Journal of Computational Science 55 (2021) 101451

Contents lists available at ScienceDirect

Journal of Computational Science

journal homepage: www.elsevier.com/locate/jocs

# YUKI Algorithm and POD-RBF for Elastostatic and dynamic crack identification

Brahim Benaissa [a,*], Nourredine Aït Hocine [b], Samir Khatir [c,d], Mohamed Kamel Riahi [e,f], Seyedali Mirjalili [g,h]

[a] Toyota Technological Institute, Department of Mechanical Systems Engineering, Design Engineering Lab, 468-8511 Aichi, Nagoya, Tempaku Ward, Hisakata, 2 Chome-12-1, Japan
[b] INSA CVL, Univ. Tours, Univ. Orléans, LaMé, 3 rue de la Chocolaterie, CS 23410, 41034, Blois, Cedex, France
[c] Soete Laboratory, Faculty of Engineering and Architecture, Ghent University, Technologiepark Zwijnaarde 903, B-9052, Zwijnaarde, Belgium
[d] Faculty of Civil Engineering, Ho Chi Minh City Open University, Ho Chi Minh City, Viet Nam
[e] Department of Mathematics, Khalifa University of Sciences and Technology, P.O. Box 127788, Abu Dhabi, United Arab Emirates
[f] Emirates Nuclear Technology Center (ENTC), Khalifa University of Science and Technology, United Arab Emirates
[g] Centre for Artificial Intelligence Research and Optimisation, Torrens University Australia, Fortitude Valley, Brisbane, 4006, QLD, Australia
[h] Yonsei Frontier Lab, Yonsei University, Seoul, Republic of Korea

ARTICLE INFO

*Keywords:*
Yuki Algorithm
POD-RBF
Crack identification
Inverse problem
Static and dynamic analysis

ABSTRACT

This paper proposes a new metaheuristic algorithm with a search space reduction capability guided by simple formalism. The search population focuses partially on the inside the local search area while the rest explore globally, looking for better search areas. We call the new algorithm by YUKI Algoritm (YA) and employ it in a crack identification problem. With the aid of a set of measurements taken on the defected structure, we aim at identifying the crack parameters such as length and orientation. To this end, we use the so-called model reduction technique through Proper orthogonal Decomposition (POD) endorsed with Radial Basic Function (RBF), which helps in predicting (numerically) the measurement at new points (out of the set of sensors) via interpolation. This method is widely used in this context and was proven very effective computational-wise. In our study of the performance of YA, we deal with two cases; Firstly, in the case of the Elastostatic study. And secondly, in the case of dynamic analysis. We compare the performance of the suggested algorithm with the performance of well-known optimization methods, such as Teaching Learning Based Optimization (TLBO), Cuckoo Search (CS), and the Gray Wolf Optimizer (GWO). The results show that YA provides accurate and faster results compared to the mentioned algorithms.

## 1. Introduction

Structural health monitoring (SHM) is essential for engineering applications such as beams, plates, bridges, and offshore. Inverse crack identification predicts the unknown crack characteristics inside a structure (size, position, and orientation) using the structural response to an exerted excitation. Theoretical backgrounds related to inverse crack identification in mechanics are thoroughly developed; we refer the reader to [1–3]. Elastostatic approaches consider crack identification as a geometrical inverse problem based on measurable data for given boundary conditions, namely the material properties, geometry, and loading conditions [3,4]. On the other hand, approaches based on dynamic studies formulate the problem using the vibrational response of structures [5,6]. There exist non-mechanical crack detection techniques such as [7–9].

In inverse crack identification, the evaluation function generally compares, in an iterative search, between the measured structural response and the estimated counterpart. In such cases, where the behavior is not linear, metaheuristic optimization methods are powerful tools for predicting the unknown crack parameters by minimizing such comparison function at the cost of significant computational efforts. Moreover, they often require thousands of fitness evaluations, corresponding to several simulation runs at every iteration [4,10–19]. Nonetheless, model reduction techniques contribute heavily to such computationally demanding approaches.

Proper Orthogonal Decomposition (POD) is an efficient model reduction technique based on the projection of the original model onto a subspace using full model snapshots [20–22]. This method offers a computational cost advantage [23–27]. Winton et al. [23], the authors

* Corresponding author.
*E-mail address:* benaissa@toyota-ti.ac.jp (B. Benaissa).

https://doi.org/10.1016/j.jocs.2021.101451
Received 9 April 2021; Received in revised form 20 August 2021; Accepted 15 September 2021
Available online 23 September 2021
1877-7503/© 2021 Elsevier B.V. All rights reserved.

coupled the POD with Levenberg–Marquardt optimization to approximate the property of hydraulic conductivity in saturated domains. Furthermore, in [24], this model reduction method has been used to estimate unknown heat conductivities and film coefficient distribution. Brigham et al. [25] presented a POD and Finite Element Method for Elastostatic material characterization. Finally, Rogers et al. [26] tackled the conduction problem in different structures to estimate the Isotropic material parameters of a 3D bar in tension.

Radial Basis Functions (RBF) is a powerful interpolation method characterized by handling a random distribution of data points. It gives the possibility to consider a large number of variables [28–35]. In [28], Buljak and Mier developed the POD-RBF approach based on simulation data to estimate material properties in the indentation test. Similarly, in [29], Hoang et al. used this method to characterize interface tissue in dental implant systems. In [32], POD-RBF is used to estimate the depth residual stress profiles induced by surface treatment of metal components. Also allowed using data that is entirely experimental [36]. In [22] POD-RBF model is built up based on the vibration response of a composite beam to identify crack position and size. In [37], Bocciarelli et al. used the POD-RBF techniques to characterize metallic materials in the case indentation test using experimental data. Deep Neural Networks (DNN) are being used to advance damage identification research. In [38], Cosmin et al. suggest an approach to the boundary values problem. The mesh-free method uses scattered sets of points in the training and evaluation sets. Such learning methods can be used for solving the forward and inverse problem simultaneously.

Metaheuristic techniques have become popular over the last decade. The popularity of these algorithms is due to several factors, including their versatility, gradient-free mechanism, and avoidance of local optima. The first two benefits stem from the fact that metaheuristics only look at the inputs and outputs when considering and solving optimization problems. They are highly adaptable when it comes to solving ill-posed inverse problems. Artificial bee colony algorithm was used in flaws prediction problem, using extended Finite Element Method (XFEM) in a dynamic study of a plate [5]. Genetic Algorithm (GA) was used in [6,39] to detect cracks in flat membrane structures under static and dynamic loads. [40] compares the performance of Particle Swarm Optimization to GA in the crack identification study based on POD-RBF. Jaya algorithm is used with Artificial Neural Network to identify crack size and orientation based on experimental vibration modes [41].

This paper suggests a new metaheuristic algorithm to solves the inverse crack identification problem. We are considering two cases. The first case is an Elastostatic study based on FEM. And the second is the dynamic study based on experimental data. Furthermore, finally, we compare the performance of the suggested algorithm to the well-known algorithms. Namely Cuckoo Search (CS) [42], Teaching Learning Based Optimization (TLBO) [43], and the Gray Wolf Optimizer (GWO) [44].

We organize the rest of the paper as follows: Section 2 describes the YUKI algorithm in detail. In Section 3, we present the POD-RBF method. Section 4 discusses the crack identification formulation in the Elastostatic study, where first, we present the POD-RBF model details and the prediction validation with FEM results. Section 4.3 to 4.5 discuss the crack identification results in cases of different crack sizes and orientations, with noisy data, and with a limited number of sensors. Section 4.6 compares the performance of YA to other algorithms in terms of precision and computational cost. Section 5 discusses crack identification in the case of a dynamic study based on experimental data. In 5.1, we present the experimental setup. In Section 5.2, the POD-RBF model details are presented, followed by validation with experimental output. Finally, in Sections 5.3 and 5.4, the results of crack identification are discussed, Comparing YA with other algorithms in terms of precision and computational cost.

