Damage assessment of suspension footbridge using vibration measurement data combined with a hybrid bee-genetic algorithm

Lan Ngoc-Nguyen¹,², Hoa Ngoc-Tran², Samir Khatir³, Thang Le-Xuan², Quyet Huu-Nguyen², G. De Roeck³, Thanh Bui-Tien² & Magd Abdel Wahab⁴

Optimization algorithms (OAs) are a vital tool to deal with complex problems, and the improvement of OA is inseparable from practical strategies and mechanisms. Among the OAs, Bee Algorithm (BA) is an intelligent algorithm with a simple mechanism and easy implementation, in which effectiveness has been proven when handling optimization problems. Nevertheless, BA still has some fundamental drawbacks, which can hinder its effectiveness and accuracy. Therefore, this paper proposes a novel approach to tackle the shortcomings of BA by combining it with Genetic Algorithm (GA). The main intention is to combine the strengths of both optimization techniques, which are the exploitative search ability of BA and the robustness with the crossover and mutation capacity of GA. An investigation of a real-life suspension footbridge is considered to validate the effectiveness of the proposed method. A baseline Finite Element model of the bridge is constructed based on vibration measurement data and model updating, which is used to generate different hypothetical damage scenarios. The proposed HBGA is tested against BA, GA, and PSO to showcase its effectiveness in detecting damage for each scenario. The results show that the proposed algorithm is effective in dealing with the damage assessment problems of SHM.

Civil engineering structures, including bridges, highways, and dams, experience deterioration during their service life. Structures deteriorate due to various factors, including fatigue caused by overload traffic or loads of extreme events such as earthquakes, floods, or accidental loads. These damages threaten the safety of the structures in the long run, which may consequently lead to catastrophic failure or collapse. Structural damage detection at the earliest possible stage has become the core research in the field of civil engineering for many years¹. Among the current non-destructive evaluation techniques, Structural Health Monitoring (SHM) has emerged to be an indispensable system for the construction and maintenance of modern lifeline infrastructures². SHM is the process of structural inspection and evaluation over time using a wide range of sensors to collect different information that may affect the working condition of the structure, such as displacement, temperature, wind level, etc. of the structure. Based on the health evaluation of the structure, an early repair and strengthening plan can be made to maintain the safety and integrity of the structure, which would help to reduce the maintenance cost significantly. Throughout the years, the task of improving the damage detection capabilities of SHM systems has captured the attention of many researchers all over the world³–¹⁰. Although many improvements and progress have been made, the “ultimate” damage detection tool, which is the one to solve them all, is still yet to be found. This is one of the reasons why more and more efforts are being spent to augment the efficacy of SHM.

¹Laboratory Soete, Department of Electrical Energy, Metals, Mechanical Constructions, and Systems, Faculty of Engineering and Architecture, Ghent University, 9000 Gent, Belgium. ²Department of Bridge and Tunnel Engineering, Faculty of Civil Engineering, University of Transport and Communications, Hanoi, Vietnam. ³Department of Civil Engineering, KU Leuven, 3001 Leuven, Belgium. ⁴Faculty of Mechanical-Electrical and Computer Engineering, School of Engineering and Technology, Van Lang University, Ho Chi Minh City, Vietnam. ⁵Faculty of Civil Engineering, Ho Chi Minh City Open University, Ho Chi Minh City, Viet Nam. *email: magd.a.w@vlu.edu.vn; magd.abdelwahab@UGent.be
Recently, the emergence of OAs has immensely unlocked the potential of many problem solvers in dealing with complex issues\textsuperscript{11–21}. The application of OAs has also proven to be extremely compelling in enhancing the effectiveness of SHM systems\textsuperscript{14–17}. The OAs help to minimize the difference between calculated and measured results. Therefore, the calibrated model possibly predicts the behaviour of the considered structures accurately. In recent decades, many researchers have successfully applied OAs to improve the efficacy of SHM tools. For instance, Nguyen-Ngoc et al.\textsuperscript{18} proposed using Particle Swarm Optimization (PSO) algorithm coupled with an Artificial Neural Network (ANN) for damage detection, in which the proposed method was accurately superior compared to traditional PSO and ANN in detecting damages of a bridge. A novel hybrid algorithm combined with experimental data from wireless triaxial sensors was proposed by Tran et al.\textsuperscript{19} and applied for model updating of a large-scale truss bridge. In the work of\textsuperscript{20}, two recently proposed algorithms, namely Salp Swarm Optimizer (SSA) and Atom Search Optimization (ASO) were used to identify damages in a 37-bar planar truss, a 52-bar planar truss, and a 52-bar space truss.

