The Use of Genetic Algorithms in Dynamic Finite-Element Model Identification for Aerospace Structures

S.A. Dunn
Airframes & Engines Division
Aeronautical & Maritime Research Laboratory
Defence Science & Technology Organisation
PO Box 4331, Melbourne, Victoria, Australia
E-mail Shane.Dunn@dsto.defence.gov.au

Abstract

In this paper, it will be demonstrated how genetic algorithms can be used with experimentally determined frequency response function data to identify finite-element models of structures for dynamic analyses. The concepts will be demonstrated with a simple 2 degree-of-freedom simulation and then actual identifications will be given for a beam and a large truss structure. A modified genetic algorithm is then described and its performance in the identification of a tailplane is demonstrated.

1. INTRODUCTION

In order to have a finite-element model (FEM) of an aircraft structure that is tractable for dynamic analyses, we must take the aircraft structure, as depicted in Figure 1a, and represent it as a much simpler structure, such as that shown in Figure 1b.

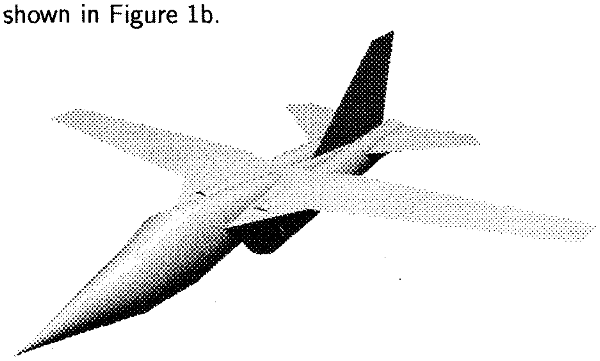

FIGURE 1a.

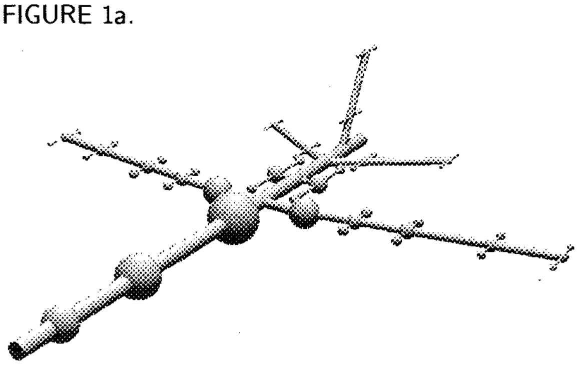

FIGURE 1b. Finite-element representation of an aircraft structure.

Over the past 3 decades, there has been a growing body of research dedicated to finding the best means of creating such dynamic finite-element models, the results of which, adequately describe the true behaviour of the structure. Such research has typically been based on taking experimentally determined modal frequencies and mode shapes and adjusting the model to give better correlation, with the constraint that the modifications are kept to a minimum (eg. Kabe(1)); no such constraints will be applied here. The updating of the mathematical model can be carried out on the mass and stiffness matrices without regard for the physical properties of the structure, or on the physical properties themselves, from which new mass and stiffness matrices will be determined. The coordinate system used may be a reduced, or modal, system, or the physical coordinates may be maintained.

The solution process employed is typically non-linear due to the high degree of dependency of one parameter on one or many of the others. For such a problem, iterative solutions are usually required. These iterative procedures usually fall into two general classes:

i. calculus based approaches, and
ii. naive processes.

A calculus based approach depends upon having a sound mathematical understanding of the system under investigation and being able to estimate the local derivative of the objective function with change in various combinations of parameters. Naive processes, on the other hand, only require that some sort of cost function can be determined for any likely set of parameters, and so are a much more general class of procedures.

Naive procedures can be broken into two categories:

i. enumerative, and
ii. guided random.

Enumerative techniques and purely random searches, are usually prohibitive in terms of processing time. Guided random techniques, however, are generally started with random elements, but they 'learn' as they progress. With this 'learning', they attempt to approach optimal solutions whilst still employing the use of stochastic elements.

In this paper, the results of a technique by which a finite-element model can be identified from experimental data will be shown. The method, which is based on the artificial intelligence tool of genetic algorithms, involves directly determining the optimal mass and stiffness properties of the finite-elements, which comprise the mathematical model, in physical coordinates. The identification problem involves estimating the physical properties such that some measure of the errors between experimental data and those produced by the mathematical model are minimised.

2. GENETIC ALGORITHMS

Some introductory reading focussing on the philosophy of genetic algorithms (GAs) can be found in Holland(2) and Forrest(3). Good introductions to the technical aspects of GAs can be found in Whitley(4) and Beasley *et al*(5).