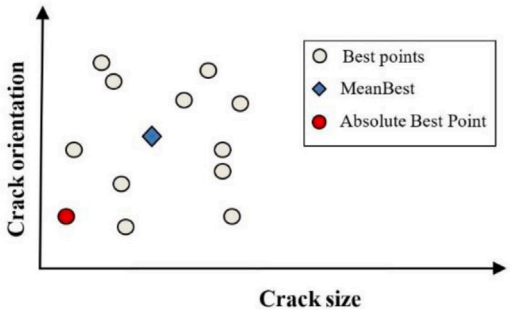


**Fig. 1.** Illustration of the MeanBest point.

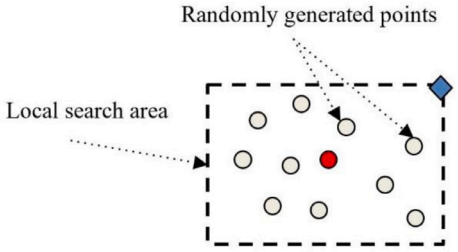


**Fig. 2.** Illustration of the local search area. The red circle stands for the absolute best point, and the blue rhombus represents the MeanBest point. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

## 2. Yuki algorithm for inverse crack identification

This algorithm suggests a principle of dynamic search space reduction. By creating a local search area around the absolute best solution found so far. The algorithm adapts the size of the search area dynamically throughout the progress of the search, using the distance between two points as a reference.

The first point is the absolute best point. It corresponds to the minimum fitness value. Furthermore, The second point is called MeanBest. We calculate this point as the center of the "best points" could. The best points here are the best solutions found so far by each member of the YA population (see Fig. 1).

The following expression calculates the local boundaries:

$$D = \left| X_{best} - X_{MeanBest} \right| \tag{1}$$

$$\boldsymbol{LT} = X_{best} + D \tag{2}$$

$$\boldsymbol{LB} = X_{best} - D \tag{3}$$

Respectively $\boldsymbol{LT}$ and $\boldsymbol{LB}$ are the Local top and bottom boundaries of the local search space. Furthermore $\boldsymbol{X_{best}}$ Is the point corresponding to the absolute best fitness value at the current iteration. And the term $\boldsymbol{X_{MeanBest}}$ is the mean of the Best Points vector (see Fig. 2).

The second concept is to create a random distribution of points $\boldsymbol{X_{loc}}$ Inside the local search area, split the search into two parts; One part of the population to explore outside the local search area. The other is assigned to focus on searching around the center of the search area. The size of the exploration population changes iteratively in this manner:

$$if\ rand < 1 - K\ /\ K_{max} \tag{4}$$

Eq. (4) condition compares the randomly generated value, between 0 and 1, to the output of the Eq. (4) expression, which is also between 0 and 1. $\boldsymbol{K}$ is the current iteration and $\boldsymbol{K_{max}}$ Represent the value of maximum iteration set initially as a stopping criterion. At early iterations, the output corresponds to a high possibility for the "if

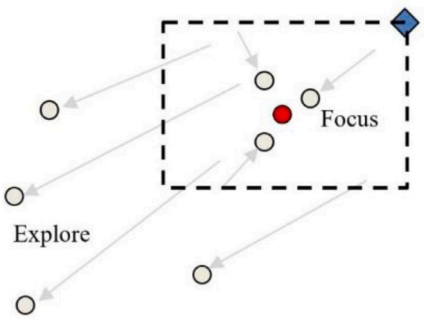


**Fig. 3.** Illustration of the exploration and focus concepts.

statement" to be True, and at late iterations, this possibility decreases linearly. This way, we assign most of the population toward exploring initially, then slowly shift toward exploitation.

Exploration population look in the direction away from the MeanBest expressed as follows:

$$E = X_{loc} - randi(10) \times X_{MeanBest} \tag{5}$$

Where ***randi*(4)** is a random integer between 1 and the value of the exploration control parameter chosen as an integer between 2 and 10, the following equation calculates the new solutions:

$$X_{new} = X_{loc} + rand \times E \tag{6}$$

"rand" here stands for a random value between 0 and 1. Note here that we use the same random value for all design variables.

The rest of the population, not assigned to exploration, is pushed to search around the local area center. The following equation governs this process:

$$\boldsymbol{F} = X_{loc} - X_{best} \tag{7}$$

$$X_{new} = X_{loc} - rand \times \boldsymbol{F} \tag{8}$$

Where $\boldsymbol{F}$ is the distance between the selected local point to the absolute best solution, similarly, *rand* here is the random value between 0 and 1. Note also here that we use the same random value for all design variables (see Fig. 3).

We update the absolute best solution and the individual best solutions at the end of each iteration. For details, Fig. 4 shows the YA pseudo-code.

In this study of crack identification, the fitness function is:

$$\begin{cases} f(\boldsymbol{P}) = \dfrac{\|\boldsymbol{u}(\boldsymbol{P}_0) - \boldsymbol{u}(\boldsymbol{P})\|^2}{\|\boldsymbol{u}(\boldsymbol{P}_0)\|^2} \\ f\left(\boldsymbol{P}_{optimal}\right) = min[f(\boldsymbol{P})] \end{cases} \tag{9}$$

Where the response vectors $\mathbf{u}(\mathbf{P})$ of the suggested cracks is compared to the reference response $\mathbf{u}\left(\mathbf{P}_0\right)$ That was caused by the unknown crack.

The following points discuss the theory of the suggested algorithm.

- By exporting new areas situated away from the MeanBest point, we focus the exploration effort in one direction, which increases the possibility of finding new solutions compared to randomly exploring the global area.
- By multiplying the MeanBest value to a random integer, we create a pulsating effect, essential for suggesting different solutions in every iteration. It also helps extend the reach of the search.
- As long as there is good coverage of the global search space, the distance between Best Points remains safely large. Thus the local search area avoids collapsing on a local optimum.

Yuki algorithm pseudo-code

*Load the search parameters*
*Initiate the population X*
*Evaluate the fitness*
*Calculate the* $X_{MeanBest}$ *and* $X_{best}$
***for*** $K = 1$ *to* $K_{max}$
  *Calculate the local boundaries*
  *Generate the local population randomly*
  ***if*** $rand < 1 - K/K_{max}$
    *Calculate the exploration solutions*
  ***else***
    *Calculate focus solutions*
  ***end***
  *update the* $X_{MeanBest}$
  *update the* $X_{best}$ *if better solutions are found*
***end***
*return* $X_{best}$

**Fig. 4.** YUKI Algorithm.

- The search area sizes are independent across search dimensions. Because this algorithm can create variations for which the local search area is significant in one dimension and very small in another dimension. Thus the search focus is reacting to the sensitivity of each design variable.
- The local search area's size can increase dynamically if the algorithm finds a new best solution far from the current local search area.
- As solutions converge to the optimum, the local search area gets smaller,

## 3. POD-RBF for static crack identification problem

POD is used in this study, coupled with the interpolation method RBF to provide the structural response in the inverse problem. Based on data measured a priori and stored in the matrix U.

$$\mathbf{U} = \begin{bmatrix} u_1^1 & u_1^2 & \cdots & u_1^S \\ u_2^1 & u_2^2 & & u_2^S \\ \vdots & \vdots & \ddots & \vdots \\ u_N^1 & u_N^2 & \cdots & u_N^S \end{bmatrix} \tag{10}$$

**U** is constituted of N snapshots, denoted as $u^j$ (for $j = 1, 2 \ldots S$). S is the size of the snapshot vector. In this study, the size of the structural response data. On the other hand, the matrix $\mathbf{P} = ([\mathbf{L_1}, \boldsymbol{\theta_1}], \ldots, [\mathbf{L_N}, \boldsymbol{\theta_N}])$ stores the crack parameters. L is the crack length, and $\boldsymbol{\theta}$ is its orientation. In the Elastostatic study, detailed in Section 4.1, the snapshot consists of boundary displacement measurements at boundary nodes. Furthermore, in the dynamic study, detailed in Section 5.2, the Frequency Response Function measurements constitute the snapshot matrix.

