings of the IEEE International Conference on Neural Networks. IEEE, Piscataway, NJ, USA, pp. 1942–1948 (1995).
15. Clerc, M., Kennedy, J.: The particle swarm: explosion, stability, and convergence in a multidimensional complex space. IEEE Transactions on Evolutionary Computation 6 (1), 58–73 (2002).
16. Yang, X.S.: Nature-Inspired Metaheurisic Algorithms. Luniver Press, Bristol, UK (2008).
17. Yang, X.S.: Fireﬂy algorithms for multimodal optimization. In: Watanabe, O., Zeugmann, T. (Eds.), Proceedings of Fifth Symposium on Stochastic Algorithms, Foundations and Applications. In: Lecture Notes in Computer Science, vol. 5792. Springer, pp. 169–178 (2009).
18. Yang, X.S., Deb, S.: Cuckoo search via Lévy ﬂights. In: Proceedings of World Congress on Nature & Biologically Inspired Computing (NaBIC 2009). IEEE Publications, USA, pp. 210–214 (2009).
19. Davies, N.B.: Cuckoo adaptations: trickery and tuning. Journal of Zoology 284 (1), 1–14 (2011).
20. Yang, X.S.: Flower pollination algorithm for global optimization. In: Durand-Lose, J., Jonoska, N. (Eds.), Unconventional Computation and Natural Computation (UCNC 2012), vol. 7445. Springer, Berlin, Heidelberg, Germany, pp. 240–249 (2012).
21. Yang, X.S., Karamanoglu, M., He, X.S.: Multi-objective ﬂower algorithm for optimization. Procedia Computer Science 18 (1), 861–868 (2013).