AIAA JOURNAL
Vol. 40, No. 9, September 2002

Damage Detection for Composite Plates Using Lamb Waves and Projection Genetic Algorithm

Y. G. Xu,* G. R. Liu,† and Z. P. Wu‡
National University of Singapore, Singapore 119260, Republic of Singapore

A method of damage detection for composite plates using Lamb waves and a projection genetic algorithm is investigated in this study. This method first formulates the damage detection as an optimization problem of minimizing the error between the measured and calculated surface displacement response derived from Lamb waves. The calculated response is obtained by using the strip element method with the trial crack parameters (location, size). Then a projection genetic algorithm is used to solve the optimization problem and thus identify the actual crack parameters. The projection genetic algorithm is developed from the hybridization of the modified micro-genetic algorithm with a projection operator. It has an excellent convergence performance, taking only 9.9 ~ 14.2% of the number of function evaluations required by the conventional micro-genetic algorithm to obtain the global optima for six benchmark functions. Numerical examples are presented to verify the proposed method for detection of cracks inside composite plates. The maximal error of detected crack parameters is − 4.3% for four simulated cases, which is achieved by running only 60 generations of projection genetic algorithm.

Nomenclature

$\boldsymbol{A}_0, \boldsymbol{A}_1, \boldsymbol{A}_2, \boldsymbol{M}$	=	coefficient matrices
a_c, d_c, l_c	=	horizontal location, depth, and length of crack, respectively
c	=	better individual of c_1 and c_2
c_1, c_2	=	individuals obtained from local optimizer
$\boldsymbol{D}_k, d_{ij}^k$	=	elastic constant matrix and elastic constants for kth element, respectively
$E(a_c, d_c, l_c)$	=	error of displacement response, also used as fitness function
$f_i(a_c, d_c, l_c), f_i^m$	=	calculated and measured surface displacement responses at ith point, respectively
$f(\cdot)$	=	fitness function or objective function
H, m	=	thickness of plate and number of laminates, respectively
i_{dum}	=	initial random number seed
i_p	=	position of gene that is different from that in chromosome under comparison
$\boldsymbol{K}$	=	stiffness matrix
$\boldsymbol{L}$	=	differential operator matrix
N	=	population size
N_p	=	number of points at surface of plate
n	=	number of parameters of each individual
n_{gj}	=	number of genes in jth substring corresponding to jth parameter
$n_{\text{pGA}}, n_{\text{mGA}}, n_{\text{hGA}}$	=	numbers of generations required to obtain global optimum when using pGA, mGA, and hGA, respectively
$\bar{n}_{\text{pGA}}, \bar{n}_{\text{hGA}}, \bar{n}_{\text{mGA}}$	=	means of $n_{\text{pGA}}, n_{\text{mGA}}, n_{\text{hGA}}$, respectively
$\boldsymbol{P}(j), \boldsymbol{C}(j), \boldsymbol{C}_h(j)$	=	population of individuals, offspring, and updated offspring at jth generation, respectively.
$p_{\text{cross}}, p_{\text{mutate}}$	=	possibilities of crossover and mutation operation, respectively
$p_{\text{ji}}, c_{\text{ji}}, i = 1, \ldots, N$	=	ith individual and ith offspring at jth generation, respectively
p_j^b, p_j^s, p_{j-1}^b	=	best individual, second best individual at jth generation and best individual at $(j-1)$th generation, respectively
$\boldsymbol{Q}, \boldsymbol{V}$	=	vectors of external load amplitude and displacement amplitude, respectively
$q(t), q_0, \omega$	=	time-harmonic load, load amplitude and frequency, respectively
$\boldsymbol{R}, \boldsymbol{S}$	=	vectors of stress and equivalent external force, respectively
$R_{\text{mGA}}, R_{\text{hGA}}$	=	ratios; $R_{\text{hGA}} = (5 \times \bar{n}_{\text{pGA}})/(7 \times \bar{n}_{\text{hGA}})$, $R_{\text{mGA}} = (5 \times \bar{n}_{\text{pGA}})/(7 \times \bar{n}_{\text{mGA}})$
t	=	time
$\boldsymbol{U}_k$	=	vector of displacements at kth element
$u(x, z), w(x, z)$	=	displacements in x and z direction, respectively
x, y, z	=	coordinate system for composite plate
α, β	=	coefficients of extrapolation and interpolation, respectively
γ	=	threshold to define population convergence
$\delta(\tau)$	=	Dirac's impulse function
$\eta, \Gamma(x)$	=	noise length and noise at point x, respectively

Received 13 February 2001; revision received 15 June 2001; accepted for publication 9 March 2002.

*SMA Research Fellow, Center for Advanced Computations in Engineering Science, Department of Mechanical and Production Engineering, 10 Kent Ridge Crescent.

†SMA Fellow, Center for Advanced Computations in Engineering Science, Department of Mechanical and Production Engineering, 10 Kent Ridge Crescent; mpeliugr@nus.edu.sg.

‡Senior Research Engineer, Center for Advanced Computations in Engineering Science, Department of Mechanical and Production Engineering, 10 Kent Ridge Crescent.

I. Introduction

DAMAGE detection of composite structures has become a very active topic in recent years.[1] Varied detection techniques have been developed so far,[1,2] among which elastic waves based methods have been receiving the increasing attention.[3] A lot of work has been carried out to reveal the relationship between the character of elastic wave scattering and the location and size of cracks inside the composite plates. For example, Karunasena et al.[4] studied the plane-strain-wave scattering by cracks and obtained the dispersion equations for Lamb waves. Datta et al.[5] investigated the scattering of Lamb waves caused by cracks and flaws in plates. Karim et al.[6] did the similar work by combining the finite element method and guided wave mode expansions. Liu et al.[7] studied the transient scattering of Lamb waves by a surface-breaking crack using both analytical and

experimental methods. Liu and colleagues[8,9] and Lam et al.[10] investigated the characterization of both horizontal and vertical cracks in the composite plates subjected to a moving or fixed source load. All of these works have shown that the scattered elastic waves are really related to the location and size of cracks inside the composite plates. Thus, if a scattered elastic wave field can be experimentally measured, detection of the damages in composite plates can be transferred as an optimization problem of finding a set of crack parameters (location, size) that yields the best match with the measured elastic wave field.