At this point, we extract a set of orthogonal vectors $\boldsymbol{\Phi}$, upon which we project the measurement data matrix **U** to obtain the amplitude matrix **A** as follows:

$$\mathbf{A} = \boldsymbol{\Phi}^{\mathrm{T}} \cdot \mathbf{U} \tag{11}$$

Matrix **A** approximates the structural response data. It is worth noticing here that we calculate $\boldsymbol{\Phi}$ through the POD operation [28,45], by extracting the eigenvectors of the covariance matrix $\mathbf{C} = \mathbf{U}^{\mathrm{T}} \cdot \mathbf{U}$, through singular value decomposition. We then perform a truncation to reduce the basis vectors $\boldsymbol{\Phi}$ and formulate a lower rank version $\hat{\boldsymbol{\Phi}}$ by preserving the first k ($k \ll S$) columns of $\boldsymbol{\Phi}$ that correspond to the largest eigenvalues. Consequently, a new amplitude matrix $\hat{\mathbf{A}}$ writes:

$$\hat{\mathbf{A}} = \hat{\boldsymbol{\Phi}}^{\mathrm{T}} \cdot \mathbf{U}. \tag{12}$$

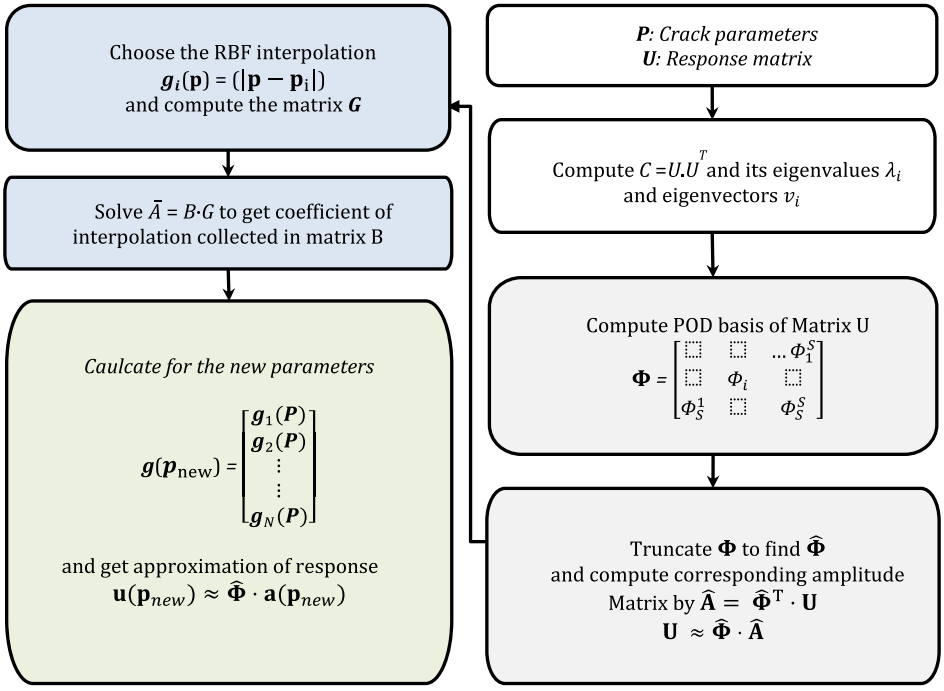


**Fig. 5.** POD-RBF methodology.

Note here that one can recover the original data using the reduced model by the following approximation.

$$\mathbf{U} \approx \widehat{\mathbf{\Phi}} \cdot \widehat{\mathbf{A}}. \tag{13}$$

At this step, we link the structural response data to the crack parameters through Radial Basis Function (RBF) and a coefficient matrix **B** and an interpolation matrix **G** as $\mathbf{B} \cdot \mathbf{G} = \widehat{\mathbf{A}}$.

In the case where **G** is a non-singular matrix, one can write the following expression

$$\mathbf{B} = \widehat{\mathbf{A}} \bullet \mathbf{G}^{-1}. \tag{14}$$

Here, the matrix of interpolation **G** is persymmetric, and it englobes the RBF function: $\mathbf{g}_i(\mathbf{p}) = |\mathbf{p} - \mathbf{p}_i|$ as such

$$\mathbf{G} = \begin{bmatrix} g(|p_1 - p_1|) & g(|p_1 - p_2|) & \dots & g(|p_1 - p_N|) \\ g(|p_2 - p_1|) & g(|p_2 - p_2|) & & g(|p_2 - p_N|) \\ \vdots & \vdots & \ddots & \vdots \\ g(|p_N - p_1|) & g(|p_N - p_2|) & \dots & g(|p_N - p_N|) \end{bmatrix} \tag{15}$$

$\mathbf{g}_i(\mathbf{p})$ calculated for each parameter in the matrix **G** where $\mathbf{p}_i$ is the parameter corresponding to $\mathbf{U}_i$ (i=1,2,...,N). The argument of the *i*th RBF is the distance $|\mathbf{p} - \mathbf{p}_i|$, **p** and $\mathbf{p}_i$ being respectively current and reference parameters.

After we calculate the coefficient matrix **B**, a low-dimensional model of Eq. (14) is put in vector form, as follows, and used to calculate the amplitude vector $\mathbf{a}(\mathbf{p}_{new})$ for new parameters, outside the positions of the sensors:

$$\mathbf{a}(\mathbf{p}_{new}) = \mathbf{B} \cdot \mathbf{g}(\mathbf{p}_{new}) \tag{16}$$

Eq. (14) rewrites then in a vector form to calculate the interpolated data **u** corresponding to the new cracks parameter $\mathbf{p}_{new}$. This way, we calculate the structural response corresponding to the new crack:

$$\mathbf{u}(\mathbf{p}_{new}) \approx \widehat{\mathbf{\Phi}} \cdot \mathbf{a}(\mathbf{p}_{new}) \tag{17}$$

The POD-RBF model, Eq 16. is, therefore, able to reproduce unknown structural responses corresponding to any set of crack parameters **p**. However, note that it can lead to weak precision due to the extrapolation outside the range of measured crack parameters **p**. Fig. 5 summarize building the POD-RBF.

## 4. Identification using static boundary displacement

### 4.1. FEM simulation setup

For this two-dimensional stress analysis problems under plane strain conditions, eight basic independent parameters: $\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{\epsilon_x}, \boldsymbol{\epsilon_y}, \boldsymbol{\gamma_{xy}}, \boldsymbol{\sigma_x}, \boldsymbol{\sigma_y}, \boldsymbol{\tau_{xy}}$ [46]. $u$ and $v$ are the displacements in X and Y direction respectively, $\boldsymbol{\epsilon_x}, \boldsymbol{\epsilon_y}$ are normal strain components, $\boldsymbol{\gamma_{xy}}$ is the Shear strain component, $\sigma_x, \sigma_y$ are normal stress components and $\boldsymbol{\tau_{xy}}$ is the shear stress component.

Elastostatics here means the study of linear elasticity under the conditions of equilibrium, in which all forces on the elastic body sum to zero and $\boldsymbol{\tau_{xy}} = \boldsymbol{\tau_{yx}}$. The displacements are not a function of time. Furthermore, the components of Cauchy's strain tensor for minor strain problems follow this strain displacement relationship:

$$\boldsymbol{\epsilon_x} = \frac{\partial \boldsymbol{u}}{\partial x}, \boldsymbol{\epsilon_y} = \frac{\partial \boldsymbol{v}}{\partial y}, \boldsymbol{\gamma_{xy}} = \frac{\partial \boldsymbol{v}}{\partial x} + \frac{\partial \boldsymbol{u}}{\partial y} \tag{18}$$

For isotropically elastic materials, the stress components are related to the strain components by Hooke's law as $\boldsymbol{\sigma} = \boldsymbol{E\epsilon}$ where $\boldsymbol{E}$ is Young's modulus.

For the FEM simulation, we consider a plate with 1 mm thickness, 30 mm height, and 10 mm width (Fig. 6). With a Young modulus $\boldsymbol{E}$ of 210 GPa and Poisson coefficient of 0.3. Under a displacement of 0.3 mm at the upper and lower sides of the plate.

The vertical side is meshed using 120 quadrilaterals and the horizontal side with 40 elements. The meshing of the crack consists of two types of areas: the crack tips, covering both crack endings, and the middle of the crack. Crack tips are meshed by the sweep function. The circular area has 0.5 mm of the radius, divided into 20 elements along the contour of the circle. And 10 elements along the radius of the circle. There is a total of 200 elements concentrated at each tip of the crack.

