RESEARCH PAPER

Distributed multilevel optimization for complex structures

J. W. Wind · D. Akçay Perdahcıoğlu · A. de Boer

Abstract Optimization problems concerning complex structures with many design variables may entail an unacceptable computational cost. This problem can be reduced considerably with a multilevel approach: A structure consisting of several components is optimized as a whole (global) as well as on the component level. In this paper, an optimization method is discussed with applications in the assessment of the impact of new design considerations in the development of a structure. A strategy based on fully stressed design is applied for optimization problems in linear statics. A global model is used to calculate the interactions (e.g., loads) for each of the components. These components are then optimized using the prescribed interactions, followed by a new global calculation to update the interactions. Mixed discrete and continuous design variables as well as different design configurations are possible. An application of this strategy is presented in the form of the full optimization of a vertical tail plane center box of a generic large passenger aircraft. In linear dynamics, the parametrization of the component interactions is problematic due to the frequency dependence. Hence, a modified method is presented in which the speed of component mode synthesis is used to avoid this parametrization. This method is applied to a simple test case that originates from noise control.

Keywords Multilevel optimization · Neural networks · Genetic algorithms · Component mode synthesis

1 Introduction

During the development of many technical products, an automated scheme to evaluate the possible advantages and disadvantages of new design considerations is highly desired. This wish has lead to the development of a flexible optimization scheme.

A fully stressed design (FSD)-based strategy allows the optimization of up to hundreds of design variables and very complex separable structures (Haftka and Gürdal 1992; Berke 1992; Sobieszczanski and Loendorf 1972). This is achieved by separating the structure optimization problem into a large number of sub-problems known as component optimization problems. The scheme consists of a *structure evaluation*, where the interactions between the components (e.g., component loads) are calculated and *component optimizations* where the components are optimized separately while taking into account the interactions. The iterative process of structure evaluation and component optimization is continued until convergence.

Please note that the term FSD refers to the historical background of the method. In the application, both buckling and strength constraints are present. Hence, the component is not necessarily fully stressed, but optimized up to a point where constraints prevent further improvement. The FSD scheme is inherently suited for

J. W. Wind (✉) · D. Akçay Perdahcıoğlu · A. de Boer
Faculty of Engineering Technology,
Applied Mechanics Section, University of Twente,
P.O. Box 217, 7500 AE Enschede, The Netherlands
e-mail: j.w.wind@ctw.utwente.nl

D. Akçay Perdahcıoğlu
e-mail: d.akcay@ctw.utwente.nl

A. de Boer
e-mail: a.deboer@ctw.utwente.nl

parallel computing, and it is efficient in cases where a change in one component has a small influence on the optimum of the other components. An important disadvantage of this scheme is the fact that it will not result in an optimal solution unless certain strict conditions are met. For information on convergence and optimality in the current test cases, the reader is referred to Sections 3.4 and 4.4.

In the component optimization strategy, the objective and constraint functions are approximated based on finite element (FE) results using neural networks (NNs). The optimum in this approximate optimization problem is identified with a genetic algorithm (GA).

This paper is built up as follows. In Section 2, the component optimization strategy is described, and in Section 3, an application concerning an aircraft vertical tail plane (VTP) is discussed. In Section 4, component mode synthesis (CMS) is introduced into the optimization strategy to make it suitable for problems in linear dynamics. A benchmark problem derived from the placement of sensors and actuators in active structural acoustic control (ASAC) is used as an example. Finally, conclusions are given in Section 5.

2 Component optimization strategy

2.1 Neural networks

The artificial neural network (ANN) structure is inspired by the working principle of the brain. The neurons considered in ANNs are simple abstractions of biological neurons, and they are used to predict the relations between the particular input–target data set. As can be deduced from Hornik (1989), a two layer NN having sigmoid transfer functions with sufficient number of neurons in the hidden layer and a linear transfer function in the output layer can be trained to approximate any function (see Fig. 1). This ability to approximate functions to any desired degree of accuracy makes NN an attractive tool for use in a response

Fig. 1 A two layer NN structure

surface analysis. A mathematical description of a two layer NNs can be given as,

$$\widehat{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{b} \tag{1}$$

$$\widetilde{\mathbf{x}} = f(\widehat{\mathbf{x}}) \tag{2}$$

$$\mathbf{y} = \mathbf{B}\widetilde{\mathbf{x}} + \mathbf{c} \tag{3}$$

where $\mathbf{x} \in \mathbb{R}^{N_i \times 1}$, $\mathbf{y} \in \mathbb{R}^{N_h^2 \times 1}$ represent the input–target vectors, N_i, N_h^1, and N_h^2 denote the number of input vector elements, hidden layer neurons, and output vector elements, respectively. The function f used in the hidden layer stands for the set of nonlinear (sigmoid) transfer functions and allows the network to learn nonlinear and linear relationships between the input–target pairs. The linear output layer lets the network produce the values outside the sigmoid functions range. $\mathbf{A} \in \mathbb{R}^{N_h^1 \times N_i}$, $\mathbf{B} \in \mathbb{R}^{N_h^2 \times N_h^1}$, $\mathbf{b} \in \mathbb{R}^{N_h^1 \times 1}$, and $\mathbf{c} \in \mathbb{R}^{N_h^2 \times 1}$ denotes the network parameters. The weights $\mathbf{A}$, $\mathbf{B}$ have an effect on the slope of the network response, and the bias terms $\mathbf{b}$, $\mathbf{c}$ shift the entire network response on the coordinate axis (Demuth and Hagan 1996).

This NN architecture is known as a feed forward network, and the parameters of the network can be found using a backpropagation algorithm as follows. Based on a set of known input–target pairs, the obtained network output is compared to the target value. Then, the network parameters (weight and bias terms) are adjusted to minimize the error between output and target values in the sense of least squares. The errors propagate backwards from the output nodes to the inner nodes. In this study, the NN with one hidden layer is considered. Levenberg Marquart and conjugate gradient methods are the most popular methods which are used for the estimation of network parameters (Menhaj and Hagan 1994). The Levenberg Marquardt method is utilized in the applications.