Bee Algorithm (BA) was first proposed in 2006 by Pham et al.\textsuperscript{21}, which mimicked the food-foraging process of a honeybee colony. Since then, this algorithm has become one of the most widely used swarm-based algorithms in solving optimization problems. For example, Kavoussi et al.\textsuperscript{22} employed BA to eliminate low-order harmonics in a cascaded multilevel inverter. The obtained results outperformed GA in terms of accuracy. BA was also used to optimize the wireless sensor network coverage in the work of\textsuperscript{23}. Niknam et al.\textsuperscript{24} proposed an improved BA to optimize the placement of renewable electricity generators. Nevertheless, BA still has many drawbacks, such as easily being trapped in local minima, slow convergence speed, and depending mainly on the quality of the initial populations. These issues may reduce its effectiveness, especially in solving the optimization of complex structures. Hence, in this work, to avoid these problems, we propose a novel hybrid algorithm combining BA with GA to deal with the damage detection problem of a suspension bridge. The working principle of the proposed approach is that BA with global search techniques is used to generate initial populations with high quality; then, GA with crossover and mutation operator is used to improve the quality of the populations after each iteration. According to the No Free Lunch Theorem, no algorithm can solve all the optimization problems. For the other metaheuristic algorithms, many researchers have tried to improve them such as in the works of\textsuperscript{11,14,24,31}. Therefore, in this work, we aim to improve the BA algorithm to enrich the algorithm systems to solve optimization problems. For a better illustration, the main contributions of this work are depicted as follows:

- A hybrid algorithm combining BA and GA is proposed to enhance the effectiveness of traditional BA and GA.

- The effectiveness and correctness of the proposed method are proved by detecting damages for a real-life suspension bridge, in which both single and multiple damages are considered.

- A comparison between the proposed method and numerous other algorithms, including traditional GA and BA is conducted.

This paper is split into four main sections. Apart from the introduction part, the methodology of BA, GA, and the proposed method is introduced in "Methodology". The case study of the Na Xa suspension footbridge is presented in “Application of HBGA to damage detection of Na Xa suspension footbridge”, along with the evaluation of the effectiveness of the proposed approach, respectively. Finally, the main conclusions are drawn.

Methodology

Bee algorithm. Since its first introduction in 2006 by Pham et al.\textsuperscript{21}, BA has become one of the most widely used swarm-based algorithms in solving optimization problems. The bee algorithm is inspired by the food-foraging process of a honeybee colony. During the harvesting season, a colony will try to maximize its food-searching capacity by sending several scout bees to look for food. The scout bees search for food randomly to identify flower patches that contain plenty of nectar. When they return to the hive, the nectar qualities are assessed. If the nectar’s quality is judged to be above the standard threshold, the scout bee will perform a ritual called “waggle dance”\textsuperscript{25}. The purpose of this ritual is to notify the colony of different information regarding the high-quality nectar it found, including the location of the food source, its distance from the colony, and its rating of the nectar quality. Based on that information, the required number of bees to forage nectar at the food source could be estimated and dispatched effectively. This process helps to optimize the food-foraging process of the bee colony and ensure its survivability.

In the Bee algorithm, the bee population is divided into two main groups: “scout” bees and “elite” bees. “Scout” bees are tasked to explore randomly in the solution space to look for possible high-fitness solutions. Once the good solutions are identified, the “elite” bees are then employed to perform a local search within these solutions to look for further possible fitness improvement. The number of “elite” bees selected for a particular neighborhood depends on the ranking of the fitness of visited solutions, in which the selected process imitates the “waggle dance” of bees in real life. From there, the new solutions which have similar values to the current best finds are created and assessed. The “waggle dance” of the bees is formulated as shown in Eq. (1) below.

\[ y = x + k \]  

(1)

where \( y \) is the new position of the bee and \( x \) is the old position of the bee, respectively. \( k \) is a random parameter with values between \([-r,r]\), in which \( r \) is the radius of the bee’s patches.

Genetic algorithm. Genetic Algorithm (GA) is a metaheuristic search algorithm that was first proposed in 1973 by John Holland\textsuperscript{26}. The creation of GA is considered the founding pillar of the instrumental rise of the
so-called “evolutionary algorithms (EA). For more than 40 years, GA has remained to be one of the most widely
used OAs in solving optimization problems, especially in civil engineering$^{27-31}$. The algorithm is inspired by
Darwin’s theory of natural evolution$^{33}$, which mimics the natural selection process of biological genetics, where
only the fittest individuals are selected for reproduction to evolve better progenies for the next generation.

GA consists of three main operators, which are used to produce the next generation from the current popula-
tion: selection, crossover, and mutation. Selection helps to determine the input of the reproduction process by
selecting the individuals so-called “parents” arbitrarily. This operator also indicates the convergence rate of GA
in solving optimization problems. Some of the well-known selection techniques are tournament, Boltzmann,
roulette wheel, etc.$^{34}$. For this research, the roulette wheel is selected for the selection process of GA since it
can prevent too quick convergence from happening, which may cause the missing of minima points. Crossover
operator is used to combine the genetic information of two or more parents to form offspring. This can be done by
replacing the genes of one pair of the parent with the respective genes of the other. The third operator—mutation—is
introduced in GA to generate the genetic diversity from the parent’s population to the subsequent offspring. The
mutation is used to avoid the trapping of local minima in the optimization process by preventing the generation
of identical offspring, which may cause disruption in the convergence process to achieve the global optimum of
the problem. In GA, the mutation is formulated as follows:

$$y = x + \sigma \times \text{random} \tag{2}$$

$$\sigma = \alpha \times \frac{\text{Var}_{\max} - \text{Var}_{\min}}{2} \tag{3}$$

where $x$ and $y$ is the old and new position of the search, respectively; random is a random parameter with the
values between [-1,1] and $\alpha$ is the parameter encoding the maximum width of the mutation events.