The idea for genetic algorithms came about from a realisation that, according to Darwinian evolution, nature finds relatively optimal solutions in a naive way to the problem of how to exist on earth. That is, natural evolution does not occur by looking ahead and attempting to determine which features will improve the fitness of a species, but rather tries out different features and those which prove beneficial are preferentially selected. This preferential selection, through an increased likelihood of mating, leads to a higher probability that a fit individual's genes will be spread throughout the species over subsequent generations. The observation that the forces of nature are really the impetus behind a massive highly non-linear optimisation routine led workers to consider whether mimicking natural evolution on computers could be used to solve the relatively much simpler optimisation problems found in engineering. The optimisation problem faced by nature is obviously huge, with the number of possible specific features that an organism can exploit to survive, n, forming an n–dimensional search space where n is tending towards infinity. This search space is also full of local optima, or niches, as is evidenced by the vast array of life on earth from viruses to amoeba to plants, insects, fish, reptiles and mammals. In contrast, the types of optimisation problems that arise in engineering seem trivial. We do not have millions of years to come up with a good solution, but our fitness landscapes are not as complex as those in nature.

Genetic algorithms are based on starting with a randomly generated population of individual possible solutions scattered over a pre-determined search space (the region in which the true answer is thought to lie). The relative fitness of these individuals is determined and a stochastic selection process biased towards the fitter individuals is used to select parents for mating. In mating, attributes of the parents are mixed to form offspring which may, or may not, be fitter than one or both of the parents. In forming offspring, occasional random mutations can occur which also have the possibility of leading to a fitter individual. The process of selection, mating and mutation is repeated over a number of generations to allow the solution to evolve towards an optimum.

These algorithms were first rigorously analysed by Holland(6) who mathematically analysed why genetic algorithms work and developed the concept of *implicit parallelism*. Implicit parallelism describes the manner in which, by starting from an initial random population and then evolving in a manner which results in many diverse individuals, many possible solutions spaces are searched at the same time. Holland argues that it is this feature which makes GAs such a powerful analytic tool for finding global optima.

An example of the power of genetic algorithms to search a large solution space is given in Holland(2). Here, it is described how workers at General Electric and Rensselaer Polytechnic Institute used a genetic algorithm in the design of a high by-pass ratio jet engine. In searching for the optimal design of a jet engine, approximately 100 parameters are varied. Traditionally, this has been done by an engineer changing various parameters, running a simulation and observing the effect the changes had upon the engine's performance. This process would typically take up to eight weeks before a solution would be settled upon. Using an *expert system* which is based on rules learned through previous experience and is capable of estimating the effects of changing one or two variables, an engineer can come up with a design in less than a day which would have twice the improvements derived from the manual method. Such expert systems, however, become stuck and cannot predict the changes that occur when many parameters are changed at the same time. Using expert system solutions in the initial population, a genetic algorithm was run over two days and resulted in half again as much improvement as the expert system results.

In many respects, the problem of the optimal design of a jet engine is analagous to the parameter identification problem for dynamic finite-element models. The traditional way in which dynamic finite-element models are improved is for an engineer to compare experimental results with those from the FEM. Various parameters in the FEM are then updated until a satisfactory solution is found. Also, the changing of one parameter within the FEM may exhibit a high degree of influence on the optimal value of other parameters. It may be thought that genetic algorithms are not applicable to these sorts of problems because they are optimisation tools whereas in the identification of physical properties, we are searching for the one–and–only solution. It must be remembered, however, that in such highly non–linear identification problems, guaranteeing that an answer is the one–and–only global minimum is impossible. Given this, we must be satisfied with the improved result that a non-linear identification technique will give us.

3. Identification of Physical Parameters in Dynamic FEMs

One of the most fundamental questions to be considered when looking at such problems concerns whether modal or frequency response function (frf) data should be used. As has been shown in Dunn[7], modal data give a much more attractive search space for non-linear identification techniques, whereas frf data typically result in a search space which leaves traditional non-linear identification techniques very dependent upon good initial estimates. The apparent attractiveness of modal data, however, is misleading, the reason being that modal data has already been subjected to an identification procedure resulting in reduced data which is likely to be biased by the modal identification procedure. Also, when looking at modal responses, we are observing the structure in a regime where it is subject to the greatest effects from geometric and structural non-linearities. As the aim here is to identify a linear model of the structure and the desire is to work with data that are unbiassed, this paper is concerned with the use of frf data.

3.1 Simulated Two Degree of Freedom System

To illustrate the problems which arise when working with frf data, consider a 2 degree-of-freedom (dof) mass/spring system loaded by a known sinusoidal excitation, as shown in Fig. 2.

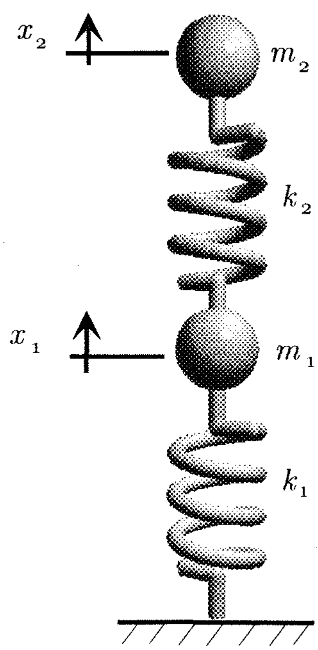

FIGURE 2 Two degree-of-freedom mass/spring system used in simulation.