During the network training, it is possible to drive the error on the training set to a very small value, but this does not necessarily mean that input–target pairs that are not part of the training set are also predicted well. This effect is known as overfitting. In the results given in Sections 3 and 4.3, the number of hidden layers are limited to prevent overfitting. On the other hand, generalization methods such as early stopping (Chen and Hagan 1999) and Bayesian regularization (Mackay 1992a,b) are more advanced tools to avoid this problem. A comparison of these strategies in the current setting requires further research.

Two layer networks are capable of approximating any reasonable nonlinear function if sufficient neurons are provided in the hidden layer. In some situations, the backpropagation algorithm does not present the
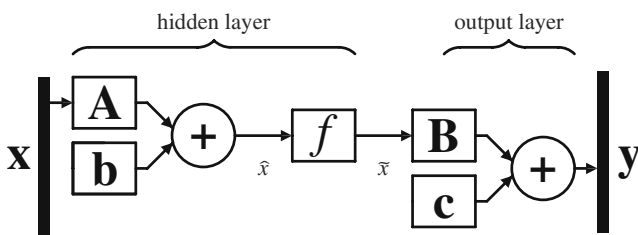
correct weights and biases for the optimum solution. The main reason is that the nonlinear transfer functions in the hidden layer introduce many local minima to the error function and as the numerical techniques used to minimize the error function are gradient based methods, depending on the initial conditions, network solution can be trapped in one of the local minima. This can be prevented by reinitializing the network and retrain it several times until satisfactory convergence is obtained.

2.2 Genetic algorithms

The GA is a method for solving parameter optimization problems in the global sense by imitating the principles of natural evolution. The GA generates a population of points in each iteration and the best point of the population approaches an optimal solution which increases the possibility of finding the global optimum. During its process, the GA does not require any derivative information of the objective function.

There are many ways to handle constraints in GAs (Coello 2002). In this study, an augmented death penalty strategy is used for the constraints, which can be briefly summarized as follows. If the produced point does not violate any constraint, then it is used in the current generation. If it violates the constraints, the point is not taken into account, and the operation is repeated with redefined crossover and mutation functions until a feasible point is found. If no feasible results can be found, then the obtained results for a point is sorted based on predefined criteria, and the most appropriate point is selected.

The logic behind the GA enables handling general classes of optimization problems that are not well suited for gradient based optimization algorithms. The GA can process optimization problems that have discontinuous, nondifferentiable, stochastic, or highly nonlinear objective functions. All these nice features come with a cost; the GA requires more function evaluations than the gradient-based algorithms. Furthermore, GAs only estimate the exact optimum; however, gradient-based methods find it exactly. For specific optimization problems, with a good initial guess close to the global optimum, a gradient-based method will likely be much faster and more accurate than GAs.

2.3 Combined strategy

Design optimization starts with the *modeling* of the problem and then *optimizing* it. Modeling consists of selection of design variables, formulation of the objective function, and decision of the constraints.

Optimization consists of selection of any suitably chosen optimization algorithm and optimizing the objective function under defined constraints using the algorithm.

Engineering optimization usually involves the evaluation of numerical (FE) models. Many popular algorithms require the derivatives of the objective and the constraint functions as well as the function value itself. The derivatives are calculated using either finite difference, analytical, semi-analytical or automated derivative methods (Kirsch et al. 2005; Uri 2004). As discussed in the previous section, some of the disadvantages of gradient-based methods are avoided using GAs. However, the direct application of a GA is unacceptable due to its computational cost.

As an alternative to direct optimization methods, *approximate optimization* methods are used (see for example Jin et al. 2001; Ong et al. 2004). In these methods, the first step is to make the decision of the design of experiments (DOE) set, which holds the information of different design configurations. The next step is to run FE calculations for each of the DOE set elements and to obtain a training set containing different design configurations and corresponding FE model responses. The following step is to fit a response surface through the obtained training set, which is done using NNs in this study. Finally, an optimization is performed on the NN model using GAs. As the NN model approximates the FE model in a fast and accurate way, the required time drawback of GAs is compensated using NNs.

Depending on the accuracy of the NN model, the approximate optimum may not be satisfactory compared with the FE result. In this case, it is possible to supply the training set with the FE result, train the NN again, compare the network response with the FE response, and iterate this process until a desired result is obtained. This approach constitutes the basis of the component optimization strategy in this study (see Fig. 2).

The presented optimization strategy is especially suited for structures that can be separated into components, where the interactions between them (e.g., boundary forces) can be parameterized. The variables that define a component can be separated into three groups (see Fig. 3). *Design variables* are variables that are optimized. *Fixed parameters* (e.g., the required length or width) are the parameters of the component that are not optimized. *Interactions* (e.g., the applied loads) represent the parameters of the components that depend on the design variables of other components. On the global level, only the interactions are iteratively updated.
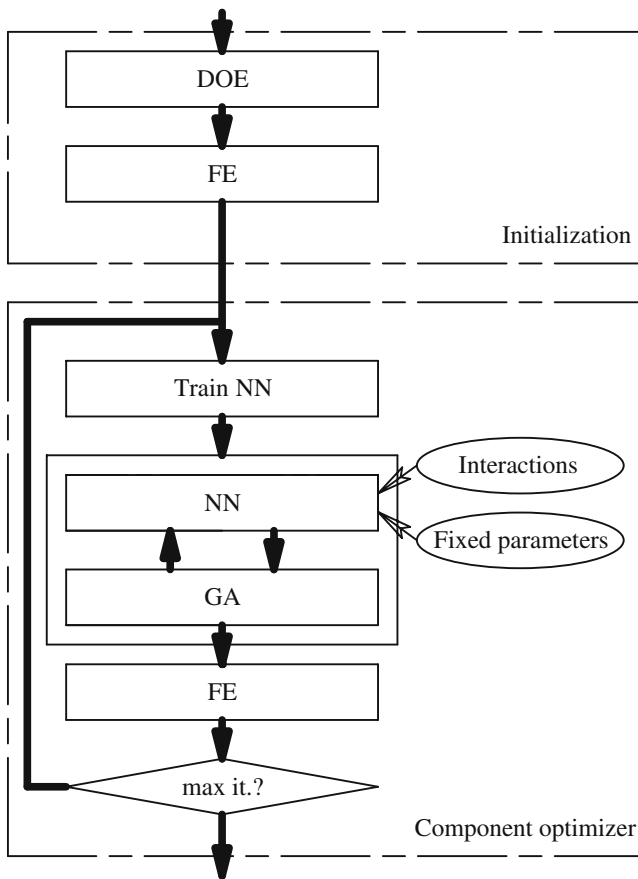
Fig. 2 Component optimization strategy