### 4.2. The POD-RBF model validation

The snapshot matrix **U** consists of the measured displacement at the right vertical side (Eq. (10)). The crack length $\boldsymbol{s}$ and the crack

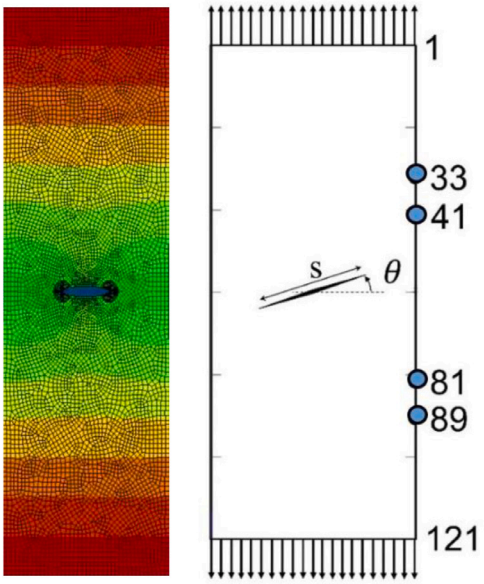


**Fig. 6.** Cracked specimen.

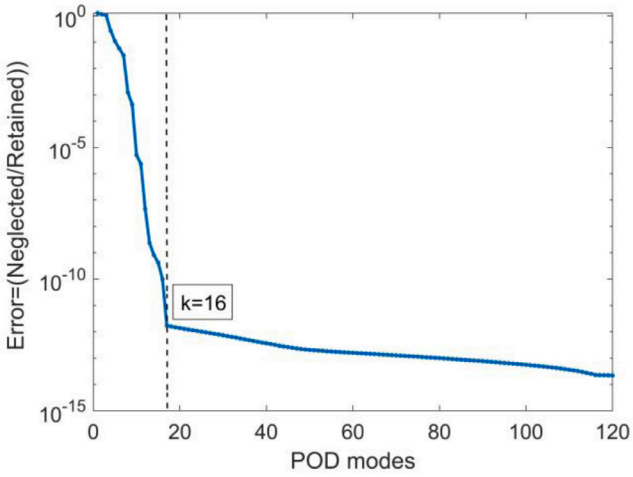


**Fig. 7.** The truncation point on the Elastostatic model.

orientation $\theta$ are the parameters that are subject to identification within this interval:

$$0 \text{ mm} \leq s \leq 0 \text{ mm} \leq 5 \text{ mm} \quad \text{and} \quad 0° \leq \theta \leq 60°. \tag{19}$$

The reduced model is constructed using the boundary displacement measured from 111 FE simulations, with length and orientation values of crack obtained by dividing the intervals of (Eq. (18)) into steps of 6° for $\theta$ and 0.5 mm for $s$. Plus, one case representing structure without crack.

We truncate the reduced basis model using the first 16 vectors, at the point where the ratio between the eigenvalues of the neglected vectors and the retained ones becomes less than $10^{-11}$, as shown in Fig. 7. Fig. 8 compares the POD-RBF predictions with the FE boundary displacements.

Fig. 6 report the error between both boundary displacements in each boundary node. For a crack with a length of 3.3 mm and with an orientation of 37.5°. This figure highlights a good agreement between the model predictions and FE results, except around the middle of the plate, around node 62, where appears a sudden divergence. We see this discrepancy because the node 62 at the center line, where displacements are close to zero independently from the loading level. This effect makes the error values that can be considered small appear very large in extremely small-displacement values.

### 4.3. Fitness evaluation based on all boundary data

10 test cracks are studied (C1 to C10) with different sizes and orientations. To solve this inverse problem, YA is run 20 times for each test crack, using a population of 30, and the maximum number of iterations is 100. Table 1 present the results in the form of an average over the 20 runs.

Figs. 9 displays the example of fitness convergence and the convergence of the crack parameters for the cases C2 (Table 1). In Fig. 9(a) at the 40th iteration, the function $f(P)$ Reach a value of $\sim 1.0^{-4}$. While Fig. 9(b) highlights that the predicted crack length converged toward a solution near the exact value after the 40th iteration. However, the crack orientation in Fig. 9(c) converged toward a solution close to the exact value after the 70th iteration.

The test cracks cover the different sizes and orientations, with cases of the same size and variable orientations. Also, cases with the same orientations and variable size. The results in Table 1. present error as the absolute difference between the actual and predicted crack. The maximum error for crack size is 0.61 mm in the case of the largest crack. As well as for crack orientation is 3.94° also in the case of largest orientation.

### 4.4. Fitness evaluation based on noisy data

We study the sensitivity of the crack identification approach against measurement uncertainty. We simulate it by introducing white Gaussian noise to the input displacement vector. Four levels of 1%, 2%, 3%, 5%, and 10%.

Fig. 10 present the average errors across all crack cases as a function of the noise level. The error here is the absolute difference between the actual crack parameter and the predicted ones. This figure shows that errors increase with the rise of the noise levels. However, one can note that the effect on crack orientation prediction is more significant, reaches 40% for the highest noise level of 10%. These results suggest that the approach proposed in this work shows higher stability for crack size prediction than crack orientation. However, the 5% and 10% noises are far more extensive than measurement uncertainty tolerance levels in the industry.

In reasonable uncertainty levels of 1% and 2%, the results slightly increase the crack size and orientation error. It is noticeable that the orientation errors exceed size errors beyond 3% noise. This is because variation in crack orientation reflects on the displacement at a smaller number of boundary nodes than the variation in crack length. Therefore, it is harder to distinguish between boundary displacements corresponding to different crack orientations from the inverse problem perspective.

### 4.5. Identification using a small number of sensor points

The results discussed above were determined using the displacements of all boundary nodes of the specimen. This section discusses the results obtained using only four sensor points represented in Fig. 6 by circles. The nodes 33, 41, 81, and 89 are located at 22 mm, 20 mm, 10 mm, and 8 mm. Under a noise level of 2%. Table 2. present the results. They showed that reducing the sensor point number to 4 did not globally affect the identification performance of the model.