For this problem, we will consider the two spring stiffnesses as being unknowns. A problem with two unknowns serves as a good illustrative example because the search space for the two unknown parameters can be plotted as a surface over which the solution process must search in order to find the optimal solution. This surface has been plotted for the above system in Fig. 3. Figure 3 is a plot of the error, ε, found by taking the simulated measurements, x, for $k_1 = k_2 = 10$N/m and comparing them with x', which is found by varying k_1 and k_2, over the region sampled. ε is given as

$$\varepsilon = \sum_{i=1}^{n} |x - x'| \qquad (1)$$

where n is the number of frequencies tested. The frequencies used here were $\omega = 0.4 \rightarrow 2$rad/sec in steps of 0.4rad/sec, therefore, $n = 5$. Given that the aim here will be to minimise the error, ε, the surface shown in Fig. 3 can be considered to be the inverse of the fitness function such that the maximum fitness is found when ε is a minimum. It should be noted that as such measurements are directly related to the compliance, rather than the stiffness matrix, the search space is formed in terms of $1/k$.

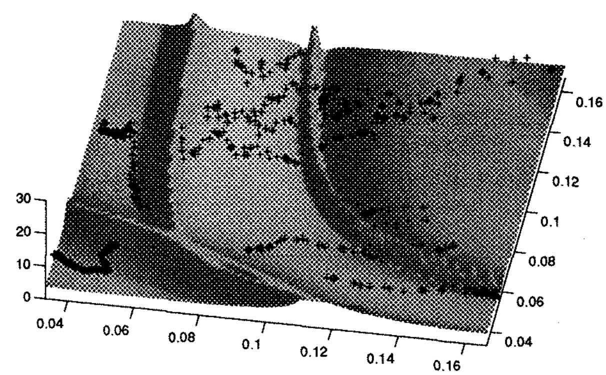

FIGURE 3 Landscape of search region for frf data from the simulated 2 dof system represented in Fig. 2 (the + symbols show the search paths taken by a stochastic hill-climbing algorithm).

As can be seen in Fig. 3, the landscape over which the problem must search is far from simple. For a problem described in this manner, gradient based techniques will always tend 'downhill'. Given the structure of ridges shown in Fig. 3, it is clear that gradient based techniques starting on the wrong side of these ridges will never approach the optimal solution. The crosses marked on Fig. 3 show the paths taken by several runs of a naive gradient technique known as stochastic hill-climbing (Juels and Wattenburg[8]).

Now let us investigate how a genetic algorithm can be used to tackle to problem of identifying the two spring stiffnesses for the problem depicted in Fig. 3 using the same simulated measurements as were used in the stochastic hill-climbing approach; ie. the landscape of the search space is as shown in Fig. 3. In Fig. 4 , every point searched in a random search of 2500 attempts is shown. This can be compared with the search shown in Fig. 5 which is for a genetic algorithm. For the genetic algorithm, the population for each of the 50 generations was 50, thereby leading to 2500 evaluations of ε as described in eqn(1). Each of the runs for the stochastic hill-climbing algorithm shown in Fig. 3 was stopped after 2500 evaluations. For more details on the specifics of the genetic algorithm used here, the reader is referred to Dunn[9].

The random search depicted in Fig. 4, clearly samples the search-space fairly evenly and shows no interesting features, as is to be expected. In Fig. 5 we can see that the region about the actual solution (0.1,0.1) is well searched by the genetic algorithm resulting in a greater likelihood of finding the optimal solution.

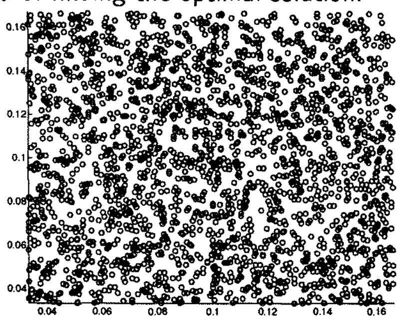

FIGURE 4 Points tried in a random search for the spring stiffnesses of the system depicted in Fig. 2 from simulated data.

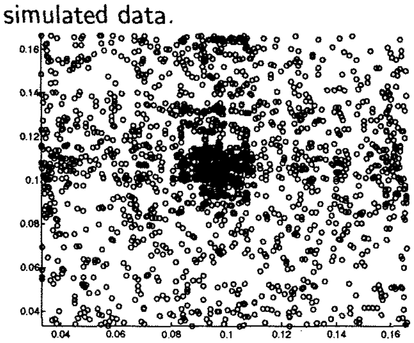

FIGURE 5 Points tried by a genetic algorithm in attempting to identify the spring stiffnesses of the system depicted in Fig. 2 from simulated frf data.

3.2 12 Degree of Freedom Representation of a Beam Using Experimental Data

A uniform rectangular cross–section beam was loaded, as shown in Fig. 6, with broad band noise and the transfer functions between force input and translational accelerations were measured at six evenly–spaced points. This beam was modelled as a 12 dof beam, as shown in Fig. 7. The aim of this task: using a finite–element model in which the geometry of the structure is known and the node locations specified, find the physical characteristics of the beam elements which will best reflect the experimental data. The knowledge that the beam is uniform will be used, meaning that the properties for each of the six beams are taken to be the same.