As the relationship between the crack parameters and the scattered elastic wave field is hardly possible to express in an explicit form for such complex a dynamics problem of wave propagation, the search space of crack parameters is generally highly nonlinear, and multimodal, advanced optimization algorithms are thus critical to solve this complex problem and obtain the corresponding detection results.

Hybrid genetic algorithms (GAs) have been known as a powerful tool to solve the nonlinear, multimodal, and nondifferentiable problems in recent years.[11] A genetic algorithm is good at global search but slow to converge, whereas a local search is good at fine tuning of the solution but often falls into local optima. The hybrid GAs of combining the conventional GAs and local search techniques can compensate their individual weak points and outperform either one individually. With the development of varied hybrid GAs,[11–13] the present authors have proposed an effective algorithm called an hGA.[14] This algorithm is derived from the hybridization of a modified micro-genetic algorithm (mGA) with a local optimizer. The "micro" means this genetic algorithm uses only a very small population size (typically $5 \sim 8$).[15] The local optimizer is one of direct search techniques. It finds the even better solutions by jumping along the move direction of best individual. One individual in the GAs means a possible solution of optimization problem, that is, a set of crack parameters for the present detection problem. Test functions and application examples have shown that this hGA has an excellent convergence performance over the conventional GAs without incorporating with the local optimizer and also has a good computational efficiency when compared with the other hybrid GAs.[14] Nevertheless, there is likely a situation in which this hGA does not perform particularly well. That is, when the individual p_j^b is identical to p_{j-1}^b at the jth generation in the evolution process the local optimizer fails to find the new individuals different from those in the $\boldsymbol{C}(j)$. This would decrease the population diversity and also increase the unnecessary evaluations for the same individuals in one generation.

In this study, improvements for the previously proposed hGA are made to overcome these problems and further increase the searching efficiency of algorithm, including the following:

1) The local optimizer is improved by using an alternative way to construct the move direction of beset individual so that it can find the even better individuals different from those in the $\boldsymbol{C}(j)$ with a significantly increased possibility. That is, the move direction of best individual is made by using either p_j^b and p_{j-1}^b or p_j^b and p_j^s when p_j^b is identical to p_{j-1}^b.

2) Only the best one of two new individuals obtained from the local optimizer is used to replace the worst individual in the current $\boldsymbol{C}(j)$ to implement the hybridization process. This is obviously beneficial to avoiding the decrease of population diversity caused by the insertion of two new individuals that are close to each other and also beneficial to decreasing the population size.

3) Mutation operation is employed in the evolution process to increase the population diversity. This is especially beneficial for the case the improved local optimizer fails to find the new individuals different from those in the $\boldsymbol{C}(j)$ when p_j^b, p_{j-1}^b, and p_j^s are all identical. This new hybrid GA is termed as a projection GA (pGA) for the improved local optimizer is taken as a projection operator in the evolution process. Performance test using six benchmark functions has shown that this pGA has excellent convergence performance over the conventional mGA as well as the previous hGA.

Detection method based on the use of this pGA is verified by numerical examples. It has been found that this method is very effective and efficient for the damage detection of composite plates.

II. Formulation of Damage Detection as Optimization Problem

A. Statement of the Problem

Figure 1 shows a composite plate under investigation. It is assumed to occupy the region $-\infty \le (x, y) \le \infty$ and $-H/2 \le z \le H/2$. A horizontal or vertical crack is hidden inside this plate, whereby its location and size are defined by three parameters a_c, d_c, and l_c. The crack is considered to be infinite in the y direction to simplify the problem into a two-dimensional problem.

A time-harmonic load $q(t)$ is applied on the upper surface of plate in order to excite the elastic waves propagated inside this plate. This load is also assumed not to vary in the y direction. It is expressed as

$$q(t) = q_0 e^{-i\omega t} \tag{1}$$

The surface displacement response consists of a number of Lamb waves scattered by the hidden cracks.[8] It carries the information about the location and size of crack and thus can be used as a feature information for damage detection.

B. Objective Function

With the use of well-known error norms,[16] the crack detection problem is formulated as an optimization problem with the following objective function:

$$\min E(a_c, d_c, l_c) = \sum_{i=1}^{N_p} \left| f_i(a_c, d_c, l_c) - f_i^m \right| \qquad i = 1, \ldots, N_p \tag{2a}$$

or

$$\min E(a_c, d_c, l_c) = \left[\sum_{i=1}^{N_p} \left| f_i(a_c, d_c, l_c) - f_i^m \right|^2 \right]^{\frac{1}{2}} \qquad i = 1, \ldots, N_p \tag{2b}$$

s.t.

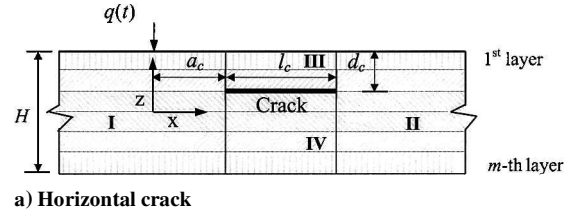
$$0 \le a_c \le 10H, \qquad 0 \le d_c \le H, \qquad 0 \le l_c \le 10H$$

for horizontal crack

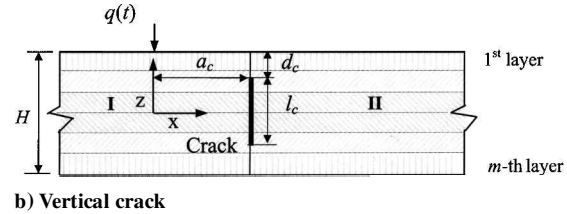
$$0 \le a_c \le 10H, \qquad 0 \le d_c \le H, \qquad 0 \le l_c \le H$$

for vertical crack

Comparatively, Eq. (2a) is more sensitive to the spatial distribution of data, whereas Eq. (2b) is more stability in noisy environment.[16] Both of them mean that if a set of crack parameters, a_c, d_c, and l_c, can generate the displacement response very close to the measured value, this set of crack parameters can be taken as the actual crack



a) Horizontal crack



b) Vertical crack

Fig. 1 An m-layer composite plate with a embedded crack.