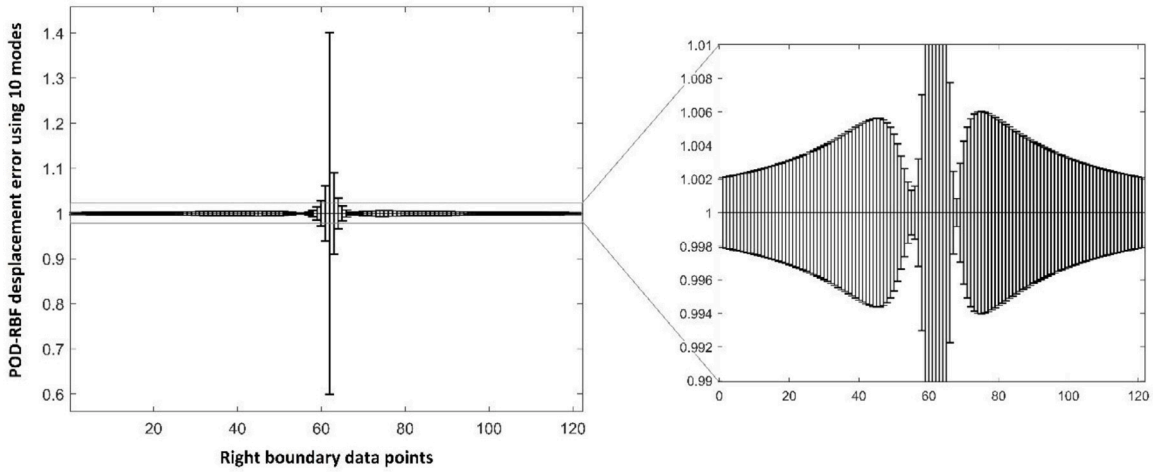


**Fig. 8.** The error of POD-RBF boundary displacement.

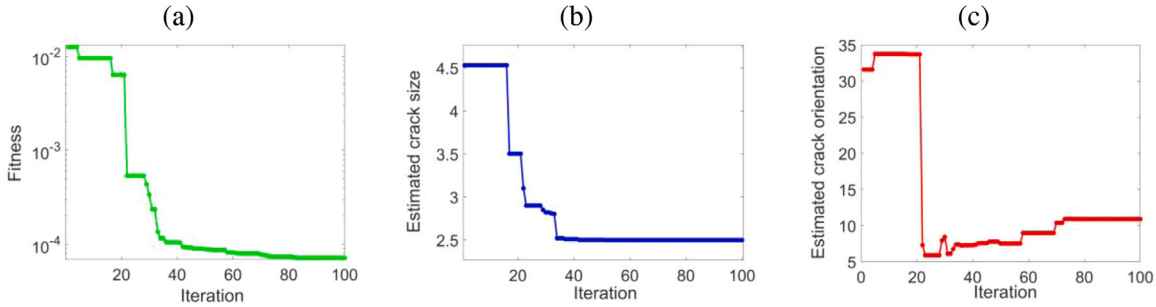


**Fig. 9.** C2 Crack identification.

**Table 1**
Identified cracks using boundary displacement at all boundary nodes.

| Studied cases | Real crack | | Estimated crack | | Error | |
|---|---|---|---|---|---|---|
| | $s$ (mm) | $\theta$ (°) | $s$ (mm) | $\theta$ (°) | $s$ (%) | $\theta$ (%) |
| **C1** | 1.2 | 10.0 | 1.33 | 10.18 | 10.83 | 1.8 |
| **C2** | 2.5 | 10.0 | 2.06 | 10.06 | 17.6 | 0.6 |
| **C3** | 3.3 | 10.0 | 3.57 | 9.43 | 8.18 | 5.7 |
| **C4** | 4.9 | 10.0 | 4.97 | 10. 35 | 1.42 | 3.5 |
| **C5** | 1.2 | 15.0 | 1.32 | 14.22 | 10 | 5.2 |
| **C6** | 1.2 | 37.5 | 1.34 | 35.29 | 11.6 | 5.89 |
| **C7** | 3.3 | 17.0 | 3.51 | 14.27 | 6.36 | 16.05 |
| **C8** | 3.3 | 37.5 | 3.19 | 37.49 | 3.33 | 0.02 |
| **C9** | 3.3 | 60.0 | 3.12 | 56.06 | 5.45 | 6.56 |
| **C10** | 5.0 | 37.5 | 4.39 | 33.88 | 12.2 | 9.65 |

**Table 2**
Identified cracks using boundary displacement at four nodes.

| Studied cases | Real crack | | Estimated crack | | Error | |
|---|---|---|---|---|---|---|
| | $s$ (mm) | $\theta$ (°) | $s$ (mm) | $\theta$ (°) | $s$ (%) | $\theta$ (%) |
| **C1** | 1.2 | 10 | 1.51 | 9.74 | 25.83 | 2.6 |
| **C2** | 2.5 | 10 | 2.99 | 9.06 | 19.6 | 9.4 |
| **C3** | 3.3 | 10 | 3.51 | 10.88 | 6.36 | 8.8 |
| **C4** | 4.9 | 10 | 4.94 | 10.15 | 0.81 | 1.5 |
| **C5** | 1.2 | 15 | 1.49 | 14.85 | 24.16 | 1 |
| **C6** | 1.2 | 37.5 | 1.30 | 36.94 | 8.33 | 1.49 |
| **C7** | 3.3 | 17 | 3.44 | 18.04 | 4.24 | 6.11 |
| **C8** | 3.3 | 37.5 | 3.11 | 38.06 | 5.75 | 1.49 |
| **C9** | 3.3 | 60 | 3.16 | 58.48 | 4.242 | 2.53 |
| **C10** | 5 | 37.5 | 4.52 | 36.06 | 9.6 | 3.84 |

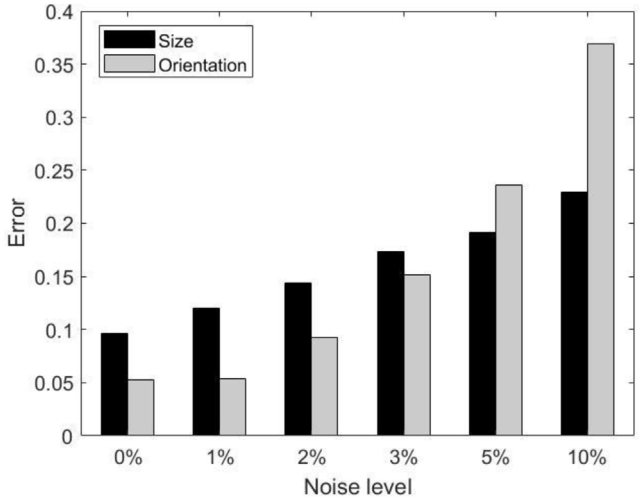


**Fig. 10.** Effect of noise level on the average prediction error for all cracks.

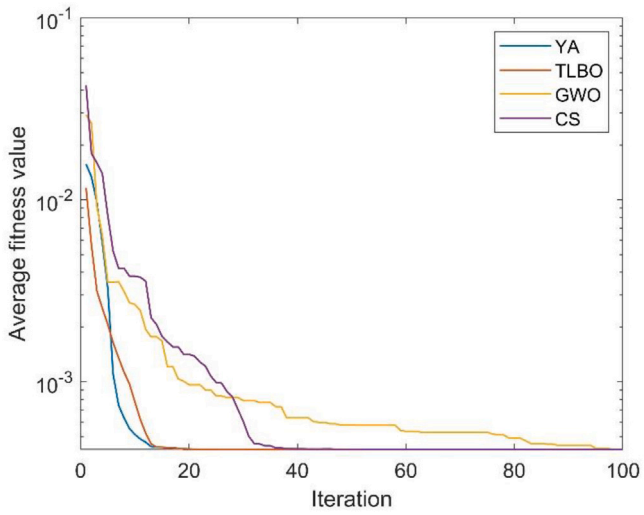


**Fig. 11.** Average fitness convergence by YA, TLBO, GWO, and CS algorithms in C8.

### 4.6. Optimization algorithms performances

In iterative identification problems, the computational time is an issue because of the repetitive nature of the computation. Fitness evaluation is the most significant contributor to computational cost. However, since the evaluation time is considered invariable for the same problem, the optimization convergence speed becomes an important issue. Therefore, the efficiency of the identification calculation highly depends on the chosen optimization algorithm.

CS is a well-known search algorithm inspired by the parasitic reproduction strategy of cuckoo birds. They lay their eggs in the nests of the host's birds, who may discover that and either destroy the egg or abandon the nest [42]; in the formulation of this algorithm, the number of available host nests is constant. Thus, a host can discover an alien egg with a probability of Pa between 0 and 1.

The TLBO is a parameter-less algorithm that mimics teaching and learning abilities in a classroom [43], with two primary stages in each iteration. The teacher phase, where the best solution influences all population members. And on the other hand, there is the learning phase. Here the fitnesses of individuals are compared with each other. GWO is a widely used optimization algorithm inspired by the hunting behavior of a gray wolf pack, and it simulates the hierarchical system by ranking by fitness values.

In this case, we consider YA, GWO, and TLBO along with CS using the data issued from the four sensor points described earlier, with a 2% noise level. For the sake of performance comparison, each algorithm performs 20 runs, with a population size of 30 and a maximum number of iterations equal to 100. For CS, the probability Pa is chosen equal to 0.25. Fig. 10 depicts the average fitness convergence for these algorithms in the case of crack C8 (S = 3.3 mm, $\theta = 37.5°$). And Table 3 report the analysis of the results.

Fig. 11 shows that although all algorithms converge to the same fitness value, YA and TLBO algorithms converge rapidly, around the 20th iteration, relatively to CS and GWO that converge before the 40 and 80 iterations, respectively.

We note that in terms of identification quality, the results show that the crack size and orientation can be estimated precisely by all algorithms, despite the noise in the data issued and the low number of sensors. However, YA and TLBO algorithms provided quick convergence to the optimum, which means fewer iterations can be sufficient. Furthermore, YA and GWO algorithms are advantageous in computational time because both these algorithms are based on very few steps and require a single evaluation in each iteration.