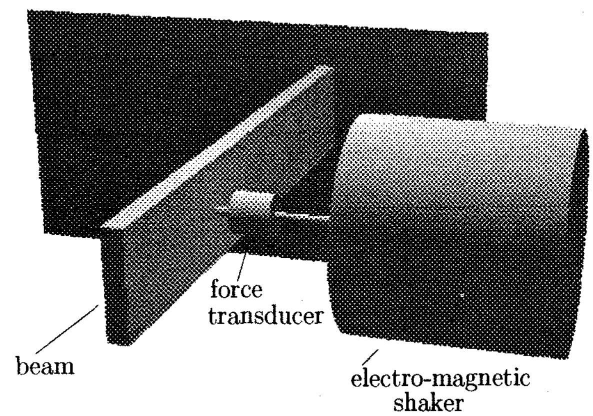

FIGURE 6. Loading arangement for rectanular beam.

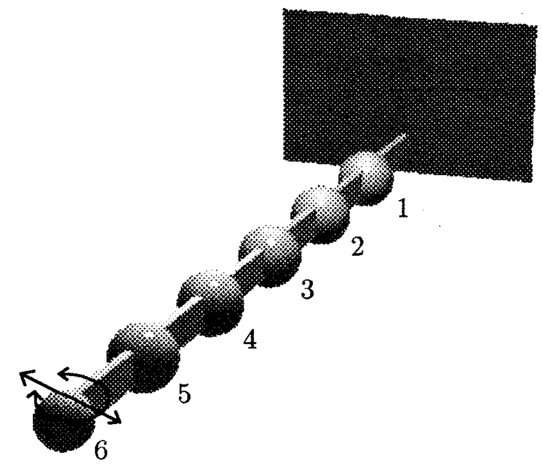

FIGURE 7 Depiction of the finite-element model of the beam shown in Fig. 6. The model has 12 dof with each node having the freedoms shown for node 6.

For a single built–in beam element with lumped mass with the freedoms as shown in Fig. 7, the equation of motion can be written as

$$\left(-\omega^2\begin{bmatrix} m & 0 \\ 0 & I \end{bmatrix} + \begin{bmatrix} 12ei/l^3 & 6ei/l^2 \\ 6ei/l^2 & 4ei/l \end{bmatrix}\right)\begin{Bmatrix} y \\ \psi \end{Bmatrix} = \begin{Bmatrix} F \\ M \end{Bmatrix} \quad (2)$$

where m is the lumped mass at the end of the beam, I is the mass moment of inertia at the end of the beam, e is the Young's modulus, i is the area moment of inertia of the cross–section of the beam and l is the length of the beam element. For the case where the geometry, force inputs and some displacements are known, the unknowns in the above equation can be taken to be m, I and ei. Given that this problem has been specified as identifying the physical properties of six identical beam elements, the unknowns for this problem are the same as for a single beam depicted by eqn(2). If we now consider the full finite-element equation as

$$\left(-\omega^2\mathcal{M}+\mathcal{K}\right)\{x\}=\{\mathcal{F}\} \qquad (3)$$

where $\mathcal{M}$ and $\mathcal{K}$ represent the full finite-element mass and stiffness equations respectively, $\{x\}$ is the vector of displacements and $\{\mathcal{F}\}$ is the vector of force inputs. Rewriting eqn(3) and multiplying the displacements by ω^2 to make the accelerations the subject

$$\{\ddot{x}\}=-\omega^2\left(-\omega^2\mathcal{M}+\mathcal{K}\right)^{-1}\{\mathcal{F}\} \qquad (4)$$

Equation (4) being for a 12 dof finite-element model, is a matrix equation representing 12 individual equations. Given that only the six translational accelerations were measured here, the measurement vector $\{\ddot{x}\}$ is incomplete. If we represent the reduced vector encompassing only the measured freedoms at the jth frequency as $\{\ddot{\mathcal{X}}'\}_j$ and the modelled results for a given m, I and ei for the same freedoms as $\{\ddot{\mathcal{X}}\}_j$ we can write the objective function as

$$\min\left(\varepsilon\left(m,I,ei\right)\right)=\sum_{j=1}^{N}\sum_{i=1}^{n}\left|\ddot{\mathcal{X}}_{i,j}(m,I,ei)-\ddot{\mathcal{X}}'_{i,j}\right| \qquad (5)$$

where N is the number of measured frequencies and n is the number of measured freedoms used in the optimisation process.

The question to consider before implementing a genetic algorithm based on the objective function given in eqn(5) is: at which frequencies should measurements be carried out to obtain the stated aim of identifying a finite-element model which represents the linear structure? As modal frequencies are approached, for a given force input, the amplitude of the response grows dramatically. With this growth in response, structural and geometric non-linearities have a much greater influence on the measured results. Whilst such influences are certainly of importance when considering a structure's modal response, when trying to create a model which best represents the linear characteristics of the structure, measurements near modal frequencies should be avoided. Based on this philosophy, an identification was carried out using a genetic algorithm on the mass/beam model represented in Fig. 7 giving the results presented in Fig. 8. Clearly, the identified model is a very close approximation to the actual structure within the frequency band of the test.