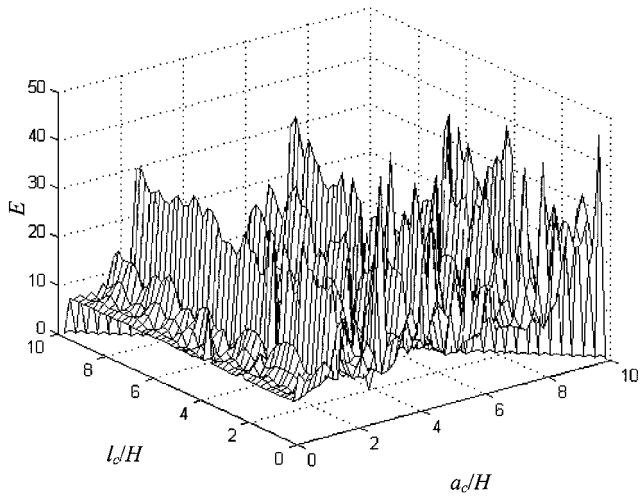


Fig. 2 Search space for two crack parameters within domain $a_c + l_c \leq 10H$, $d_c = 1.67H$.

parameters. It is notable that the search space for the actual crack parameters is very complex. As an example, Fig. 2 presents a two-dimensional sketch of $E(a_c, d_c, l_c)$ calculated from Eq. (2b), where the true crack parameters are $a_c = 4H$, $d_c = 0.167H$, and $l_c = 2H$.

C. Briefing on Calculation of $f_i(a_c, d_c, l_c)$

As it is hardly possible to calculate $f_i(a_c, d_c, l_c)$ by an analytic expression for such complex a dynamic problem of wave propagation, numerical methods had to be used. The strip element method (SEM), being verified to be especially effective for mechanical analysis of composite plates,[17] has been used to this end. The procedure of using the SEM to calculate $f_i(a_c, d_c, l_c)$ is briefly outlined as follows:

1) Divide the inspected domain of composite plate into some strip elements in the z direction. In each strip element, saying the element k, and expressing the system of governing different equations (no body force) as

$$\rho_k \ddot{\boldsymbol{U}}_k - \boldsymbol{L}^T \boldsymbol{D}_k \boldsymbol{L} \boldsymbol{U}_k = 0 \tag{3}$$

where $\boldsymbol{U}_k^T = \{u(x, z), w(x, z)\}$; $\boldsymbol{D}_k = \{d_{ij}^k\}$, $i, j = 1, 2, 3$; $\boldsymbol{L}$ can be found in Ref. 17.

2) Express $\boldsymbol{U}_k$ in a discrete form using the interpolation functions. Then the principle of virtual work is applied for the discretized governing equation. This results in a one-dimensional-reduced differential equation. By assembling these differential equations for all of the strip elements, a set of second-order approximate governing differential equations for the inspected domain is obtained as follows:

$$\boldsymbol{Q} = -\boldsymbol{A}_2 \frac{\partial^2 \boldsymbol{V}}{\partial \boldsymbol{x}^2} + \boldsymbol{A}_1 \frac{\partial \boldsymbol{V}}{\partial \boldsymbol{x}} + \boldsymbol{A}_0 \boldsymbol{V} - \omega^2 \boldsymbol{M} \boldsymbol{V} \tag{4}$$

Both $\boldsymbol{Q}$ and $\boldsymbol{V}$ are acting on the element node lines along the x direction. Matrices $\boldsymbol{A}_i (i = 0, 1, 2)$ and $\boldsymbol{M}$ can be found in Ref. 17.

3) Solve the preceding equation in an analytical method. This results in a set of linear equations that gives the relationship between the displacements $\boldsymbol{V}$ and stresses $\boldsymbol{R}$ at the nodal points on the vertical boundaries.

$$\boldsymbol{R} = \boldsymbol{K} \boldsymbol{V} + \boldsymbol{S} \tag{5}$$

Matrix $\boldsymbol{K}$ can been also found in Ref. 9.

4) For the plates hiding the horizontal cracks, it is usually divided into four subdomains, I, II, III, and IV, as shown in Fig. 1a. Each subdomain corresponds to a set of equations similar to Eq. (5). By assembling all of the four sets of equations and solving the integrated equation, elastic wave field and the surface displacements are obtained.[3]

5) For the plates hiding the vertical cracks, it is usually divided into two subdomains, I and II, as shown in Fig. 1b. The surface displacements are obtained similarly to the horizontal crack case.[10]

III. Projection Genetic Algorithm (pGA)

A. Description of the pGA

1. Modified mGA

mGA is one of improved GAs.[15,18] It usually operates on a very small population. The small population very often converges in a few generations. When the convergence takes place, a restart strategy is implemented to generate a new population with the same size. This new population consists of the best individual from the previously converged generation and other randomly generated ones. Generally, mGA does not require the use of mutation operation.[15,19] The genetic diversity is introduced and maintained by using the restart strategy.

The modified mGA uses an improved criterion to define the population convergence.[14] This new criterion takes into account not only the number of the genes that are different from each other in two compared individuals but also their positions in the chromosome. It is expressed as

$$\sum_{k=1}^{N-1} \sum_{j=1}^{n} \sum_{i_p=1}^{n_{\mathrm{gj}}} \frac{2 i_p}{(N-1) \sum_{j=1}^{n} n_{\mathrm{gj}} (n_{\mathrm{gj}} + 1)} \leq \gamma \tag{6}$$

i_p counts from right to left for the substrings under comparison. Generally, γ is recommended to be within $5 \sim 10\%$ (Ref. 14).

2. Projection Operator

Projection operator aims to find an even better individual by jumping along the move direction of best individual at each of generations so as to improve the convergence of algorithm. It is a direct local optimizer and requires no use of function evaluations. Generally, this direct optimizer is less efficient than the conventional gradient-based optimizers in the cases where the objective functions are differential.[20] However, it is a preferable choice in the hybrid GAs because of its simplicity in implementation and suitability for many practical problems, where the derivative information is often different, computationally expensive, or even impossible to obtain.

Construction of the move direction of best individual is a key of implementing projection operator. In this study an alternative way is employed, in which p_j^b and p_{j-1}^b are used only if they are not identical. Otherwise, p_j^b and p_j^s should be used instead. This means that the better individual c is obtained by

$$f(c) = \max\{f(c_1), f(c_2)\}, \qquad c \in \{c_1, c_2\} \tag{7a}$$

$$c_1 = p_j^b + \alpha\left(p_j^b - p\right) \tag{7b}$$

$$c_2 = p_{j-1}^b + \beta\left(p_j^b - p\right) \tag{7c}$$

$$p = \begin{cases} p_{j-1}^b, & p_j^b \neq p_{j-1}^b \\ p_j^s, & p_j^b = p_{j-1}^b \end{cases} \tag{7d}$$

where α and β are recommended to be within $0.1 \sim 05$ and $0.3 \sim 0.7$, respectively. It is clear that α and β decide how far the newly generated individual c is from the present best individual p_j^b. Selection of α and β has an obvious effect on the convergence process of pGA. Detailed discussion on this point is addressed in the following examples.