Hybrid bee-genetic algorithm (HBGA). As demonstrated above, while BA is an advanced optimization,
it still possesses its shortcomings. In BA, the search for the food source is dependent entirely on the capabil-
ity and the numbers of the “scout” bees. If the “scout” bees indicate that the searched space is prone to be the
best (while in fact it might not be!), it will send the information back to the colony where “elite” bees will be
dispatched to search for the food in that region, which leads to a situation where they will be trapped in local
minima. Moreover, since the number of non-selected “elite” bees dispatched to this trapped region is signifi-
cantly high, only the selected bees are chosen to search for food preliminary evaluation information. This is
a waste since those non-selected bees can still search for the global optimum. Meanwhile, GA is known for
dealing with local minima problems effectively with the help of crossover and mutation operators. However,
it would not be able to store the preceding data since it was mutated and crossover in the previous generation.
That is why the Hybrid Bee-Genetic Algorithm (HBGA) is proposed. For the global search, instead of searching
arbitrarily, non-selected bees will be cross-over and mutated by a cross factor and mutation factor. The results
will be synthesized to select the best possible outcomes. The proposed hybrid algorithm shall help the search
to avoid premature convergence due to being trapped in local minima whilst minimizing the chance of missing
out the global optimum, as well as improving the convergent rate of the search algorithm. The process of HBGA
is indicated as follows:

**Step 1:** Creating the initialization of the bee group:

$$X_{ib}(0) = \begin{bmatrix}
    x_1 & x_1^1 & \ldots & x_1^m \\
    x_2 & x_2^1 & \ldots & x_2^m \\
    \vdots & \vdots & \ddots & \vdots \\
    x_{nsb} & x_{nsb}^1 & \ldots & x_{nsb}^m
\end{bmatrix} \tag{4}$$

where $n_{sb}$ is the number of “scout” bees, $m$ is the search dimension.

**Step 2:** Determination of the Fitness Evaluation:

$$F_{ib}(0) = \begin{bmatrix}
    f^1 \\
    f^2 \\
    \vdots \\
    f^{n_{ib}}
\end{bmatrix} \tag{5}$$

**Step 3:** Sorting of bees and updating the solution for $X_{ib}(0)$ sorted.

**Step 4:** Arranging the bees into two groups: non-selected Site and selected Site.

- **Group 1:** Selected Site ($n_s$)
  The number of “scout” bees $n_s$ in the non-selected site is calculated as:

$$n_s = n_{ib} - n_{es} \tag{6}$$

where $n_{ib}$ indicates the total number of “scout” bees, $n_{es}$ indicates the number of bees of the non-selected
site.

  For the Elite Site (es):
  The number of bees in the Elite Site ($n_{es}$) is selected as:
\[ n_{es} = \text{round}(\gamma_{es} \times \frac{n_{ns}}{2}) \]  

where \( \gamma_{es} = 0.4 \) is the selection factor for the Elite Site group.

Calculating the new position in \( n_{es} \), we obtain:

\[
X_{es}(it) = \begin{bmatrix}
    x^1_1 & x^1_2 & \ldots & x^1_m \\
    x^2_1 & x^2_2 & \ldots & x^2_m \\
    \vdots & \vdots & \ddots & \vdots \\
    x^{n_{es}}_1 & x^{n_{es}}_2 & \ldots & x^{n_{es}}_m
\end{bmatrix}
\]  

Recalculating the result of these elite bees to obtain the new elite site:

\[
F_{es}(it) = \begin{bmatrix}
    f^1 \\
    f^2 \\
    \vdots \\
    f^{n_{es}}
\end{bmatrix}
\]  

For the Other Site, which is the group of bees not selected by the "scout" bees, but belongs to the selected site, the total number of "scout" bees in this group is calculated as:

\[ n_{os} = \text{round}((1 - \gamma_{es}) \times \frac{n_{ns}}{2}) \]  

Calculating the new position in \( n_{os} \), we obtain:

\[
X_{os}(it) = \begin{bmatrix}
    x^1_1 & x^1_2 & \ldots & x^1_m \\
    x^2_1 & x^2_2 & \ldots & x^2_m \\
    \vdots & \vdots & \ddots & \vdots \\
    x^{n_{os}}_1 & x^{n_{os}}_2 & \ldots & x^{n_{os}}_m
\end{bmatrix}
\]  