The identified properties are given in Table 1. It should be noted that neither the force transducer nor the accelerometer were calibrated, so the identified parameters have no dimensions. In searching for the solution, the initial search space was set very widely and subsequently reduced for better resolution when the genetic algorithm quickly indicated the region of the actual solution. Table 1 shows the reduced bounds of the search space and the results for 100 separate runs of a genetic algorithm with a population of 50, each run for 50 generations and the resolution between the bounds is 8 bits.

For the results presented in Table 1, it should be noted that of the 100 runs, each consisting of 50 generations, the optimal solution was found 18 times. It should also be noted that there is no guarantee that this solution is the true optimal solution.

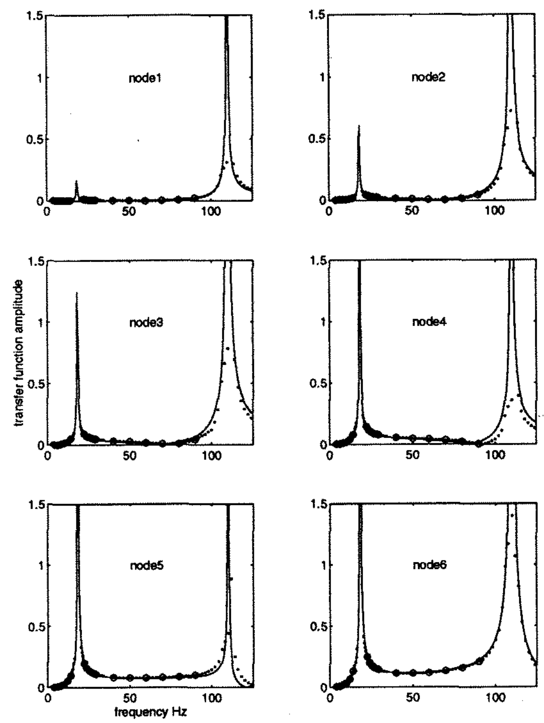

FIGURE 8 Frequency response function for the beam. The dots show all of the measurements taken, the o show the measurements used in the genetic algorithm and the solid line shows the frf of the identified model.

	ei $\times 10^3$	m $\times 10^0$	I $\times 10^{-4}$
1/bounds	$(1-3)$	$(2-6)$	$(2-8)$
opt. solution	1.27	4.17	2.01
mean solution	1.27	4.17	2.03
std. deviation	0.8%	0.6%	2.1%

TABLE 1 Results for finite-element beam properties based on the measurements shown in Fig. 8.

If the finite-element model is an adequate representation of the structure within the frequency range of the test, then as many measurements as were used in Fig. 8 are not required to determine only 3 unknowns. The extra measurements are useful from the perspective of noise rejection, but these measurements are not subject to a great deal of noise. To test the hypothesis that far fewer measurements may be sufficient, measurements at only 4 frequencies below the first mode at the free end of the beam were used; the results are presented in Table 2 and Fig. 9.

The results presented in Fig. 9 show that, at least up to the second mode, the finite-element model is a sufficient representation of the beam. Comparing the results given in Table 2 with those in Table 1, we see that ei and m are essentially unchanged whereas that for I is very different. Given that the results in Fig. 9 clearly show that the identification is a sufficient representation of the structure within the frequency range measured, we can conclude that I has very little influence on the behaviour of the structure in this regime and, consequently, such measurements are not sufficient to determine I.

	ei $\times 10^3$	m $\times 10^0$	I $\times 10^{-4}$
1/bounds	$(1-3)$	$(2-6)$	$(2-8)$
opt. solution	1.24	4.22	6.74
mean solution	1.23	4.17	6.15
std. deviation	2.0%	3.3%	29.4%

TABLE 2 Results for finite-element beam properties based on the measurements shown in Fig. 9.

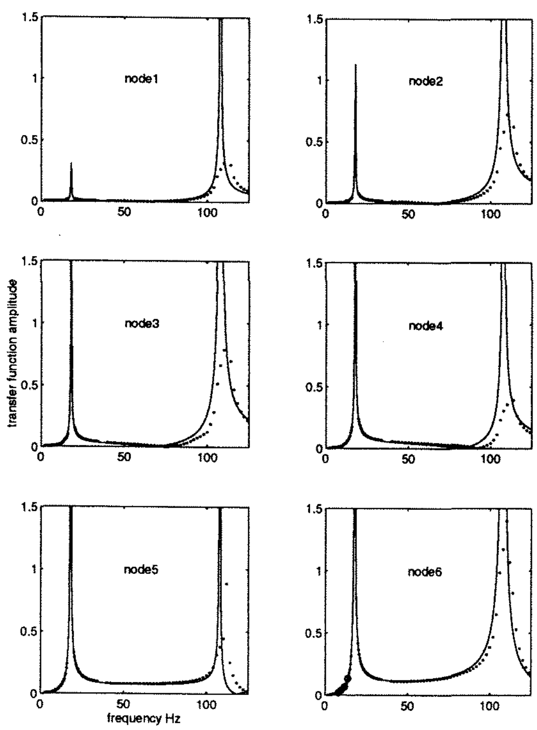

FIGURE 9 Frequency response function for the beam when only the frf measurements at the end of the beam for 4 frequencies below the first mode are used. The dots show all of the measurements taken, the o show the measurements used in the genetic algorithm and the solid line shows the frf of the identified model.