3. Hybridization of Modified mGA with Projection Operator

This can be outlined as follows:

1) Let $j = 0$, randomly initialize $\boldsymbol{P}(j) = (p_{j1}, p_{j2}, \ldots, p_{j\mathrm{N}})$. Every individual $p_{\mathrm{ji}} (i = 1, \ldots, N)$ is a possible solution. It is a set of crack parameters for the present detection problem.

2) Evaluate the fitness or objective function values of $\boldsymbol{P}(j)$.

3) Check the termination condition. If "yes," the process ends. Otherwise, $j = j + 1$ and go to the next step.

4) Carry out the conventional genetic operations: niching, selection, crossover, mutation, and elitism. These operations result in a new generation of solutions. Details on their operations can be found in the general literature on GAs.[11,12,15]

5) Generate offspring $\boldsymbol{C}(j) = (c_{j1}, c_{j2}, \ldots, c_{j\mathrm{N}})$ and evaluate their fitness values. Each of offspring is one of newly generated

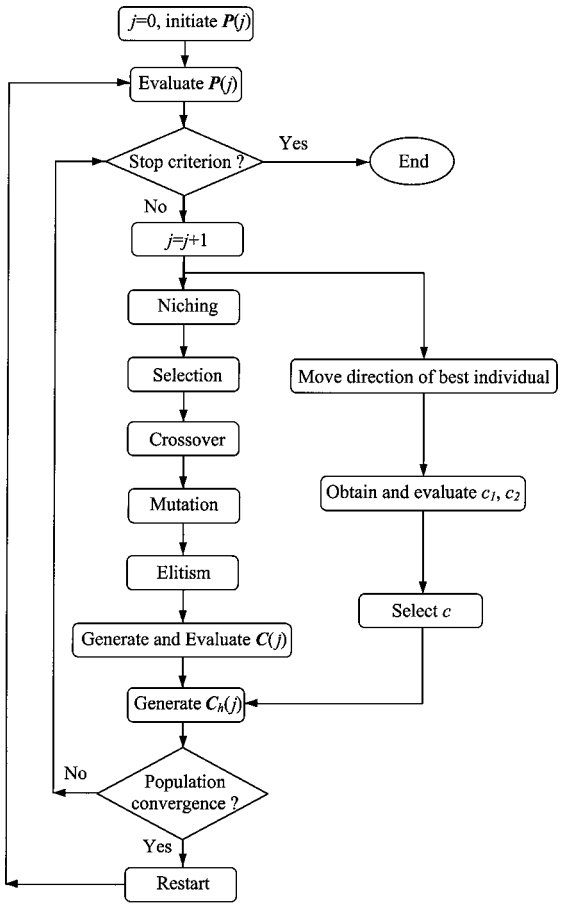


Fig. 3 Flowchart of the pGA.

solutions, that is, a new set of crack parameters for the detection problem. They are expected to be more close to the global optimum than those in the $\boldsymbol{P}(j)$.

6) Carry out the projection operator. a) construct the move direction of best individual using Eq. (7b)∼(7d); b) generate the individuals c_1, c_2, and evaluate their fitness values; and c) select the better individual c.

7) Replace the worst individual in the $\boldsymbol{C}(j)$ with the individual c. This results in a updated offspring $\boldsymbol{C}_h(j) = (c_{j1}, c_{j2}, \ldots, c, \ldots, c_{jN-1})$.

8) Check whether there occurs the population convergence in the $\boldsymbol{C}_h(j)$. If "yes," implement restarting strategy. Otherwise, go back to step 3.

This process is depicted in Fig. 3. Basically, this pGA takes the same strategy in incorporating the local optimizer into the basic loop of mGA as that done in the previous hGA.[14] It thus maintains the main advantages of hGA: 1) much less computation effort is required in the projection operator: 2) the incorporated projection operator affects the evolution process in a self-adaptive manner so as to ensure that the searching is global and the converging is fast; and 3) the hybridization process is straightforward so that it is conventional to use in engineering practice.

B. Performance Test

1. Test Functions

Six benchmark functions are used to test the proposed pGA.[14] They are

$$F1: f(x_1, x_2) = \prod_{i=1}^{2} \sin(5.1\pi x_i + 0.5)^6 \times \exp\frac{-4\log 2(x_i - 0.0667)^2}{0.64}$$

$$\pi = 3.14159, \quad 0 < x_i < 1.0, \quad i = 1, 2$$

$$F2: f(x_1, x_2) = \sum_{i=1}^{5} i\cos[(i+1)x_1 + i] \sum_{i=1}^{5} i\cos[(i+1)x_2 + i]$$

$$-10 < x_i < 10, \quad i = 1, 2$$

$$F3: f(x_1, x_2) = \frac{x_1^2}{4} - \frac{x_1^2}{2} + \frac{x_1}{10} + \frac{x_2^2}{2}$$

$$-10 < x_i < 10, \quad i = 1, 2$$

$$F4: f(x_1, x_2, x_3) = \sum_{i=1}^{3} \left[\left(x_1 - x_i^2\right)^2 + (x_i - 1)^2\right]$$

$$-5 < x_i < 5, \quad i = 1, 2, 3$$

$$F5: f(x_1, x_2, x_3) = \sum_{i=1}^{3} \left[\left(ax_1 - bx_i^2\right)^2 + (cx_i - d)^2\right]$$

$$0.999 \le a, b, c, d \le 1.001 \text{ randomly}, \quad -5 < x_i < 5, \quad i = 1, 2, 3$$

$$F6: f(x_1, x_2, x_3, x_4) = \sum_{i=1}^{5} \frac{1}{\sum_{j=1}^{4}[x_j - d(i, j)]^2 + c(i)}$$

$$\boldsymbol{d}[4, 5] = (4, 4, 4, 4; 1, 1, 1, 1; 8, 8, 8, 8; 6, 6, 6, 6; 3, 7, 3, 7),$$

$$\boldsymbol{c}[5] = (0.1, 0.2, 0.2, 0.4, 0.4), \quad 0 < x_i < 10, \quad i = 1, 2, 3, 4$$

These functions have been specially designed for testing various GAs. They have many local optima and one or more global optima (Table 1). As an example, Fig. 4 shows the search space of function F1.