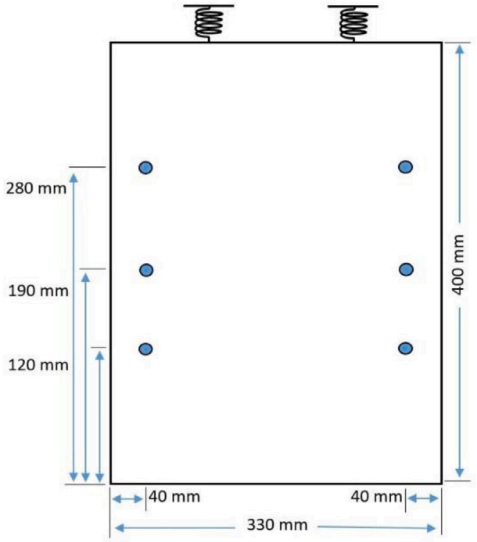


**Fig. 12.** Plate dimensions and accelerometer positions.

## 5. Identification using dynamic analysis

### 5.1. Experimental setup

This section expands on the study made in Ref. [41] using experimental data issues from a frequency response of a 400 mm by 330 mm metallic plate, shown in Fig. 12. Young's modulus, 2.25 1011 (N/M2), the Poisson's ratio 0.3, the density of 7850 kg/m3, and the thickness of 2 mm.

6 PCB Accelerometers (356A15), excited with a PCB hammer (086C03) to measure the vibration. Fig. 12 shows the positions of the six accelerometers. Fig. 13 shows the experimental setup.

### 5.2. POD-RBF based on experimental data

We consider in this study three crack orientations. The horizontal crack (0°) and the crack at an angle of 45° start from the center of the left side. On the other hand, the vertical crack (90°) starts from the center of the top side of the plate. In this study, the vibrational response is considered for crack varying between 0 (no crack) and 80 mm with a step of 10 mm.

**Table 3**
Performance of YA, TLBO, GWO, and CS algorithms in case C8 using four sensor points under 2% noise.

| | Average time (s) | Best fitness value | The standard deviation of the fitness | Best crack size mm (error %) | Best crack orientation ° (error %) |
|---|---|---|---|---|---|
| YA | 15.54 | $4.251\ 10^{-4}$ | $1.34\ 10^{-17}$ | 3.10 (6.06) | 37.40 (0.26) |
| TLBO | 31.72 | $4.251\ 10^{-4}$ | $2.81\ 10^{-17}$ | 3.10 (6.06) | 37.40 (0.26) |
| GWO | 15.63 | $4.254\ 10^{-4}$ | $1.01\ 10^{-8}$ | 3.10 (6.06) | 37.06 (1.17) |
| CS | 30.23 | $4.252\ 10^{-4}$ | $4.87\ 10^{-7}$ | 3.10 (6.06) | 37.62 (0.32) |

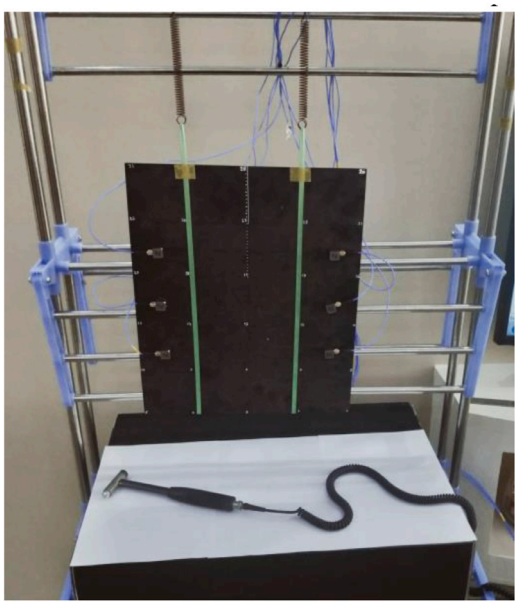

**Fig. 13.** Experimental setup.

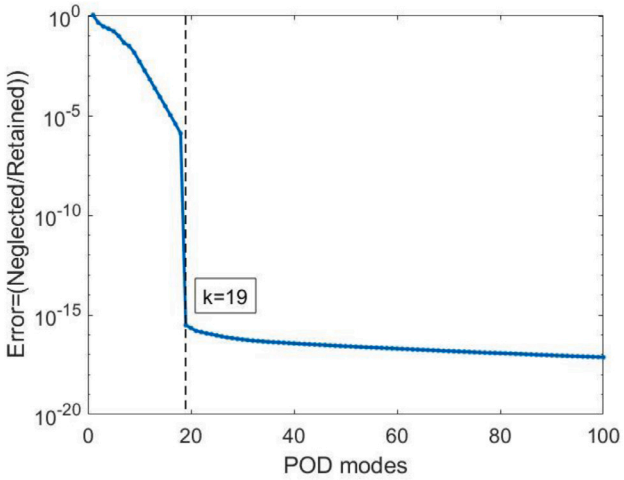


**Fig. 14.** The truncation point on the dynamic model.

Using the Frequency Response Function (FRF) measured for 27 cracks, a POD-RBF model is constructed and truncated with 19 vectors. In this point, the ratio between the neglected eigenvalues and the retained ones is less than $10^{-10}$. As shown in Fig. 14. Fig. 15 shows the experimental FRF at crack length 25 mm and orientation of 45°, compared to the predicted FRF calculated by the POD-RBF model. This figure highlights a good agreement between the model predictions and experimental measurement at modes positions.

### 5.3. Fitness evaluation base on experimental data

We consider in this study seven crack cases. To solve this inverse problem, the YA is run 20 times for each test crack to minimize Eq. (9) using a population of 30, the maximum number of iterations is 100. Table 4 summarizes the results in the form of an average over the 20 runs.

Table 4 presented the error as the absolute difference between the actual and predicted crack. The results show that the maximum error for size prediction is 2.76 mm, and the orientation is 0.43°.

An example of fitness convergence and the convergence of the crack parameters is presented in Fig. 16 for cases C14 (Table 4). In Fig. 16(a), The fitness value $f(P)$ Reach a value of $1.0\ 10^{-3}$ before the 20th iteration. For example, Fig. 16(b) and Fig. 17(c) show that crack size and orientation converged to a solution close to the exact value in 20 iterations.

### 5.4. Optimization algorithms performances

We compare the performance of the four algorithms YA, GWO, TLBO, and CS in this section. Each algorithm performs 20 runs, where the population size is 30, and the maximum number of iterations is 100. Fig. 17 depicts the average fitness convergence for these algorithms in the case of crack C14 (S = 75 mm, $\theta$ = 45°). And Table 5 report the analysis of the results. Table 6, on the other hand, report the performances of the algorithms in the case C11, where the error is high.

Fig. 17 shows that although all algorithms converge to the same fitness value, YA and TLBO algorithms converge rapidly, around the 30th iteration, relatively to CS that converge before the 50th and GWO converged at 100 iterations.

The results show that the crack size and its orientation can be estimated precisely by all algorithms in terms of identification quality. However, YA and TLBO algorithms provided quick convergence, with an advantage over TLBO. Finally, we should mention that TLBO uses twice the number of evaluations in each iteration as opposed to the other algorithms. Thus it spends double the computational cost on evaluation. This observation is apparent in Table 5 results. Results in Table 6 suggest that the high error found earlier in Table 4 is due to the limitation of the POD-RBF model. As the model is built using a limited set of experimental results, POD-RBF estimation accuracy is lower when cases near the boundaries of the known data, like in the case of C11 and the case C15, that correspond to 0° and 90° respectively.

## 6. Conclusion

We have presented in this paper a new metaheuristic algorithm and applied it to the study of inverse crack identification based on the POD-RBF technique. We considered two studies, namely, Elastostatic based on FEM analysis and a vibrational study based on experimental data. The method is effective compared to recently published algorithms, such as GWO, CS, and TLBO.

Moreover, we have shown that the presented method is stable concerning noisy input data, mimicking the measurement uncertainty, and limited sensors.

We compare the performance of the suggested algorithm to GWO, CS, and TLBO algorithms. The results indicate that these algorithms can efficiently identify the size and orientation of the crack accurately, with YA having an advantage in terms of computation time.