If the physical meaning of the design variables of two components are the same, they are said to have the same *design configuration*. In this case, they share the same NN model to enhance the efficiency of the optimization. If the design configurations are different, then separate NN models are used. During the component optimization process, the optimizer may choose between different design configurations.

There also exist cases where the differences between component optimization problems cannot be parameterized in a brief way. This is true in certain optimization problems in dynamics for instance. A NN-based optimization strategy that is efficient for this type of problem is discussed in Section 4.1.

In literature, optimization strategies that rely on response surfaces or meta-models have received considerable attention over the past few years
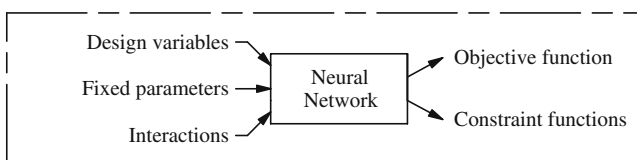

Fig. 3 Input and output of the approximate model

(Simpson et al. 2001). In our implementation, the use of feed-forward NNs are not considered superior to the more conventional approximation techniques for small-scale problems (Wind 2005). Results on large-scale problems on the other hand are considered competitive (see Section 3). Hence, it is concluded that the optimization strategy derives its power from the accumulation of a large number of FE results in the NNs. The use of other response-surface techniques as well as better NN training strategies requires further study.

Advanced multilevel schemes have also received considerable attention in literature (Yates et al. 1994). The current scheme has been chosen for its simplicity and its comparatively wide range of applications.

3 An application: the vertical tail plane

3.1 Problem

The center box of an aircraft VTP provides stiffness and strength to the VTP structure. Hence, the design problem consists of a weight minimization of the structure subject to strength and buckling constraints. The structure is modeled as a conjunction of a large number of spar, skin, and rib panels that serve as components in the optimization strategy (see Fig. 4).

The components are modeled as rectangular panels with design variables such as the panel thickness, stiffener, and hole locations. These design variables are chosen in such a way that the panels are symmetric
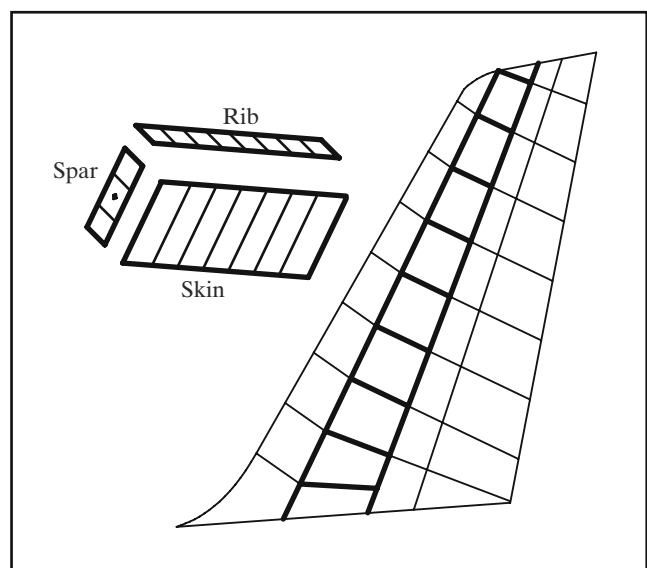

Fig. 4 Schematic representation of an aircraft VTP with skin, spar, and rib panels

(see Fig. 5a). In the optimization of the spars, approximately 20 design configurations, which are distinct in the number of transverse stiffeners, longitudinal stiffeners, and holes, are taken into account (see Fig. 5b). The fixed parameters of this optimization are the panel length and width. The interactions, which are updated iteratively, consist of in-plane boundary forces that represent bending and shear (see Fig. 5c and d).

The constraints are applied on the component level. The buckling constraint is derived from a FE model of the simply supported, stiffened panel. Although these constraints include panel and stiffener buckling, buckling modes that span several components are not taken into account. The strength constraint is based on a maximal principal strain in the panel.

The VTP structure has a height of approximately 16 m and consists of 36 spar, 18 rib, and 36 skin panels. Access holes are required in all spar panels except the six lower ones. The geometry and loads are based on the Airbus A380 VTP, but differences exist in the material models and allowable values. More detailed

information on the problem and implementation are given in Ruijter et al. (2003), Ruijter and Spallino (2003), and Entzinger (2004).

3.2 Implementation

The general component optimization strategy has been presented in Sections 1 and 2.3, but a number of problem-dependent matters are discussed in this section.

On the global level, all components are assembled in a large FE model with a very coarse mesh. Although this model does not predict the local buckling load and stress accurately, the boundary forces on all components are predicted well. These forces are derived and used in the component optimization.