2. Convergence Performance of the pGA

Convergence performance of the pGA is investigated in terms of the number of generations (or times of function evaluations) required to obtain the global optimum for the above benchmark functions. To make the results meaningful statistically, each of benchmark functions is tested for 40 times using the pGA with the different

Table 1 Convergence performance of the pGA and comparison with the hGA and mGA

No.	Global optimum	Function value	$\bar{n}_{pGA}$	$\bar{n}_{hGA}$	$\bar{n}_{mGA}$	R_{hGA}, %	R_{mGA}, %
F1	(0.0669, 0.0669)	1.0	158	189	984	59.7	11.5
F2	(5.4829, −1.4265)[a]	−186.73	185	348	1136	38.0	11.6
F3	(−1.0467, 0.0)	−0.352	175	437	983	28.6	12.7
F4	(1.0, 1.0, 1.0)	0.0	261	544	1561	34.3	11.9
F5	(1.0, 1.0, 1.0)	0.0	229	583	1648	28.1	9.9
F6	(4.0, 4.0, 4.0, 4.0)	−10.153	652	1195	3271	39.0	14.2

[a]One of global optima.

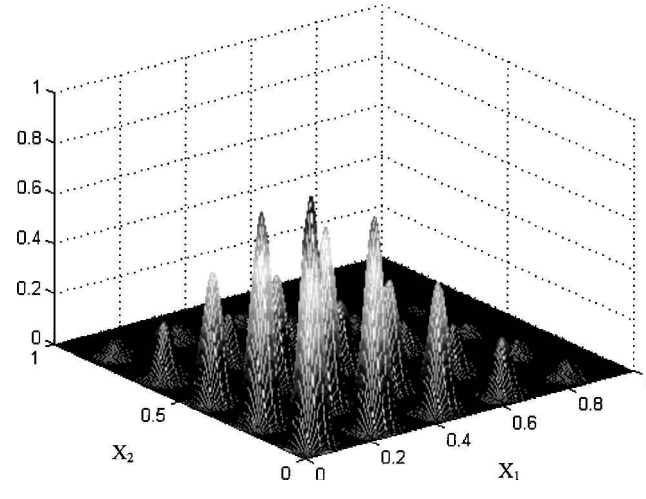


Fig. 4 Function F1.

combinations of α, β, and i_{dum}. These combinations are sampled at $\alpha = 0.1, 0.3$; $\beta = 0.5, 0.618$; and $i_{dum} = -100, -5000, -10{,}000, -15{,}000, -20{,}000, -30{,}000, -35{,}000, -40{,}000, -45{,}000$, and $-50{,}000$ using the full fractional combination method. The genetic operators and other operation parameters used are uniform crossover of $p_{cross} = 0.5$, mutation of $p_{mutate} = 0.02$, tournament selection, one child, niching,[15] elitism,[15] $N = 5$, and $\gamma = 5\%$. Table 1 shows the mean number of generations ($\bar{n}_{pGA}$) to obtain the global optimum for six benchmark functions.

To have appropriate comparison, the hGA and mGA are also run for 40 times with genetic operators and operation parameters similarly to those of pGA (except for $N = 7$) for the six functions just used. The corresponding results ($\bar{n}_{hGA}$, $\bar{n}_{mGA}$) are also shown in Table 1. It can be found that the present pGA demonstrates a much faster convergence than the conventional mGA as well as the preceding hGA. It takes only 9.9 ~ 14.2% (or 28.1 ~ 59.7%) of the number of function evaluations required by the mGA (or the hGA) to obtain the global optimum for the given benchmark functions. Figure 5 shows the convergence processes of functions F3 and F6 using the pGA against the hGA and mGA, from which the evolution processes can be seen clearly.

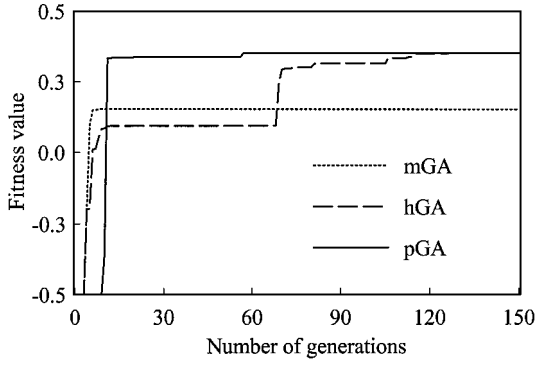
3. *Effect of the Mutation Operation*

Traditionally, the mutation operation is not used in the mGA.[15,18,19] However, it is recommended to apply in the pGA for increasing the population diversity. For testing the effect of the mutation operation, the preceding benchmark functions are investigated using the pGA with and without mutation operation, respectively. It can be found from Table 2 that the pGA with the mutation operation finds the global optimum faster than that without the mutation operation for all of the benchmark functions. Further investigations on the effect of the mutation operation associated with the variation of α, β, and i_{dum} are also performed, and the results are given in Tables 3–5.

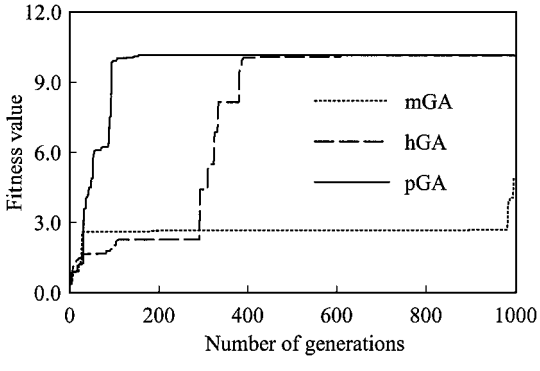
Table 2 Effect of mutation operator on the performance of pGA[a]

Scheme	F1	F2	F3	F4	F5	F6
n_{pGA}, mutation	97	126	104	173	170	412
n_{pGA}, no mutation	107	156	116	352	417	>1000
Ratio, %	90.7	80.8	89.7	49.1	40.8	<41.2

[a]$\alpha = 0.2$, $\beta = 0.5$, and $i_{dum} = -10{,}000$.