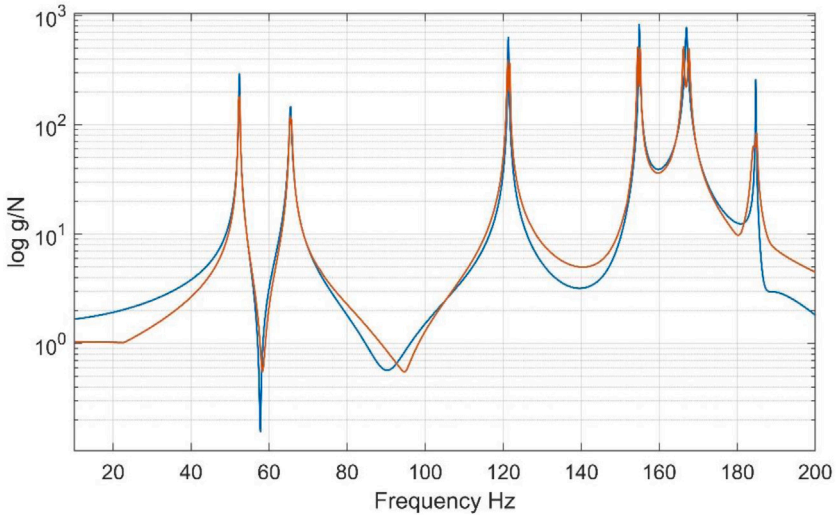


**Fig. 15.** Experimental FRF from 25 mm crack at 45° vs. predicted using POD-RBF.

**Table 4**
Identified cracks based on the experimental dynamic study.

| Studied cases | Real crack | | Estimated crack | | Error | |
|---|---|---|---|---|---|---|
| | $s$ (mm) | $\theta$ (°) | $s$ (mm) | $\theta$ (°) | $s$ (%) | $\theta$ (%) |
| **C11** | 25 | 0 | 22.24 | 0.032 | 11.04 | – |
| **C12** | 75 | 0 | 73.29 | $1.26\ 10^{-5}$ | 2.28 | – |
| **C13** | 25 | 45 | 24.89 | 45.05 | 0.44 | 0.11 |
| **C14** | 75 | 45 | 74.73 | 44.57 | 0.36 | 0.95 |
| **C15** | 10 | 90 | 7.710 | 89.99 | 22.9 | 0.01 |
| **C16** | 50 | 90 | 49.82 | 89.99 | 0.36 | 0.01 |
| **C17** | 75 | 90 | 74.05 | 89.94 | 1.26 | 0.06 |

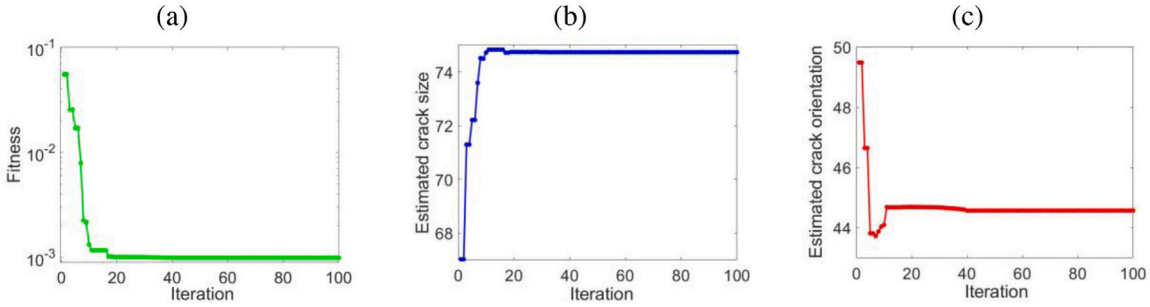


**Fig. 16.** C14 crack parameter identification versus iterations.

**Table 5**
Performance of YA, TLBO, GWO, and CS algorithms in case C14.

| | Average time (s) | Best fitness value | The standard deviation of the fitness | Best crack size mm (error %) | Best crack orientation ° (error %) |
|---|---|---|---|---|---|
| YA | 12.30 | 0.000985 | $2.4\ 10^{-15}$ | 74.73 (0.36) | 44.57 (0.95) |
| TLBO | 24.11 | 0.000985 | $8.44\ 10^{-15}$ | 74.73 (0.36) | 44.53 (0.95) |
| GWO | 12.72 | 0.001015 | $2.93\ 10^{-5}$ | 74.72 (0.37) | 44.57 (1.04) |
| CS | 23.96 | 0.000985 | $1.68\ 10^{-8}$ | 74.73 (0.36) | 44.57 (0.95) |

**Table 6**
Performance of YA, TLBO, GWO, and CS algorithms in case C11.

| | Average time (s) | Best fitness value | The standard deviation of the fitness | Best crack size mm (error %) | Best crack orientation ° |
|---|---|---|---|---|---|
| YA | 12.35 | 1.63 E−03 | 6.51 E−07 | 22.24 (11.04) | 0.03 |
| TLBO | 25.33 | 1.63 E−03 | 3.21 E−17 | 22.17 (11.32) | 0.00 |
| GWO | 12.49 | 1.63 E−03 | 4.56 E−10 | 22.18 (11.28) | 0.00 |
| CS | 24.68 | 1.63 E−03 | 2.79 E−09 | 22.17 (11.32) | 0.00 |

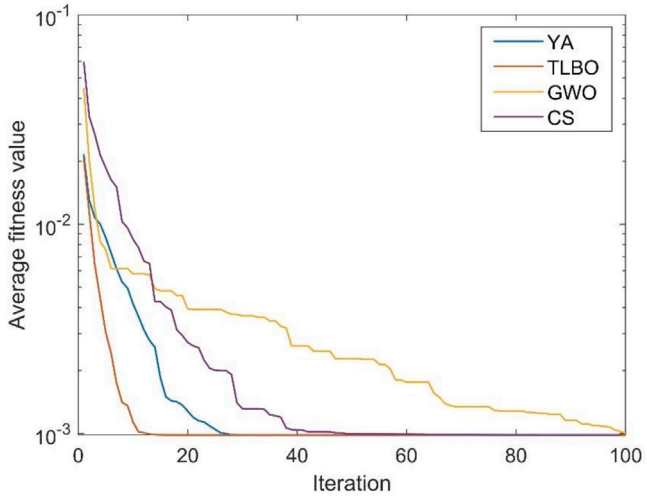


**Fig. 17.** Average fitness convergence by YA, TLBO, GWO, and CS algorithms in C14.

The future study combines the new algorithm YA with the local optimization method in crack detections using eddy current tomography. As well as investigate the performance of metaheuristic algorithms for robust minimization procedures based on DNN, as such methods provide higher flexibility in optimization and uncertainty evaluation and constitute frameworks to solve both inverse and forward problems.

## Replication of Results

The results presented in this article can replicate by implementing the data structures and algorithms presented in the article. All the references are reported in the paper, as well as the algorithm. In any case, the code (developed in MATLAB) is available upon request from the authors; the YUKI algorithm package is planned to be released after the acceptance of this paper.

## CRediT authorship contribution statement

**Brahim Benaissa:** Conceptualization, Methodology, Software, Validation. **Nourredine Aït Hocine:** Conceptualization, Investigation. **Samir Khatir:** Methodology, Validation, Resources. **Mohamed Kamel Riahi:** Investigation, Writing – review & editing. **Seyedali Mirjalili:** Writing – review & editing.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## References