3.3 Identification of NASA 8-bay truss

Research at the NASA Langley Research Center on damage detection for large space structures, led to the compilation of a comprehensive set of measurements for the 8-bay truss depicted in Fig. 11. Full experimental details are given in Kashangaki[10].

The finite-element model identification of the truss was carried out in the same manner as that for the beam in the previous section, though here, the truss elements are assumed to carry only tensile and compressive loads and no bending loads. The identified parameters were: the product of the Young's Modulus and the cross-sectional area for the diagonal elements and the longerons and battens respectively; the mass of the beams for the diagonal elements and the longerons and battens respectively; and the lumped mass representing the node balls and instrumentation at each node of the truss. The model identified by the GA produces the results shown in Fig. 10 using only the transfer function measurements of the accelerations in the direction of loading with respect to the load input at the 8 nodes numbered in Fig. 11.

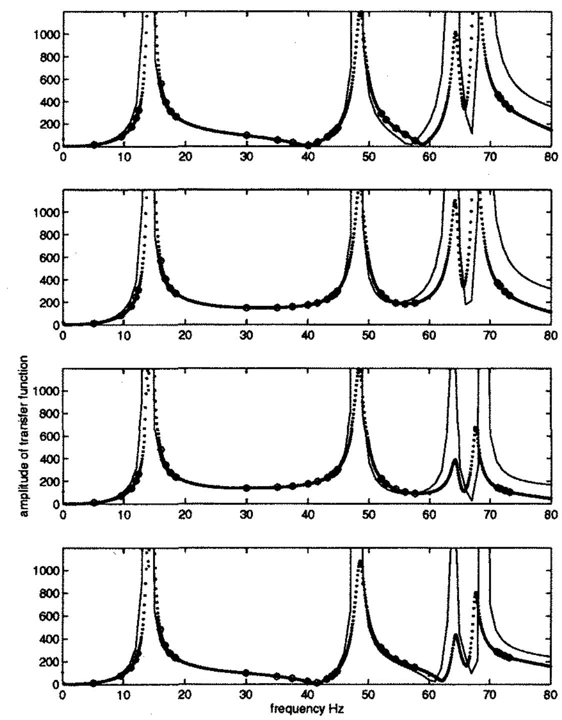

FIGURE 10. Frequency response function for the 8–bay truss. The dots show all of the measurements taken, the o show the measurements used in the genetic algorithm and the solid line shows the frf of the identified model.

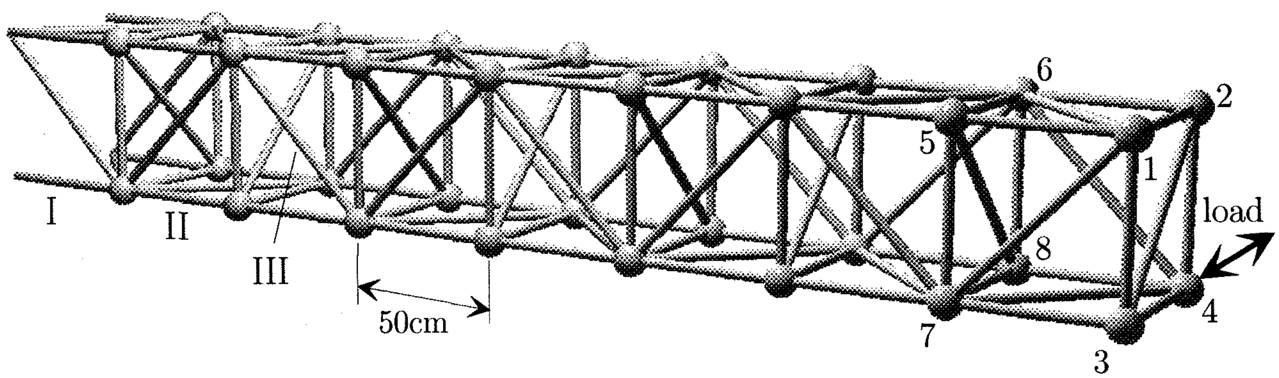

FIGURE 11. NASA 8–bay truss showing point of application of load and nodes from which measurements were used for the updating procedure.

The previous identification was carried out on the structure in an undamaged state. Damage was then simulated in the truss by removing elements and repeating the experiments. The aim of the identification here was to repeat the identification of the truss, as described previously, with the added unknown that 1 element may be missing, and if so, which one. For the damage cases analysed, the element properties are determined well, but solving for the missing element is not necessarily so straight–forward. For the case of element III, as shown in Fig. 11, the GA quickly identifies the missing element; therefore, it can be concluded that for the given loading with the measurements from the 8 nodes, as described above, the information is sufficient to uniquely identify the system. A number of runs of the GA for the case with element I missing, however, shows that there is little difference between the objective functions for element I or II missing; therefore, the given measurements and loading are insufficient to determine the damage uniquely. Nevertheless, the region in which the damage has occurred is narrowed down to one of two elements.