Function F3



Function F6

Fig. 5 Convergence processes of the pGA for benchmark function F3 and F6.

Table 3 Effect of coefficient α on the performance of pGA[a]

α	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8
n_{pGA}, mutation	147	86	50	119	38	81	63	56
n_{pGA}, no mutation	149	109	86	130	44	91	74	91
Ratio, %	98.7	78.9	89.3	91.5	86.4	89.0	85.1	61.5

[a]For function F1 where $\beta = 0.5$ and $i_{dum} = -10{,}000$.

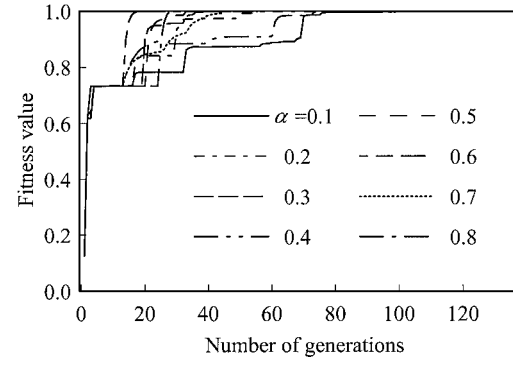
Table 4 Effect of coefficient β on the performance of pGA[a]

β	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
n_{pGA}, mutation	486	217	70	86	80	164	372	243
n_{pGA}, no mutation	121	106	98	109	94	74	109	99
Ratio, %	401.6	204.7	71.4	78.9	85.1	221.6	341.3	245.5

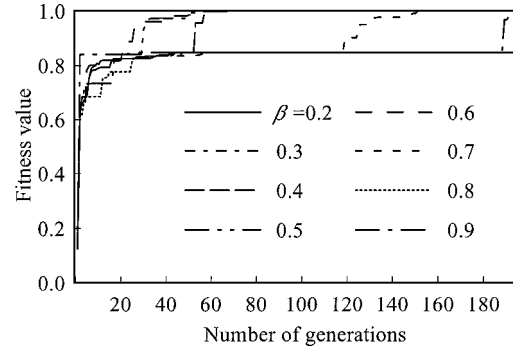
[a]For function F1 where $\alpha = 0.2$ and $i_{dum} = -10{,}000$.

Table 5 Effect of the random number seed i_{dum} on the performance of pGA and mGA

$i_{dum}(\times -10^2)$	1	50	100	150	200	300	350	400	450	500
n_{pGA}, mutation	245	164	97	150	196	229	60	175	67	197
n_{pGA}, no mutation	279	230	107	77	186	238	185	128	395	199
n_{mGA}	823	2741	1229	1512	1105	240	1043	415	526	208



Effect of α. $\beta = 0.5$ and $i_{dum} = -10{,}000$



Effect of β. $\alpha = 0.2$ and $i_{dum} = -10{,}000$

Fig. 6 Effects of coefficients α and β on the convergence processes of pGA for benchmark function F1.

4. *Effect of the Coefficients α and β*

To study the effect of α and β on the pGA, benchmark function F1 is investigated again using the different α and β, but the same genetic operators and other operation parameters ($i_{dum} = -10{,}000$). Two schemes of the pGA with and without mutation operation are used. It has been found from Tables 3 and 4 that n_{pGA} corresponding to the

different α ($\alpha = 0.1 \sim 0.8$, $\beta = 0.5$) ranges from 38 to 147, whereas that corresponding to the different β ($\beta = 0.2 \sim 0.9$, $\alpha = 0.2$) ranges from 70 to 243. Figure 6 shows their convergence processes.

Further investigation has shown that it is difficult to specify exactly the value of α and β that can get the best convergence performance for all of the benchmark functions. However, it has been found that any combination of α and β always results in the pGA having a significantly fast convergence over the conventional mGA using the same genetic operators and operation parameters. It is recommended in this study that α and β are within $0.1 \sim 05$ and $0.3 \sim 0.7$, respectively. Our favorite choice is $\alpha = 0.1 \sim 0.3$ and $\beta = 0.5$, which generally ends in good results in our numerical experiments.

5. Effect of the Random Number Seed i_{dum}

A total of 10 different i_{dum} has been used to investigate the convergence performance of pGA. The 10 i_{dum} are all selected to be negative according to the suggestion made by Carroll[15,19] in a public version (1.7) of the GA program. To show the effect of i_{dum}, Table 5 presents the corresponding n_{pGA} and n_{mGA} for function F1 when pGA and mGA use the different i_{dum}. In these investigations, both the genetic operators and the operation parameters remain the same ($\alpha = 0.2$, $\beta = 0.5$). From Table 5 it can be found that the pGA is not as sensitive as the mGA to i_{dum}. The ratio of the maximal n_{pGA} to the minimal n_{pGA} is about four. This feature makes the pGA more robust to use in practice.

IV. Numerical Examples

A composite plate $[C90/G45/G\text{-}45]_s$ with six symmetrically stacked layers is used in this study to demonstrate the detection of crack using the proposed pGA. C and G stand for carbon/epoxy and glass/epoxy layers, respectively. The next number denotes the fiber orientation with respect to the x axis. Material constants of carbon/epoxy and glass/epoxy are given in Table 6.

Each layer in this plate is divided into four strips in the z direction for the use of the SEM. The horizontal cracks are assumed to locate at the junctions of two adjacent strip elements, whereas the vertical cracks are assumed to be throughout several successive strip elements. All of them are within the region $0 \le x \le 10H$ and $0 \le z \le H$.

We assumed that there are four sets of surface displacement responses experimentally measured at 250 points on the surface of this plate (Fig. 7), corresponding to four crack cases. These responses are excited by an external load $q(t)$ at $x = 0$ having the amplitude $q_0 = 1$ and the normalized frequency $\omega = [3.14\sqrt{(c_{44}/\rho)}]/H$. The pGA is used to detect these "unknown" crack parameters from the "measured" responses.