The objective function in the component optimization is panel weight. This function is not approximated by the NN but calculated analytically for each solution generated in the GA. Furthermore, when a feasible
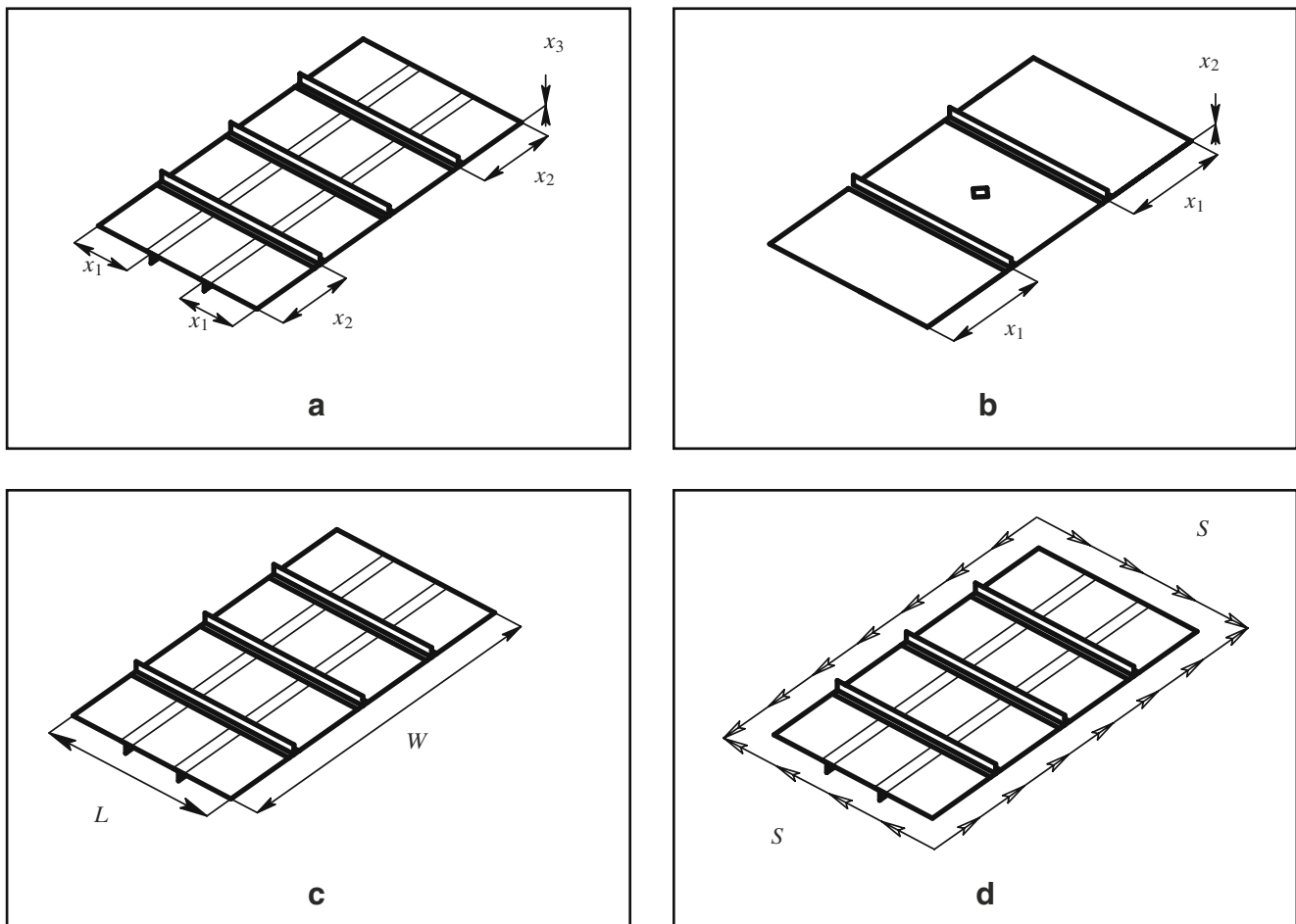

Fig. 5 Design variables, design configurations, fixed parameters, and interactions for the VTP problem. **a** Design variables, **b** alternative design configurations, **c** fixed parameters, **d** interactions

design has been found, a new design cannot be an improvement on the component level unless it is lighter. Hence, the requirement that each new design must be lighter than the best feasible solution found for that component is implemented as an explicit constraint in the GA. This selection rule combined with a rule that enforces the evaluation of different design configurations prevents stagnation of the optimization process. Finally, separate NNs are used for skin, spar, and rib panels because they have different load-cases and design configurations. This means that each NN can be identified uniquely by a design configuration number and a panel type.

3.3 Results

The proposed strategy has been implemented and applied to the VTP problem. Twenty MS Windows machines with a clock speed of 2.6 GHz have been used. Convergence has been achieved after 3 structure iterations using a fixed number of 35 component iterations. With an average FE solving time of approximately 1 min per panel, the optimizations completed in 8–9 h. This runtime is merely an indication because runtimes depend strongly on the mesh sizes as well as the number and speed of the machines available. Nevertheless, it is concluded that overnight optimization of a complete structure is possible provided that enough computers are available.

The resulting design configurations of the spar optimization are given in Fig. 6. In the results of the rear spar, it can be seen clearly that large and heavily loaded panels have more stiffeners than small

Fig. 6 Resulting design configurations of a global-local optimization. **a** Front spar, **b** rear spar

and mildly loaded panels. Although the effects are less dramatic, similar trends can be observed for the front spar.

3.4 Discussion

The optimization strategy has provided good results in a short amount of time. However, it is stressed that there are no guarantees that the global–local strategy will provide optimal or even useful results for general structures and other load cases. On the other hand, the technique leads to an exact optimum for statically determinate trusses subject to only stress and minimum gage constraints if the areas of a cross-section are the design variables. For such trusses, the FSD optimality criterion can be stated as follows:

For the optimum design, each member of the structure that is not at its minimum gage is fully stressed under at least one of the design load conditions (Haftka and Gürdal 1992).