Recalculating the result of these elite bees to obtain the new other site:

\[
F_{os}(it) = \begin{bmatrix}
    f^1 \\
    f^2 \\
    \vdots \\
    f^{n_{os}}
\end{bmatrix}
\]  

The outcome for the selected site will be calculated based on Eq. (9) and (11) as:

\[
X_s(it) = \begin{bmatrix}
    x^1_1 & x^1_2 & \ldots & x^1_m \\
    x^2_1 & x^2_2 & \ldots & x^2_m \\
    \vdots & \vdots & \ddots & \vdots \\
    x^{n_{s}}_1 & x^{n_{s}}_2 & \ldots & x^{n_{s}}_m
\end{bmatrix}
\]

where \( X_s \) is the updated position of the selected site.

The fitness function of the selected site is calculated as Eq. (14) below:

\[
F_s(it) = \begin{bmatrix}
    F_{es}(it) \\
    F_{os}(it)
\end{bmatrix}
\]

- **Group 2: Non-selected Site \( (n_{ns}) \):**

The total number "scout" bees in the non-selected site \( n_{ns} \) is calculated as:

\[ n_{ns} = 0.5 \times n_{sb} \]  

**Crossover.** For crossover, the total number of bees for crossover \( (n_c) \) is calculated based on a random number of parents are chosen for crossover with the crossover factor \( \gamma_c = 0.8 \):

\[ n_c = 2 \times \text{round}(\gamma_c \times \frac{n_{ns}}{2}) \]  

The new position of the bees is updated with the crossover number by:

\[
y_1 = [\alpha \times x_1 + (1 - \alpha) \times x_2] \times w \\
y_2 = [\alpha \times x_2 + (1 - \alpha) \times x_1] \times w
\]
where: \( y_1, y_2 \) are the offspring obtained by using crossover and \( x_1, x_2 \) are the parent bees respectively, and \( w = 0.99 \) is the added weight which is calculated as:

\[
w = w_{\text{max}} - \frac{w_{\text{max}} - w_{\text{min}}}{T} \cdot t
\]

(18)

With \( t \) and \( T \) are the current and maximum iteration, respectively.

The matrix of the new bee position created by using crossover is calculated as:

\[
X_c(t) = \begin{bmatrix}
x_1^1 & x_1^2 & \ldots & x_m^1 \\
x_2^1 & x_2^2 & \ldots & x_m^2 \\
\vdots & \vdots & \ddots & \vdots \\
x_n^1 & x_n^2 & \ldots & x_m^n 
\end{bmatrix}
\]

(19)

Recalculating the value of the objective functions after crossover by the Eq. (20):

\[
F_c(t) = \begin{bmatrix}
f_1^1 \\
f_1^2 \\
\vdots \\
f_1^n 
\end{bmatrix}
\]

(20)

**Mutation.** For mutation, the total number of bees for mutation \( n_{\text{mu}} \) is calculated based on a random number of parents are chosen for mutation with the mutation factor \( \gamma_{\text{mu}} = 0.1 \).

\[
n_{\text{mu}} = 2 \cdot \text{round}(\gamma_{\text{mu}} \times \frac{n_{\text{ns}}}{2})
\]

(21)

The new population is updated with the mutation number as Eqs. (2) and (3) above.

\[
X_{\text{mu}}(t) = \begin{bmatrix}
x_1^1 & x_1^2 & \ldots & x_m^1 \\
x_2^1 & x_2^2 & \ldots & x_m^2 \\
\vdots & \vdots & \ddots & \vdots \\
x_{\text{mu}}^1 & x_{\text{mu}}^2 & \ldots & x_{\text{mu}}^m 
\end{bmatrix}
\]

(22)

Recalculating the value of the objective functions after mutation as:

\[
F_{\text{mu}}(t) = \begin{bmatrix}
f_1^1 \\
f_1^2 \\
\vdots \\
f_{\text{mu}}^n 
\end{bmatrix}
\]

(23)

**Step 5:** Updating the new population.

The new population of bee is updated based on the previous population of selected site, crossover and mutation operators:

\[
F_{\text{group}}(t) = \begin{bmatrix}
F_c(t) \\
F_c(t) \\
F_{\text{mu}}(t) 
\end{bmatrix}
\]

(24)

The best value from the new population is sorted from the updated new population \( F_{\text{group}}(t) \):

\[
F_{\text{new}}(t) = \begin{bmatrix}
f_1^1 \\
f_1^2 \\
\vdots \\
f_{\text{new}}^n 
\end{bmatrix}
\]

(25)

**Step 6:** Updating the value for the global best \( G_{\text{best}} \).

**Step 7:** Repeat **Step 4** until the criteria are satisfied or until the number of iterations is finished.

**Step 8:** The best solution is determined by:

\[
X = G_{\text{best}}
\]

(26)

The flowchart of HBGA is shown in Fig. 1:

**Application of HBGA to damage detection of Na Xa suspension footbridge**

**Introduction to Na Xa suspension footbridge.** Located in the remote region of Nghe An Province, Vietnam, Na Xa bridge (Fig. 2) is one of the few suspension footbridges connecting the livelihoods of the tribal communes in the local area. The bridge has a total span length of 70.2 m with two high tensile main cables of
a nominal diameter of 56 mm. Suspension cables are made of Φ20 mm steel cables. The bridge towers have a height of 8 m and consist of I-shape steels, which are fixed at 1 m deep into the concrete base at the tower leg. The abutments are made of M250-graded reinforced concrete. This bridge is a typical suspension bridge in the Mid-Northern mountainous area of Vietnam, where a lot of streams are presented.

Prior to 2018, Na Xa bridge was under heavy deterioration, which caused uneasy vibration when pedestrians were crossing the bridge, hence threatening the integrity of the structure as well as the safety of the pedestrians. In 2018, the bridge was undergone retrofitting to repair different weak and damaged spots along the bridge, especially on the bridge's deck. The bridge deck was replaced and strengthened from a wooden deck to 2 mm steel plate deck for the whole of the bridge. Additional steel bars with a diameter = 3 mm were added to reinforce the
bridge deck (Fig. 3). All the cables were also re-tensioned according to the design standard of bridges. In April 2022, a vibration measurement campaign was conducted to assess the health of the structure after retrofitting.

**Finite element model.** A Finite Element (FE) model of the bridge after retrofitting is constructed using the program ANSYS Parametric Design Language (APDL). Three different types of elements are applied to model the various components of the bridge: BEAM188 element is used to model the tower, the main beam, and the cross beam of the bridge. SHELL181 element is used to model the steel deck, and LINK180 element is used for the main cables and the suspension cables on the bridge. Non-structural components of the bridge, such as handrails, bolts, etc., are considered as additional masses on the bridge with the MASS21 element. A total of 330 nodes and 578 elements are assembling the model accordingly. The material properties of the bridge are assumed to be the same for all the bridge components with the following: Young's modulus = 200 GPa, Poisson's ratio = 0.3, Density = 7850 kg/m³.

**Boundary conditions.** The bottom of the tower legs are considered as fixed support. Fixed anchor supports are also considered at the anchorage points of the two extended main suspension cables and the additional suspension cables.

**Connections.** All connections are assumed to be rigid. Connections between the bridge deck and the main beam, the cross beam, and the main beam are realized using coupled degrees of freedom (CP) command in ANSYS.

After modal analysis, 15 natural frequencies and respected mode shapes are obtained, as shown below: From Fig. 4, it can be seen that all of the identified mode shapes are global modes. However, the natural frequencies of the identified mode shapes are very close to each other. This is a noticeable issue that may make it more difficult to separate each mode, also during the analysis of the vibration measurement data.

**Collection and analysis of vibration measurement data.** The vibration measurement of Na Xa bridge was carried out from 19/04/2022 to 21/04/2022 with the aim of identifying the dynamic characteristics of the bridge after retrofitting in 2019, including natural frequencies and mode shapes. The data acquisition system consists of high-sensitivity accelerometers (PCB-393B12-sensitivity range: 965–1083 mV/g) and a data acquisition platform 08-slot CompactDAQ chassis (cDAQ-9178) connected to eight NI-9234 modules. The accelerometers were employed at vibration-sensitive points along the bridge (Fig. 5) to collect the dynamic response signals. The main sources of excitation considered were ambient (wind load) and pedestrian load. The vibration measurement was conducted in 20 min to collect vibration data from the whole bridge.

**Modal analysis.** For the system identification and modal analysis of the measured data, MACEC toolbox 3.3 is used. Fast Fourier Transform (FFT) was used to convert the measured time-series data to the frequency domain. The output-only data have been processed using OMA algorithms: covariance-based data-driven stochastic subspace identification (SSI-COV). The measured signals have been pre-processed with a concatenation of 20 min per measured setup; Fourth-order Butterworth filtering is used to filter the signal with a lowpass frequency of 5 Hz and a high pass frequency of 0 Hz to remove all phase distortion and subsequently present the frequencies range by a decimation factor of 160. To optimize the computational cost, the number of chosen block rows of the Hankel matrix is chosen as 120, and the model order range is deduced from 2 to 120 in step of 2. The stabilization diagram is obtained as shown in Fig. 6.

It is noteworthy that the presence of uncertainties during the measurement, especially from the environmental conditions, has caused many noises that can be encountered during the signal processing, which in turn causes difficulties in locating the exact natural frequencies of the structure from the stabilization diagram. In the end,
five mode shapes are identified from the measurement data, which are correlated to the result from the initial FE model, as given in Table 1.