The true crack parameters are (I) horizontal crack: $a_c = 6.0H$, $d_c = 0.25H$, $l_c = 1.0H$; (II) horizontal crack: $a_c = 4.0H$, $d_c =$

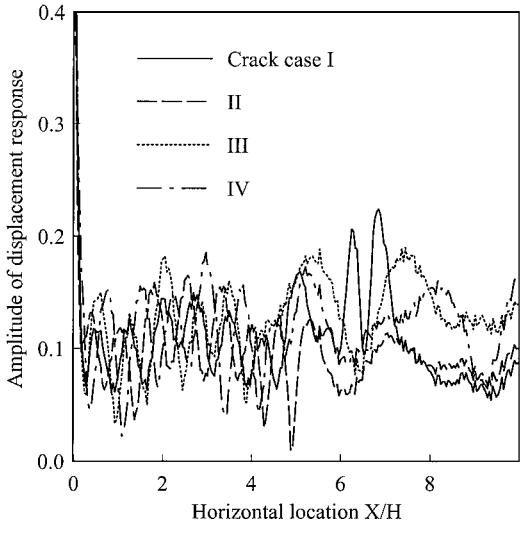
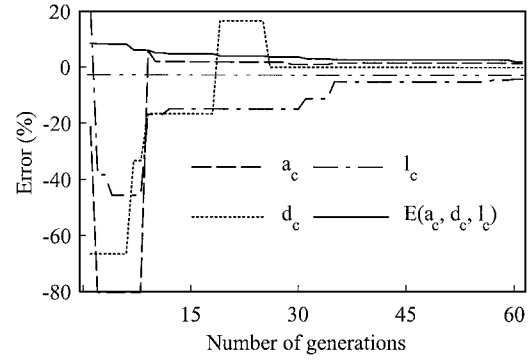


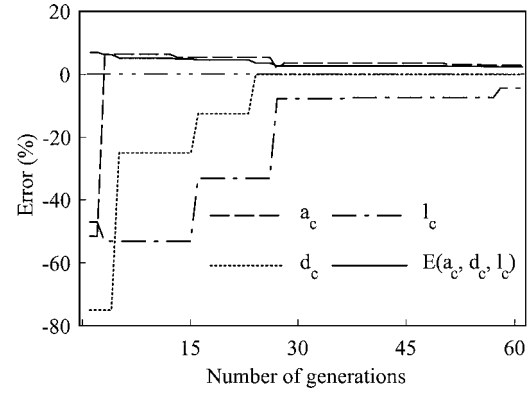
Fig. 7 "Experimentally" measured displacement responses on the surface of plate for four crack cases (noise length is 5%).

Table 6 Material constants of composite plate

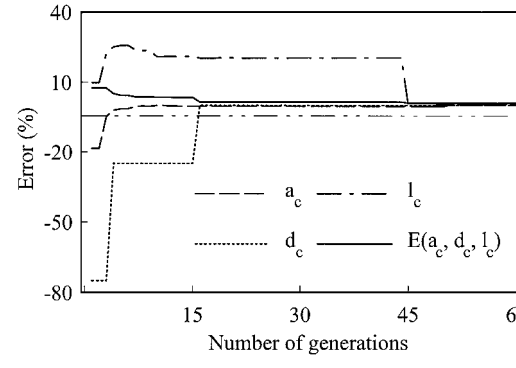
Component	E_1, GPa	E_2, GPa	G_{12}, GPa	v_{12}	v_{23}	ρ, g/cm^3
Carbon/epoxy	142.17	9.255	4.795	0.3340	0.4862	1.90
Glass/epoxy	38.49	9.367	3.414	0.2912	0.5071	2.66



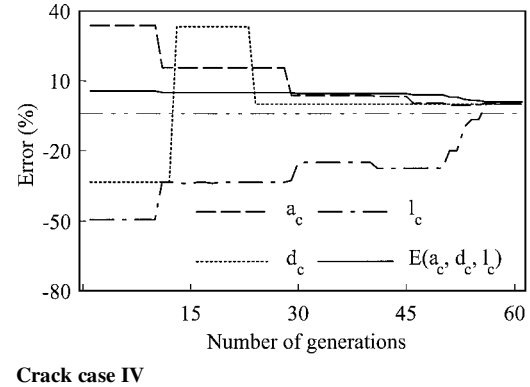
Crack case I



Crack case II



Crack case III



Crack case IV

Fig. 8 Convergence processes of the pGA for detection of four simulated crack cases.

$0.333H$, $l_c = 1.2H$; (III) vertical crack: $a_c = 4.0H$, $d_c = 0.166H$, $l_c = 0.667H$; and (IV) vertical crack: $a_c = 5.0H$, $d_c = 0.125H$, $l_c = 0.75H$. The measured displacement responses are actually simulated using the SEM code for these true crack parameters. A noise $\Gamma(x)$ defined later is added into the SEM calculated data to simulate the error of measurement:

$$\sum_{i=1}^{n} \Gamma(x_i) = 0 \tag{8a}$$

$$\sum_{i=1}^{n} \Gamma(x_i)\Gamma(x_i - \tau) = 2D\delta(\tau) \tag{8b}$$

$$D = \eta \left(\frac{1}{n-1} \sum_{i=1}^{n} x_i^2 \right)^{\frac{1}{2}} \tag{8c}$$

where $\eta = 5\%$. These true crack parameters are regarded as unknown when the damage detection is carried out.

For applying the proposed pGA, a proper error norm is selected to formulate the present damage detection into an optimization problem. Equation (2a) is used in this example for the noise is relatively weak by glancing at the response data. When noise becomes stronger, Eq. (2b) is recommended for use because it has the better stability. This formulated objective function is the fitness function used in the pGA.

The searching range of three crack parameters, a_c, d_c, and l_c, is set to be within $[0, 10H]$, $[0.042H, 0.5H]$, and $[0.2H, 5H]$ for the horizontal crack case and $[0, 10H]$, $[0, 0.5H]$, and $[0.042H, 0.96H]$ for the vertical crack case, respectively. It is decided that the number of possibilities for the three crack parameters are 32,768, 12, and 32,768 for the horizontal crack case, and 32,768, 32,768, and 24 for the vertical crack case, respectively. So the numbers of possible solutions for this detection problem are approximately 1.29×10^{10} and 2.56×10^{10} for the horizontal and vertical crack cases, respectively. Clearly this search space is very large and also with a large number of local optima similar to that shown in Fig. 2.