This statement has been adapted to serve as an intuitive optimality criterion for the VTP problem, which can be stated as follows:

For the optimum design, each component cannot be made lighter without violating any constraints.

This criterion is not exact. During the optimization of this statically indeterminate structure, components are optimized under local objective functions and local constraints. The deterioration of the other components is not taken into account. Hence, a combination of designs that are each optimal from a component point of view are not necessarily an optimum from a global perspective. The results of the FSD-based strategy are therefore treated with due suspicion and care in this case.
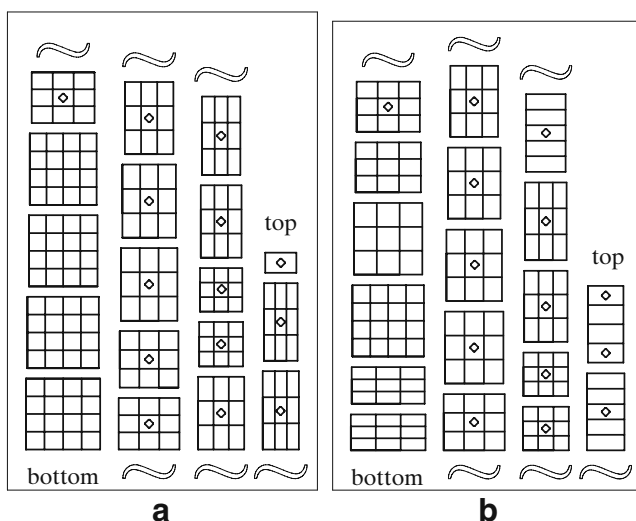
4 Optimization in structural dynamics

4.1 Optimization strategy

In the dynamic case, the boundary forces of a component are frequency dependent, and they may also depend strongly on the design variables of the component itself. Therefore, instead of the boundary forces, the rest of the structure is kept constant during the component optimization. Furthermore, the definition of a 'fully stressed' or 'optimal' component is difficult in some fields of dynamics such as noise vibration and harshness because the objective functions tend to depend on the behavior of the entire structure over a wide frequency range. Hence, each component is optimized to minimize a global objective function. Some

comments on the optimality of the results of this strategy are given in Section 4.4 and references therein.

To make the above decisions possible, the dynamic behavior of the entire structure must be available on the component level. A simple but inefficient way to do this is to use global FE models during the component optimization. In this strategy, the design variables of one component change from one FE analysis to the next, but the design variables of the rest of the structure remains the same during component optimization. As the interactions are not parameterized, the NNs can only be trained with the FE results of a single component and a single structure iteration.

A more efficient strategy can be obtained by using CMS (see, e.g., Craig 1981). CMS consists of two steps:

Model reduction Based on a FE model of a component or a remaining structure, a system of equations that approximates the dynamical behavior predicted by the FE model with a very small number of degrees of freedom is generated.

Synthesis Several reduced models are combined, and the dynamical behavior of the structure as a whole is calculated. It is convenient to calculate the component objective and constraint functions as well in this step.

In this strategy, the initialization of the component optimization consists of generating a set of reduced component models using the DOE set. Based on the initial parameters of the other components, a reduced model of the rest of the structure (the *remaining structure*) is generated (see Fig. 7). A component iteration now consists of four steps. First, the synthesis process is applied to combine each of the reduced component models with the reduced model of the remaining structure. Second, the NN is trained using input–target pairs consisting of the design variables of the considered component and the results of the synthesis process, respectively. Third, the GA is used to identify the optimal result of the NN model. Fourth, a model is evaluated based on these optimal design variables. This model evaluation consists of creating the reduced component model and synthesizing it with the existing reduced model of the remaining structure. If the maximum number of component iterations has not been reached yet, then the result of the model evaluation is added to the NN data set, and a new component iteration begins.

As the design variables of the other components change after the first structure iterations, the set of
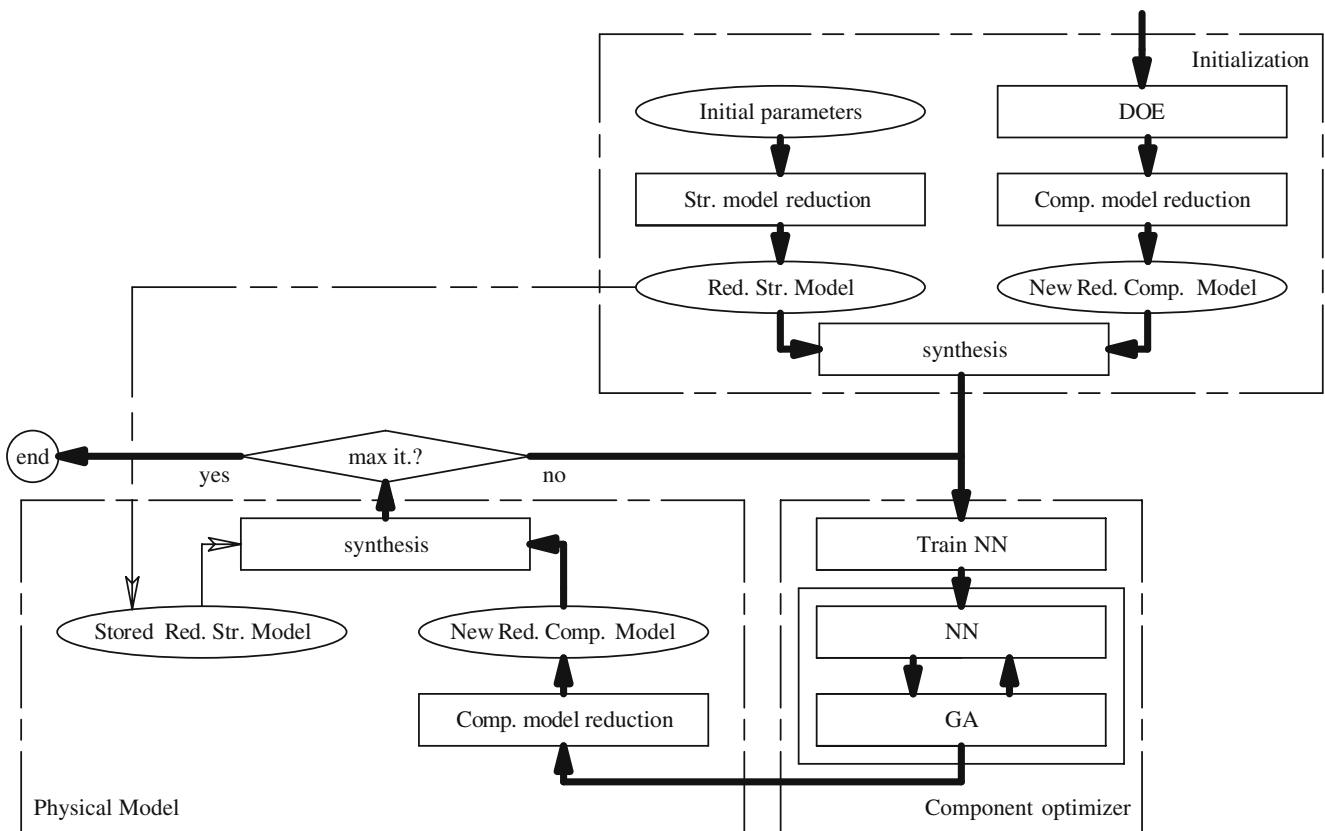