[1] M.I. Friswell, Damage identification using inverse methods, Phil. Trans. R. Soc. A 365 (1851) (2007) 393–410.
[2] H.D. Bui, Fracture Mechanics: Inverse Problems and Solutions, Vol. 139, Springer Science & Business Media, 2007.
[3] G.E. Stavroulakis, Inverse and Crack Identification Problems in Engineering Mechanics, Vol. 46, Springer Science & Business Media, 2013.
[4] S. Mellings, M. Aliabadi, Flaw identification using the boundary element method, Internat. J. Numer. Methods Engrg. 38 (3) (1995) 399–419.
[5] C. Du, et al., Dynamic XFEM-based detection of multiple flaws using an improved artificial bee colony algorithm, Comput. Methods Appl. Mech. Engrg. 365 (2020) 112995.
[6] D. Rabinovich, D. Givoli, S. Vigdergauz, Crack identification by 'arrival time'using XFEM and a genetic algorithm, Internat. J. Numer. Methods Engrg. 77 (3) (2009) 337–359.
[7] H. Haddar, Z. Jiang, M.K. Riahi, A robust inversion method for quantitative 3D shape reconstruction from coaxial eddy current measurements, J. Sci. Comput. 70 (1) (2017) 29–59.
[8] H. Haddar, M. Riahi, Near-field linear sampling method foraxisymmetric eddy current tomography, 2021, arXiv preprint arXiv:2103.01207.
[9] S. Nanthakumar, et al., Detection of material interfaces using a regularized level set method in piezoelectric structures, Inverse Probl. Sci. Eng. 24 (1) (2016) 153–176.
[10] C. Alessandri, V. Mallardo, Crack identification in two-dimensional unilateral contact mechanics with the boundary element method, Comput. Mech. 24 (2) (1999) 100–109.
[11] N. Amoura, et al., Axisymmetric and two-dimensional crack identification using boundary elements and coupled quasi-random downhill simplex algorithms, Eng. Anal. Bound. Elem. 34 (6) (2010) 611–618.
[12] C.G. Vossou, I.N. Koukoulis, C.G. Provatidis, Genetic combined with a simplex algorithm as an efficient method for the detection of a depressed ellipsoidal flaw using the boundary element method, Int. J. Appl. Math. Comput. Sci. 4 (2) (2007).
[13] T. Burczynski, W. Beluch, The identification of cracks using boundary elements and evolutionary algorithms, Eng. Anal. Bound. Elem. 25 (4) (2001) 313–322.
[14] G. Hattori, A. Sáez, Damage identification in multifield materials using neural networks, Inverse Probl. Sci. Eng. 21 (6) (2013) 929–944.
[15] S. Khatir, M.A. Wahab, Fast simulations for solving fracture mechanics inverse problems using POD-RBF XIGA and Jaya algorithm, Eng. Fract. Mech. 205 (2019) 285–300.
[16] K. Samir, et al., Genetic algorithm based objective functions comparative study for damage detection and localization in beam structures, J. Phys. Conf. Ser. (2015) IOP Publishing.
[17] S. Nanthakumar, T. Lahmer, T. Rabczuk, Detection of multiple flaws in piezoelectric structures using XFEM and level sets, Comput. Methods Appl. Mech. Engrg. 275 (2014) 98–112.
[18] L. Nguyen-Tuan, S.S. Nanthakumar, T. Lahmer, Identification of multiple flaws in dams using inverse analysis based on hydro-mechanical XFEM and level sets, Comput. Geotech. 110 (2019) 211–221.
[19] M. Alalade, et al., Damage identification in gravity dams using dynamic coupled hydro-mechanical XFEM, Int. J. Mech. Mater. Des. 14 (1) (2018) 157–175.
[20] A. Chatterjee, An introduction to the proper orthogonal decomposition, Current Sci. 78 (7) (2000) 808–817.
[21] B. Benaissa, et al., Application of proper orthogonal decomposition and radial basis functions for crack size estimation using particle swarm optimization, J. Phys. Conf. Ser. (2017) IOP Publishing.
[22] K. Samir, et al., Damage detection in CFRP composite beams based on vibration analysis using proper orthogonal decomposition method with radial basis functions and cuckoo search algorithm, Compos. Struct. 187 (2018) 344–353.
[23] C. Winton, et al., Application of proper orthogonal decomposition (POD) to inverse problems in saturated groundwater flow, Adv. Water Resour. 34 (12) (2011) 1519–1526.
[24] Z. Ostrowski, R. Białecki, A. Kassab, Solving inverse heat conduction problems using trained POD-RBF network inverse method, Inverse Probl. Sci. Eng. 16 (1) (2008) 39–54.
[25] J.C. Brigham, W. Aquino, Inverse viscoelastic material characterization using POD reduced-order modeling in acoustic–structure interaction, Comput. Methods Appl. Mech. Engrg. 198 (9) (2009) 893–903.
[26] C.A. Rogers, et al., An inverse POD-RBF network approach to parameter estimation in mechanics, Inverse Probl. Sci. Eng. 20 (5) (2012) 749–767.
[27] S. Wang, S. Khatir, M.A. Wahab, Proper Orthogonal Decomposition for the prediction of fretting wear characteristics, Tribol. Int. 152 (2020) 106545.
[28] V. Buljak, G. Maier, Proper orthogonal decomposition and radial basis functions in material characterization based on instrumented indentation, Eng. Struct. 33 (2) (2011) 492–501.
[29] K. Hoang, et al., Rapid identification of material properties of the interface tissue in dental implant systems using reduced basis method, Inverse Probl. Sci. Eng. 21 (8) (2013) 1310–1334.
[30] D.S. Broomhead, D. Lowe, Radial basis functions, multi-variable functional interpolation and adaptive networks, 1988, DTIC Document.
[31] S. Levasseur, et al., Statistical inverse analysis based on genetic algorithm and principal component analysis: method and developments using synthetic data, Int. J. Numer. Anal. Methods Geomech. 33 (12) (2009) 1485–1511.
[32] G. Bolzon, V. Buljak, An indentation ]based technique to determine in ]depth residual stress profiles induced by surface treatment of metal components, Fatigue Fract. Eng. Mater. Struct. 34 (2) (2011) 97–107.
[33] G. Bolzon, et al., Assessment of elastic–plastic material parameters comparatively by three procedures based on indentation test and inverse analysis, Inverse Probl. Sci. Eng. 19 (6) (2011) 815–837.
[34] V. Buljak, Inverse Analyses with Model Reduction: Proper Orthogonal Decomposition in Structural Mechanics, Springer, 2012.
[35] G. Bolzon, V. Buljak, An effective computational tool for parametric studies and identification problems in materials mechanics, Comput. Mech. 48 (6) (2011) 675–687.
[36] P. Druault, P. Guibert, F. Alizon, Use of proper orthogonal decomposition for time interpolation from PIV data, Exp. Fluids 39 (6) (2005) 1009–1023.

[37] M. Bocciarelli, et al., An inverse analysis approach based on a POD direct model for the mechanical characterization of metallic materials, Comput. Mater. Sci. 95 (2014) 302–308.
[38] C. Anitescu, et al., Artificial neural network methods for the solution of second order boundary value problems, Comput. Mater. Contin. 59 (1) (2019) 345–359.
[39] D. Rabinovich, D. Givoli, S. Vigdergauz, XFEM - based crack detection scheme using a genetic algorithm, Internat. J. Numer. Methods Engrg. 71 (9) (2007) 1051–1080.
[40] B. Benaissa, et al., Crack identification using model reduction based on proper orthogonal decomposition coupled with radial basis functions, Struct. Multidiscip. Optim. 54 (2) (2016) 265–274.
[41] S. Khatir, et al., Improved ANN technique combined with jaya algorithm for crack identification in plates using XIGA and experimental analysis, Theor. Appl. Fract. Mech. 107 (2020) 102554.
[42] X.-S. Yang, S. Deb, Engineering optimisation by cuckoo search, Int. J. Math. Model. Numer. Optim. 1 (4) (2010) 330–343.
[43] R. Rao, V. Patel, An elitist teaching-learning-based optimization algorithm for solving complex constrained optimization problems, Int. J. Ind. Eng. Comput. 3 (4) (2012) 535–560.
[44] S. Mirjalili, S.M. Mirjalili, A. Lewis, Grey wolf optimizer, Adv. Eng. Softw. 69 (2014) 46–61.
[45] B. Benaissa, et al., Crack identification using model reduction based on proper orthogonal decomposition coupled with radial basis functions, Struct. Multidiscip. Optim. (2016) 1–10.
[46] M. Bonnet, A. Constantinescu, Inverse problems in elasticity, Inverse Problems 21 (2) (2005) R1.

**Brahim Benaissa**, Ph.D. in computational mechanics, and works on applied computational methods in engineering. Suggested methods in structural health monitoring, and developped experince in human activity recognition and indoor localization at Kyushu Institute of Technology. He has a strong interest in metaheuristics and artificial intelligence. Joined Toyota technological Institute in 2021 to investigate application in the field of affective design.