Validation of the FE model. From Table 1, we can see that although there exist correlations between the values of the measured natural frequencies of the bridge and the values of the initial FE model, the correlated errors are still significant, which can hinder the correctness of the FE model. To improve the FE model, model updating is conducted to obtain a baseline model of the bridge, which will be used to validate the damage detection proce-
Figure 5. Installation of accelerometers on some of the vibration-sensitive points along the transversal beam (left) and tower (right).

Figure 6. Stabilization diagram with the frequency interval between 0 and 5 Hz.

Table 1. Identified vibration modes from measurement data.

| Mode number | Measured natural frequency (Hz) | Initial FE model natural frequency (Hz) | Mode shape type |
|-------------|---------------------------------|----------------------------------------|-----------------|
| 1           | 0.90                            | 0.66                                   | Vertical bending |
| 2           | 1.42                            | 1.48                                   | Vertical bending |
| 3           | 2.15                            | 2.09                                   | Torsion         |
| 4           | 2.77                            | 2.87                                   | Torsion         |
| 5           | 3.51                            | 3.91                                   | Vertical bending |
dure in the next section. It is achieved by identifying the different uncertainties of the bridge, namely the Young’s modulus of different bridge components. The result of the model updating is shown in Table 2.

Table 2. Comparison of natural frequencies before and after model updating.

| Mode number | Measured natural frequency (Hz) | Initial FE model natural frequency (Hz) | Updated FE model natural frequency (Hz) | Difference between the initial FE model and measurement (%) | Difference between updated FE model and measurement (%) |
|-------------|---------------------------------|------------------------------------------|------------------------------------------|----------------------------------------------------------|----------------------------------------------------------|
| 1           | 0.90                            | 0.66                                     | 0.85                                     | 26.7                                                     | 5.51                                                     |
| 2           | 1.42                            | 1.48                                     | 1.47                                     | 4.05                                                     | 3.62                                                     |
| 3           | 2.15                            | 2.09                                     | 2.15                                     | 2.79                                                     | 0.16                                                     |
| 4           | 2.77                            | 2.87                                     | 2.75                                     | 3.61                                                     | 0.74                                                     |
| 5           | 3.51                            | 3.91                                     | 3.52                                     | 11.40                                                    | 0.24                                                     |

From Tables 2 and 3, we can see that the natural frequencies of the updated FE model are closer to the actual values obtained from the measured data. The % differences between the natural frequencies of the model and the actual bridge are within the acceptable range. Therefore, the updated model will be used to validate the damage detection method using HBGA in the next section of the paper.

Results and discussion of damage detection of Na Xa bridge using HBGA. To validate the damage detection capability of the proposed algorithm on Na Xa bridge, two damage scenarios are generated based on the FE model of the bridge after retrofitting. In the first case, single damage is simulated by reducing Young’s modulus of one element at the main bridge beam to 50%. In the second case, multiple damages are generated by reducing Young’s modulus of two beam elements at the girder beam to 50% and 70%, respectively. The objective function is calculated as follows:

\[
\text{Objective Function} = \sum_{i=1}^{5} \left( \frac{f_i - \tilde{f}_i}{\tilde{f}_i} \right)^2 \]

where \(f_i\) and \(\tilde{f}_i\) are the natural frequencies of the analytical and experimental results, respectively, for the first five vibration modes of the structure. HBGA is employed to search for the best value of the objective function. The local search is performed by BA to search for the area, which contains the best possible solution, while GA is employed to avoid local minima and identify the optimal solutions to the problem, which are the location and the corresponding severity of the damaged beam elements within the suspension footbridge structure. To evaluate the effectiveness of the proposed method, they are compared with the original BA, GA, and Particle Swarm Optimization (PSO), which are among the most widely used OAs due to their effectiveness and simplicity. For the two damage cases, each run consists of a population size of 50 to 500 and 50 to 150 iterations. The results of the two cases are shown in Figs. 7 and 8.

Figures 7 and 8 show that for both single and multiple damage scenarios, HBGA has a better convergence rate than BA, GA, and PSO. For example, BA identifies false damages at elements 2, 3, 6, 8, and 10; GA detects damage incorrectly for all the considered elements; PSO gives wrong damage levels at elements 2, 7, and 10. For the multiple damage case, the same thing happens with false damages appearing at the incorrect locations in the case of BA, GA, and PSO: BA wrongly identifies damage at elements number 2, 3, 6, and 9, respectively; GA read...
that there exist damages in all the search elements, which is not the case considered; PSO is unable to identify the actual damage at element number 1, while it also falsely identifies damage at element number 9. From there, it can be deduced that the proposed HBGA outperforms the other considered algorithms (BA, GA, PSO) in damage identification and quantification of structure in both the single and multiple damage cases of Na Xa bridge.

In terms of computational cost, as shown in Table 4, HBGA also performs much better than BA, GA, and PSO for both single and multiple damage cases. Specifically, HBGA requires 4540 s to determine the best solution when single damage is generated in the structure, while BA, GA, and PSO cost 8029 s, 8145 s, and 8510 s, respectively. For multiple damage cases, the computational cost of HBGA is 4552 s, and for BA, GA and PSO it costs 7635 s, 7714 s, and 8525 s. It shows that HBGA is better than the other algorithms in detecting structural damages and much superior in reducing the computational cost required to solve the problem.