The pGA uses the same genetic operators in the performance tests and the following operation parameters: $N = 5$, $p_{\text{cross}} = 0.5$, $p_{\text{mutate}} = 0.02$, $i_{\text{dum}} = -10{,}000$, $\gamma = 5\%$, $\alpha = 0.2$, and $\beta = 0.5$. This means a total of five sets of crack parameters are randomly generated at first. Then they are gradually corrected with the proceeding of the convergence of pGA, until the minimal error $E(a_c, d_c, l_c)$ among five sets of crack parameters is sufficiently small. Figure 8 shows the convergence processes of the minimal $E(a_c, d_c, l_c)$ and the corresponding errors of three crack parameters for the four simulated cases. It can be seen that the pGA converges to the satisfactory detection results very fast. The maximal error of the detected crack parameters with respect to their true values is -4.1, -4.3, -0.36, and -0.35% at the 60th generation in the evolution process for the four simulated cases, respectively.

V. Conclusions

In this study a method of damage detection for composite plates using Lamb waves and the pGA is proposed. This method first formulates the damage detection as an optimization problem of minimizing the error between the measured and calculated surface displacement response derived from Lamb waves. Then a projection genetic algorithm is used to solve this optimization problem and identify the actual crack parameters. Numerical examples have demonstrated this method is effective and efficient. This provides the damage detection of composite plates with an alternative method.

The pGA plays a key role in the proposed detection method. It is developed from the hybridization of the modified mGA with a projection operator and has been verified by six benchmark functions. This pGA is also suitable to solve other optimization problems, and the developed projection operator is applicable to hybridize with other GAs in the same strategy discussed in this study.

Acknowledgment

The research is partly supported by the Singapore—MIT Alliance (SMA) program.

References

[1]Zhou, Y., Tong, L., and Steven, G. P., "Vibration-Based Model-Dependent Damage (Delamination) Identification and Health Monitoring for Composite Structures—a Review," *Journal of Sound and Vibration*, Vol. 230, No. 2, 2000, pp. 357–378.

[2]Doebling, S. W., Farrar, C. R., Prime, M. B., and Shevitz, D. W., "Damage Identification and Health Monitoring of Structural and Mechanical Systems from Changes in Their Vibrations Characteristics: A Literature Review," Los Alamos National Lab., TR LA-13070-MS, Los Alamos, NM, May 1996.

[3]Liu, G. R., and Lam, K. Y., "Characterization of a Horizontal Crack in Anisotropic laminated Plates," *International Journal of Solids and Structures*, Vol. 31, No. 21, 1994, pp. 2965–2977.

[4]Karunasena, W. M., Shah, A. H., and Datta, S. K., "Plane-Strain-Wave Scattering by Cracks in Laminated Composite Plates," *Journal of Engineering Mechanics*, Vol. 117, No. 8, 1991, pp. 1738–1754.

[5]Datta, S. K., Ju, T. H., and Shah, A. H., "Scattering of an Impact Wave by a Crack in a Composite Plate," *Journal of Applied Mechanics*, Vol. 59, Sept. 1992, pp. 596–603.

[6]Karim, M. R., Awal, M. A., and Kundu, T., "Elastic Wave Scattering by Cracks and Inclusions in Plates: In-Plane Case," *International Journal of Solids and Structures*, Vol. 29, No. 19, 1992, pp. 2355–2367.

[7]Liu, S. W., Datta, S. W., and Ju, T. H., "Transient Scattering of Rayleigh Lamb Waves by a Surface-Breaking Crack: Comparison of Numerical Simulation and Experiment," *Journal of Nondestructive Evaluation*, Vol. 10, No. 3, 1991, pp. 111–126.

[8]Liu, G. R., and Achenbach, J. D., "Strip Element Method to Analyze Wave Scattering by Cracks in Anisotropic Laminatyed Plates," *Journal of Applied Mechanics*, Vol. 62, Sept. 1995, pp. 607–613.

[9]Liu, G. R., Lam, K. Y., and Shang, H. M., "Scattering of Waves by Flaws in Anisotropic Laminated Plates," *Composites: Part B*, Vol. 27, No. 5, 1996, pp. 431–437.

[10]Lam, K. Y., Liu, G. R., and Wang, Y. Y., "Time-Harmonic Response of a Vertical Crack in Plates," *Theoretical and Applied Fracture Mechanics*, Vol. 27, No. 1, 1997, pp. 21–28.

[11]Gen, M., and Chen, R. W., *Genetic Algorithms and Engineering Design*, Wiley, New York, 1997, pp. 31–34.

[12]Goldberg, D. E., *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison Wesley Longman, Reading, MA, 1989, pp. 202–204.

[13]Back, T., Hammel, U., and Schwefel, H. P., "Evolutionary Computation: Comments on the History and Current State," *IEEE Translations on Evolutionary Computation*, Vol. 1, No. 1, 1997, pp. 3–17.

[14]Xu, Y. G., Liu, G. R., and Wu, Z. P., "A Novel Hybrid Genetic Algorithm Using Local Optimizer Based on Heuristic Pattern Move," *Applied Artificial Intelligence—An International Journal*, Vol. 15, No. 7, 2001, pp. 601–631.

[15]Carroll, D. L., "Genetic Algorithms and Optimizing Chemical Oxygen-Iodine Lasers," *Developments in Theoretical and Applied Mechanics*, Vol. 18, School of Engineering, Univ. of Alabama, Tuscaloosa, AL, 1996, pp. 411–424.

[16]Santamarina, J. C., and Fratta, D., *Introduction to Discrete Signals and Inverse Problems in Civil Engineering*, American Society of Civil Engineers, Reston, VA, 1998, pp. 184–186.

[17]Liu, G. R., and Achenbach, J. D., "A Strip Element Method for Stress Analysis of Anisotropic Linearly Elastic Solids," *Translations of the ASME*, Vol. 61, June 1994, pp. 270–277.

[18]Krishnakumar, K., "Micro-Genetic Algorithms for Stationary and Non-Stationary Function Optimization," *SPIE: Intelligent Control and Adaptive Systems*, Vol. 1196, edited by G. E. Rodriguez, Philadelphia, 1989, pp. 289–296.

[19]Carroll, D. L., "Chemical Laser Modeling with Genetic Algorithms," *AIAA Journal*, Vol. 34, No. 4, 1996, pp. 338–346.

[20]Kalyanmoy, D. E. B., *Optimization for Engineering Design, Algorithm and Examples*, Prentice–Hall of India Private Ltd., New Delhi, India, 1998, pp. 82–103.

E. R. Johnson
Associate Editor