4. Aerospace CT4 Tailplane

The test article used here was the tailplane of a CT4 - a propeller driven military trainer. The structure is of a traditional aluminium construction involving two spars, ribs and a riveted skin. The tailplane was mounted on springs and loaded at its tip as shown in Fig. 12. The loading consisted of band limited white-noise (0–100Hz) and accelerations were measured at the six locations shown in Fig. 12 with a 'roving' accelerometer. The transfer function between the accelerometer readings and the measured load input was determined. It was these transfer functions that were then used in the model identification process.

Three configurations consisting of the tailplane arrangement with different added masses (Table 3) were tested. The mounting, loading and added masses were such that no torsional modes were excited.

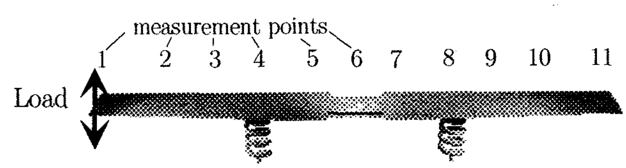

FIGURE 12 Tailplane arrangement showing the point of load application, the spring mountings and the positions where measurements were taken and masses could be added.

Configuration	Added Mass
1	2.0kg at 1 & 11
2	0.2kg at 1 & 11, 0.9kg at 4 & 8
3	1.2kg at 1 & 11, 0.575kg at 2 & 10

Table 3 Added masses for the three configurations.

The objective function for the tailplane identification is as described in eqn(5), though the model to be determined can be depicted as shown in Fig. 13. The unknown parameters are:

— the stiffnesses of beams I & II,
— the magnitudes of the masses i & ii,
— at which node should the change in stiffness occur,
— at which nodes are the masses best placed, and
— the stiffness of the mounting springs.

These unknowns lead to a search in 8-dimensional space.

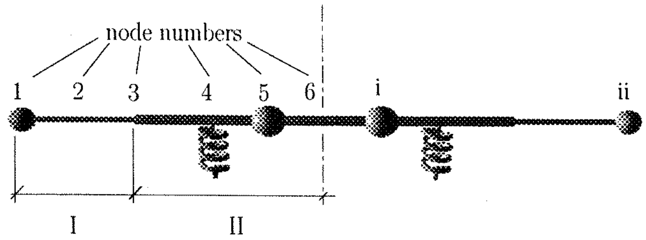

FIGURE 13 Diagram showing lay-out of finite-element model and the node positions where the masses, i and ii, and the beams, I and II, can be placed.

4.1 Modified Genetic Algorithm

The basic form of the GA used here is as has been described in detail in Dunn[9]; a modification which has been found to be of use in this problem is described as follows. As GAs progress through generations from the initial randomly selected population, it is typical that there is a rapid improvement in the cost function, followed by a progressively slower rate of improvement. If this solution is heading for a result remote from the global optimum, then, even though there are mechanisms within a GA which allow the possibility of still finding the global optimum, it can take a very long time. A technique which has been used here involves taking advantage of the features found during the apparently rapid initial convergence by carrying out quick solutions n times, and then for the $(n+1)$st time, the previous n solutions are added to the initial population. The aim of such a technique is that the initial solutions will have some attractive features which, in the subsequent runs where all of them are brought together, will be shared between these solutions via the mechanism of crossover. The resulting populations will then hopefully have individual solutions which will be made up of the attractive aspects of the previous solutions. This method is analogous to a technique described in Goldberg and Richardson[11] which has multiple subpopulations with occasional migration, thereby simulating partial geographical isolation.
This was implemented for the problem described here as follows:

— run the standard GA 10 independent times with a population of 30 for 30 generations; the initial population of 30 is randomly created
— take the 10 solutions from the previous step and a randomly selected population of 30 and run the GA now with a population of 40 for 30 generations
— take the 11 solutions from the 11 previous GAs .... now with a population of 41 for 30 gen's
⋮
— take the 14 solutions from the 14 previous GAs .... now with a population of 44 for 30 gen's.

After all of this, 15,300 cost function evaluations had been carried out. The standard GA as described in Dunn[9] was run for various populations and numbers of generations. Carrying out the identification on configuration 1, as shown in Table 1, the comparison of results from each of these techniques is shown in Table 4 where it can clearly be seen that the additional complexity of the modified technique is worthwhile.

pop.	gen's	func. evals.	Objective function best	mean	std. dev
modified GA		15300	1.67	1.94	0.35
100	153	15300	1.96	5.06	1.96
153	100	15300	2.58	3.36	0.89
300	51	15300	2.34	3.82	0.88
51	300	15300	2.15	3.02	0.71
30	500	15000	2.73	3.47	0.42
500	30	15000	3.34	3.81	0.33

Table 4 Comparison of the modified GA with more standard single run GAs. Each form of GA was run 8 independent times.

4.2 Predictions based on the identified model

The identification was carried out on the tailplane in configuration 1 as shown in Table 3. The frequencies used in the cost function determination and the frf predicted by the identified model are shown in Fig. 14 (Note: only the data for the measurements at node 1 are shown, though measurements at nodes 1 to 6 were used).