Conclusion

Based on the analysis and result presented herein, the following conclusions can be drawn:

(1) The proposed HBGA, which combines both the strength of BA and GA, can identify damages generated in a baseline model of suspension footbridge with a high level of accuracy. (2) The accuracy of the proposed method is also proven to be much superior when compared with the other optimization algorithms (BA, GA, and PSO). (3) The efficiency of the method is further assured by the computational cost statistic results.

This paper proposes a novel adaptive hybrid metaheuristic optimization algorithm, namely HBGA, to solve the damage detection problem of suspension footbridges. The HBGA is able to fuse both the strengths of the classic Bee Algorithm and Genetic Algorithm into a powerful technique that can not only identify damages, but also quantify their severity accurately, as demonstrated in the chosen case study of Na Xa bridge. For the validation of its effectiveness, the proposed HBGA outperformed both its makers as well as the widely used PSO algorithm. It is noteworthy that the proposed method has plenty of room for improvement in future research. This method can also be extended to solve various optimization of other lifeline infrastructures such as dams, skyscrapers, etc.
Data availability
The datasets used and/or analysed during the current study available from the corresponding author on reasonable request.

Received: 19 August 2022; Accepted: 15 November 2022
Published online: 22 November 2022

References
1. Khatir, S., Tiachacht, S., Thanh, C.L., Tran-Ngoc, H., Mirjalili, S., Wahab, M. A. A robust frf damage indicator combined with optimization techniques for damage assessment in complex truss structures. Case Stud. Constr. Mater. e01197 (2022).
2. Sohn, H., Farrar, C.R., Hemez, F.M., Shunk, D.D., Stinemates, D.W., Nadler, B.R., & Czarnecki, J.J. A review of structural health monitoring literature: 1996–2001. Los Alamos National Laboratory, USA, 1 (2003).
3. Doebling, S.W., Farrar, C.R., Prime, M.B., & Shevitz, D.W. Damage identification and health monitoring of structural and mechanical systems from changes in their vibration characteristics: A literature review. (1996).
4. Ko, J. M., & Ni, Y. Q. Technology developments in structural health monitoring of large-scale bridges. Eng. Struct. 27(12), 1715–1725 (2005).
5. Chen, H.P. Structural health monitoring of large civil engineering structures (2018).
6. Hyun Woo Park and Taejeong Lim. Investigating a common premise in structural health monitoring: Are higher modal frequencies more sensitive to an incipient crack on a beam than lower ones? Eng. Struct. 176, 385–395 (2018).
7. Cantero-Chinchilla, S., Beck, J.L., Chiachó, M., Chiachó, I., Chronopoulos, D., & Jones, A. Optimal sensor and actuator placement for structural health monitoring via an efficient convex cost-benefit optimization. *Mech. Syst. Signal Process.* 144, 106091 (2020).

8. Bui-Tien, T. et al. Damage detection in structural health monitoring using hybrid convolution neural network and recurrent neural network. *Frattura ed Integrità Strutturale* 16(59), 461–470 (2022).

9. Hakezzmann, G. et al. Cyber-physical design of distributed structural health monitoring with wireless sensor networks. *IEEE Trans. Parallel Distrib. Syst.* 25(1), 63–72 (2013).

10. Cho, S. et al. Structural health monitoring of a cable-stayed bridge using wireless smart sensor technology: data analyses. *Smart Struct. Syst.* 6(5–6), 461–480 (2010).

11. Kumar, S., Yadav, D. K. & Ali Khan, D. An accelerating PSO algorithm based test data generator for data-flow dependencies using dominance concepts. *Int. J. Syst. Assur. Eng. Manage.* 8(2), 1534–1552 (2017).

12. Kumar, S., Yadav, D. K. & Khan, D. A. Artificial bee colony based test data generation for data-flow testing. *Indian J. Sci. Technol.* 9(9), 1–10 (2016).

13. Singh, H., Kumar, Y. & Kumar, S. A new meta-heuristic algorithm based on chemical reactions for partitional clustering problems. *Evol. Intel.* 12(2), 241–252 (2019).

14. Ho, L. V. et al. A hybrid computational intelligence approach for structural damage detection using marine predator algorithm and feedforward neural networks. *Comput. Struct.* 252, 106568 (2021).

15. Viet, L. H. & Thi, T. T. Swarm intelligence-based technique to enhance performance of ANN in structural damage detection. *Trans. Comput. Sci. J.* 73(1), 1–15 (2022).

16. Thi, T. T. An applied grey wolf optimizer for scheduling construction projects. *Trans. Comput. Sci. J.* 73(4), 397–411 (2022).