Fig. 7 Optimization of a component with CMS: first structure iteration

input–target pairs is no longer relevant. In the above strategy, the known input–target pairs are therefore discarded, and the second structure iteration must be started by generating new reduced models. This final problem is solved as follows. Instead of maintaining a data set of FE results, a library of reduced component models is maintained. These component models are used to generate up-to-date information in each component optimization.

Therefore, the proposed optimization strategy is initialized in the same way as in Fig. 7, except that all the reduced component models are saved. The component optimization is performed similarly, and the reduced component models are also saved for later use (see Fig. 8). After the maximum number of component iterations has been reached, the optimal component design variables and configurations are passed to the global level. Here, a convergence check is performed, and the new optimal parameters of all components are distributed to all other components. Next, a structure evaluation is performed for each component. Based on the optimal parameters of the other components, a new

remaining structure is generated and stored for use in the component optimization. It is combined with all of the reduced models that have been stored for this component using the synthesis process. A new NN is trained based on the results of the synthesis process, and a new component optimization begins. As the global–local iteration continues, the size of the library of stored component models increases such that the accuracy of the NN increases as well. After a large number of component optimizations have been performed, the NNs are very accurate at the beginning of the component optimization.

It is crucial to discuss a few details. First of all, in many applications, the remaining structure will be so large that the computational cost involved in generating a FE model and performing model reduction will be unacceptable. However, the reduced models of the other components can be generated separately and then combined to become remaining structure model. Before the first structure iteration, these models must be generated explicitly. In later structure iterations, the component designs used in the remaining structure
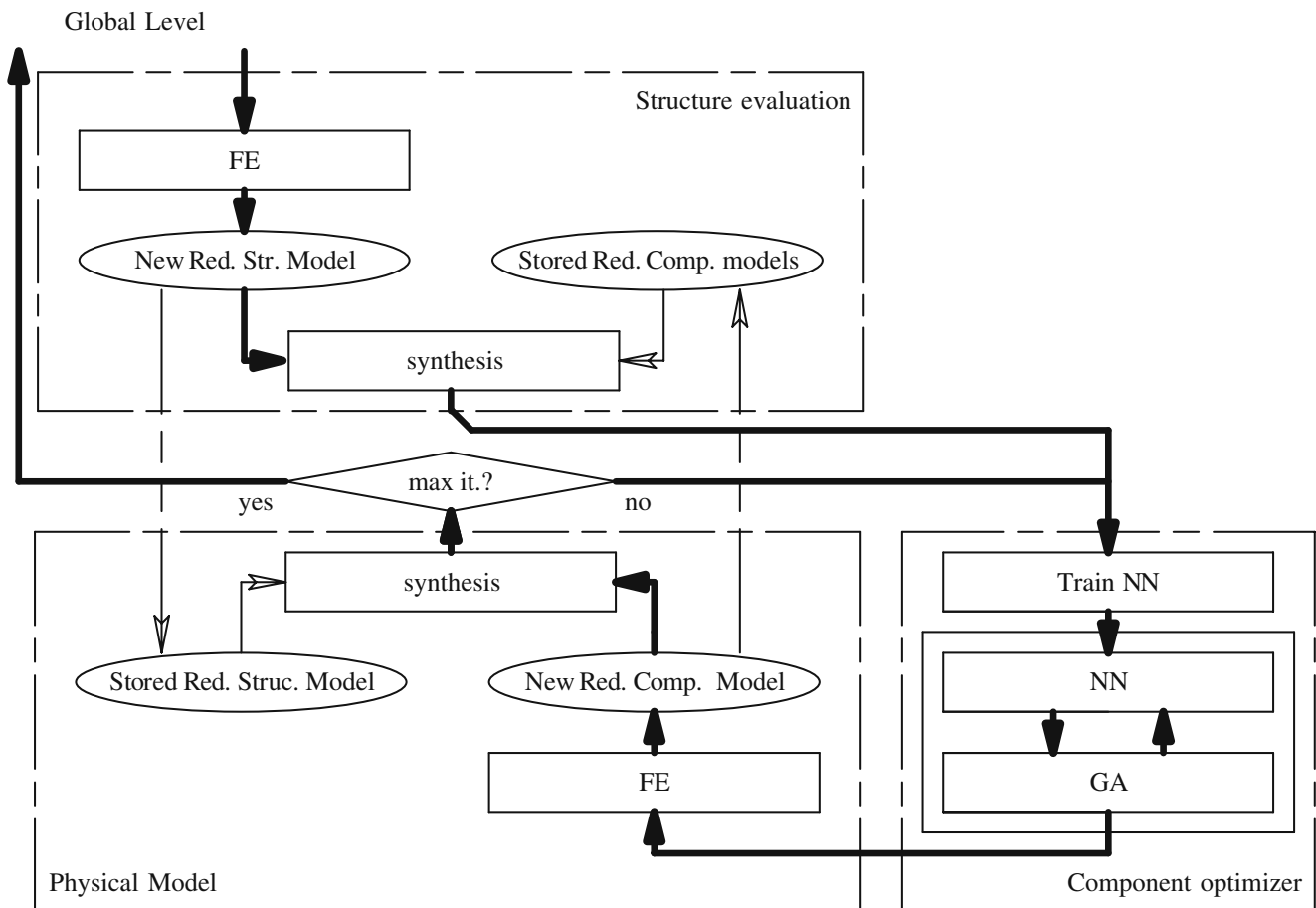