The aim of the identification was to determine a model which will predict the structure's dynamic behaviour under different conditions. Adjusting the identified model to account for the known changes in mass distribution, the predicted frf and the measured response for the tailplane in configurations 2 & 3 are shown in Fig. 14.

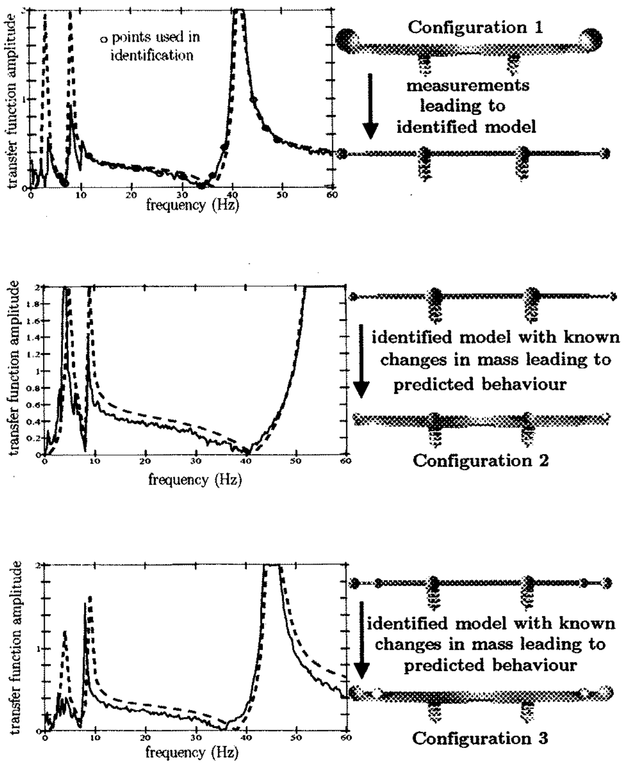

FIGURE 14 Identified and predicted dynamic behaviour of the tailplane loaded as shown in Fig. 12. Note: the dashed line represents the identified and predicted response of the finite-element model and the solid line shows the measurements.

5. Conclusions

It has been shown how genetic algorithms can be successfully employed in the identification of simple finite-element models for dynamic analyses of aerospace structures. A modification to the more standard genetic algorithm has also been described and its efficacy in identifying the finite-element model components for a light aircraft tailplane has been demonstrated. Further work in this topic includes extending the procedure to a full aircraft structure, investigating how to optimise the measurement points used in the identification and investigating the applicability of related techniques, such as simulated annealing, to these problems.

Acknowledgements

The 8–bay truss data used in section 3.3 were supplied by Structural Dynamics Branch, NASA Langley Research Centre, under the auspices of TTCP HTP-8 Collaborative Program 8C-2 'Application of System Identification Methods for Improving Finite-Element Modelling of Aircraft Structures'.

References

(1) A.M. Kabe, 'Stiffness matrix adjustment using mode data', *AIAA Journal*, 1431–1436, **23**, 9, 1985
(2) J.H. Holland, 'Genetic Algorithms', *Scientific American*, 44–50, July, 1992.
(3) S. Forrest, 'Genetic Algorithms: Principles of Natural Selection Applied to Computation', *Science*, 872–878, **261**, 13th August, 1993.
(4) D. Whitley, 'A Genetic Algorlithm Tutorial', *Statistics and Computing*, 65–85, **4**, 1994. Also available via anonymous ftp from:
ftp.krl.caltech.edu:/pub/EC/GA/papers/tutor93.ps.gz
(5) D. Beasley, D.R. Bull and R.R. Martin, 'An Overview of Genetic Algorithms: Part 1, Fundamentals', *University Computing*, 58–69, **15**, 2, 1993. Also available via anonymous ftp from
ftp.krl.caltech.edu:/pub/EC/GA/papers/over93.ps.gz
(6) J.H. Holland, *Adaptation in Natural and Artificial Systems*, MIT Press, 1975.
(7) S.A. Dunn, 'Issues concerning the updating of finite-element models from experimental data', NASA TM 109116, June, 1994.
(8) Juels, A and Wattenberg, M., 'Stochastic Hillclimbing as a Baseline Method for Evaluating Genetic Algorithms', Technical Report 94-834, UC Berkeley. Also available via anonymous ftp from
ftp.krl.caltech.edu:/pub/EC/GA/papers/csd94-834.ps.gz
(9) S.A. Dunn, 'The use of genetic algorithms and stochastic hill-climbing in dynamic finite element model identification', submitted *Computers & Structures*, 1995.
(10) T.A.L. Kashangaki, 'Ground vibration tests of a high fidelity truss for verification of on orbit damage location techniques', NASA TM 107626, 1992.
(11) D.E. Goldberg and J. Richardson, "Genetic Algorithms with Sharing for Multimodal Function Optimization," *Proceedings of the Second International Conference on Genetic Algorithms*, J.J. Grefenstette Ed., Lawrence Erlbaum Associates, 1987, pp41–49.