17. Nguyen-Ngoc, L., Tran-Ngoc, H., Nguyen-Duc, B., Nguyen-Le Minh, D., Bui-Tien, T., & Abdel Wahab, M. Damaged detection in structures using artificial neural networks and genetic algorithms. In Bui-Tien, T., Ngoc, L. N., De Roeck, G., editors, *Proceedings of the 3rd International Conference on Sustainability in Civil Engineering*, pages 33–38, Singapore, 2021. Springer Singapore.

18. Nguyen-Ngoc, L., Hoa Tran, N., Bui-Tien, T., Mai-Duc, A., Wahab, M. A., Nguyen, H. X., & De Roeck, G. Damage detection in structures using particle swarm optimization combined with artificial neural network. *Smart Struct. Syst.* 28(1), 1–12 (2021).

19. Tran-Ngoc, H., Khatir, S., Le-Xuan, T., De Roeck, G., Bui-Tien, T., Wahab, M. A. Finite element model updating of a multispan bridge with a hybrid metaheuristic search algorithm using experimental data from wireless triaxial sensors. *Eng. Comput.* 1–19 (2021).

20. Khatir, S., Tiachacht, S., Thanh, C.L., Tran-Ngoc, H., Mirjalili, S., & Wahab, M.A. A new robust flexibility index for structural damage identification and quantification. *Eng. Fail. Anal.* 129, 105714 (2021).

21. Pham, D. T., Ghanbarzadeh, A., Koc, E., Otri, S., Rahim, S., & Zaidi, M. The bees algorithm—a novel tool for complex optimisation problems. In *Intelligent production machines and systems*, pages 454–459. Elsevier (2006).

22. Kovouzi, A. et al. Application of the bee algorithm for selective harmonic elimination strategy in multilevel inverters. *IEEE Trans. Power Electron.* 27(4), 1689–1696 (2011).

23. Khalaf, O. I., Abdulshah, G. M. & Sabbar, B. M. Optimization of wireless sensor network coverage using the bee algorithm. *J. Inf. Sci. Eng.* 36(2), 377–386 (2020).

24. Niknam, T., Taheri, S. I., Aghaei, J., Tabatabaei, S. & Nayeripour, M. A modified honey bee mating optimization algorithm for multi objective placement of renewable energy resources. *Appl. Energy* 88(12), 4817–4830 (2011).

25. Grütter, C., & Farina, W. M. The honeybee waggle dance: Can we follow the steps? *Trends Ecol. Evol.* 24(5), 242–247 (2009).

26. Holland, J. H. Genetic algorithms and the optimal allocation of trials. *SIAM J. Comput.* 2(2), 88–105 (1973).

27. Furuta, H., Maeda, K. & Watanabe, E. Application of genetic algorithm to aesthetic design of bridge structures. *Comput. Aid. Civ. Infrastruct. Eng.* 10(6), 415–421 (1995).

28. Kumar, M., Hasan, Uperti, N., Gupta, D et al. Genetic algorithm: Review and application. *Available at SSRN 529843*. 2010.

29. Zhu, K., Song, H., Liu, L., Gao, J., & Cheng, G. Hybrid genetic algorithm for cloud computing applications. In *2011 IEEE Asia-Pacific Services Computing Conference*, pages 182–187. IEEE (2011).

30. Katoch, S., Chauhan, S.S., & Kumar, V. A review on genetic algorithm: past, present, and future. *Multimed. Tools Appl.* 80(5), 8091–8126 (2021).

31. Kumar, S., Yadav, D. K., & Ali Khan, D. A novel approach to automate test data generation for data flow testing based on hybrid adaptive PSO-GA algorithm. *Int. J. Adv. Intell. Paradigms* 9(2/3), 278–312 (2017).

32. Goldberg, D.E., & Samtani, M.P. Engineering optimization via genetic algorithm. In *Electronic computation—original draft; H.T. editing, methodology; T.L.X. software, validation; S.K. revision, Q.N.H. software, investigation; G.D.R. supervision; B.T.T. conceptualization M.A.W. conceptualization, supervision, and reviewing editing and writing."

**Acknowledgements**

The authors acknowledge the financial support of VLIR-UOS TEAM Project, VN2018TEA479A103, ‘Damage assessment tools for Structural Health Monitoring of Vietnamese infrastructures’ funded by the Flemish Government. This work is also funded by Vingroup and supported by Innovation Foundation (VINIF) under project code VINIF.2021.DA00192. The authors wish to express their gratitude to Van Lang University, Vietnam for financial support for this research.

**Author contributions**

L.N.N. software, validation, writing—original draft; H.T. editing, methodology; T.L.X. software, validation; S.K. revision, Q.N.H. software, investigation; G.D.R. supervision; B.T.T. conceptualization M.A.W. conceptualization, supervision, and reviewing editing and writing.

**Competing interests**

The authors declare no competing interests.

**Additional information**

Correspondence and requests for materials should be addressed to M.A.W.
Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

© The Author(s) 2022