Fig. 8 Optimization of a component with CMS: second and later iterations

were optimal in the previous component optimization. As reduced models of these components are already stored, these models do not need to be generated again. Additionally, it can be efficient to perform another model reduction on the assembly of reduced models. Second, although the NNs cannot be shared between components, the reduced models can sometimes be shared depending on the application. This sharing leads to a larger data set and to more accurate NNs. Finally, it is noted that the use of different design configurations is possible with this technique, although it is not used in the current test problem.

4.2 Test problem

A small benchmark problem is introduced to test the global–local strategy. This problem is based on the placements of actuators and sensors in ASAC. In ASAC, sensors and actuators are connected to a plate-like noise source such that shape and amplitude of flexure can be controlled to minimize sound radiation. Simple design rules for placing the sensor–actuator pairs exist, but these have a number of distinct drawbacks (Oude Nijhuis 2003). Hence, a numerical optimizer is applied to select the sensor and actuator locations.

As a test problem, we use a structure consisting of three rectangular aluminium plates separated by transverse stiffeners (see Fig. 9a). The disturbance is generated by a rectangular patch of piëzoelectric material placed on the middle plate. On the other two plates, a sensor–actuator pair is placed to reduce the noise. The

actuators are piëzoelectric patches, and the sensors are accelerometers placed on the center of each actuator. The plate is placed in an infinite baffle—an acoustically hard surface that does not vibrate—which means the Rayleigh integral can be used for acoustical calculations. The reader is referred to Wind (2005, 2006) for more detailed information on the test problem.

The optimization problem consists of finding locations for the sensor–actuator pairs that minimize radiated sound power under a broadband disturbance in a frequency range that includes the first nine eigenfrequencies of the plate. In the global–local setting, there are two component optimization problems. Each consists of optimizing one plate, while keeping the rest of the structure constant (see Fig. 9b).

Finally, it is emphasized that this test problem is used to test the optimization strategy. From that point of view, a difference in radiated sound power of 1 dB can be a significant difference, although it may be negligible compared to modeling and discretization errors.

4.3 Results

The optimizations are performed on a single MS Windows machine with a clock speed of 1.67 GHz. Using a fixed number of 25 component iterations, the runtime has been in the order of magnitude of a few hours. The design variables used for FE are selected with a simple rule. In the first iteration, the GA result is used for FE analysis. If the error between the NN and FE result is larger, then a certain preset value then the NN is retrained and the GA is used again to estimate
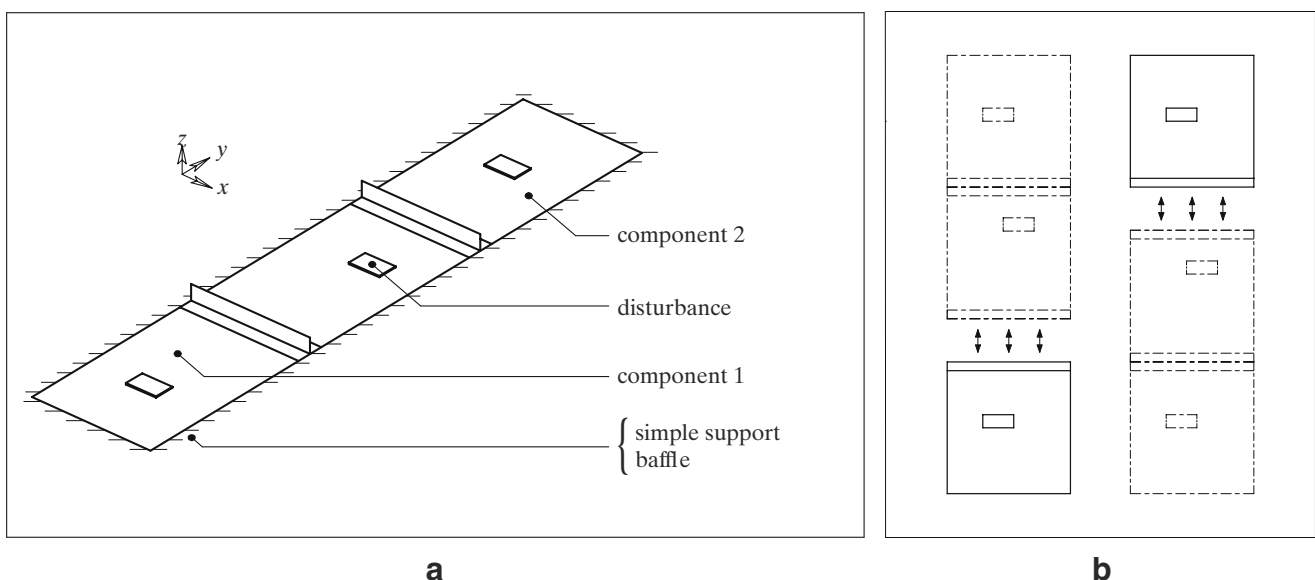

Fig. 9 The test problem: global design and global-local structure. **a** Global design of the test problem, **b** global–local problem. *Solid line* component, *dashed* remaining structure

the optimum in the next iteration. If the difference is smaller, then a random design is supplied to the next FE analysis. Several more advanced algorithms have been tested on a benchmark problem, but this algorithm demonstrated the best behavior in preventing the stagnation of the optimization process.

The result of an optimization with a broadband excitation in a frequency range of 50–320 Hz is given in Fig. 10a where the reference result and the optimization result of the global–local strategy are indicated by white and black boxes, respectively. The reference result has been obtained by applying the component optimization strategy on the global level. It can be seen that the resulting locations are different. As the structure is approximately symmetrical, the objective function has two minima with a very small difference in cost. The global–local strategy has outperformed the reference, but the margin is negligible, even for the purpose of testing the strategy. In Fig. 10a, the radiated sound power achieved in the optimization of component 1 is given as a function of the structure iteration number. It has remained constant from the second iteration onward, but the design variables switched more or less randomly between the final result and its mirror

image until structure iteration 12. After this iteration, no significant changes are observed.

A second optimization is performed with a harmonic excitation of 50 Hz. Again, a reference result is superimposed on the global–local optimization result, but results are almost indistinguishable in this case. The convergence of global–local iteration is more gradual in this case, and the optimal result is obtained at approximately structure iteration 15. It is important to note that the units considered for sound power in Fig. 10 are different.

Several other optimization runs have been performed. In the harmonic case, the optimization can exhibit limit cycling behavior or converge to the optimal result depending on the initial design. In cases where the excitation is broadband, the convergence behavior is more robust: Each optimization has converged to a point that is at least a local minimum from the global point of view.

4.4 Discussion

In the test problem, the components are optimized according to one global objective function, and all constraints are bounds. For this specific case, a theoretical study indicates that a converged result is at least a local minimum in the global scale, but convergence cannot be proven for practical cases (Wind 2005). No guarantees of optimality can be given if the constraints of one component depend on design variables of the other components.

5 Conclusions

A multilevel optimization strategy has been developed for the design optimization of complex structures. Based on the current research, the following conclusions can be drawn.

– The current strategy allows overnight optimization of the VTP center box of a commercial aircraft.
– Promising results have been obtained for a small benchmark problem in linear dynamics.
– The objective functions and constraints must be chosen with some care because the multilevel strategy can converge to strongly suboptimal solutions unless certain strict conditions are met.
– A comparison between the NN-based approximate model and other response surface methodologies as well as different selection algorithms requires further study.
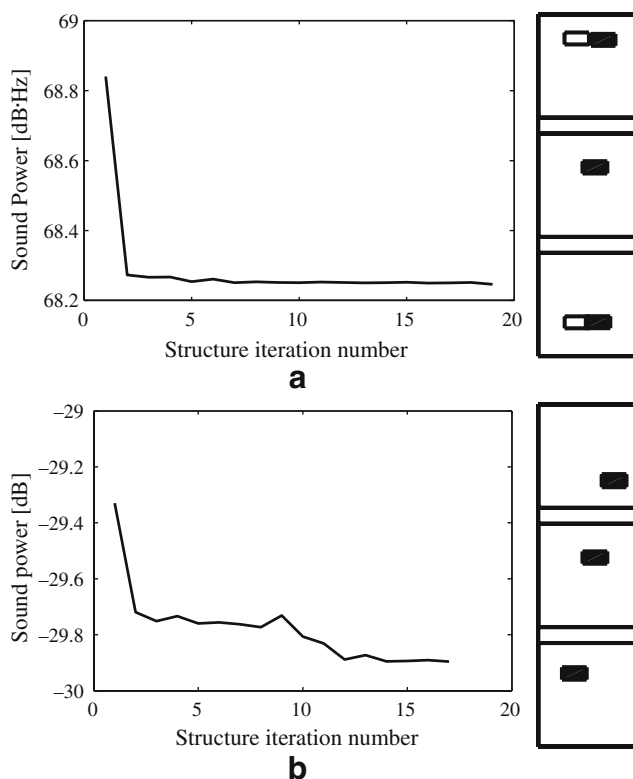

Fig. 10 Optimization results of the ASAC structure. **a** Results of broadband optimization, **b** results of harmonic optimization

References

Berke L (1992) Structural optimization of large structural systems by optimality criteria methods. NASA STI/Recon Technical Report N

Chen D, Hagan MT (1999) Optimal use of regularization and cross-validation in neural network modelling. International Joint Conference on Neural Networks. Neural Netw 2: 1275–1280

Coello CA (2002) Theoretical and numerical constraint-handling techniques used with evolu- tionary algorithms: A survey of the state of the art. Comput Methods Appl Mech Eng 191:1245–1287

Craig JRR Jr. (1981) Structural dynamics: an introduction to computer methods. Wiley, New York, 1981

Demuth MB, Hagan MT (1996) Neural network design. PWS Publishing Company

Entzinger JO (2004) Multilevel distributed structure optimization. In: Proceedings of ICAS2004, Yokohama, Japan

Haftka RT, Gürdal Z (1992) Elements of structural optimization. Kluwer, Dordrecht, The Netherlands

Hornik K (1989) Multilayer feedforward networks are universal approximators. Neural Netw (2):359–366

Jin R, Chen W, Simpson T (2001) Comparative studies of metamodelling techniques under multiple modelling criteria. Struct Multidisc Optim 23:1–13

Kirsch U, Bogomolni M, van Keulen F (2005) Efficient finite difference design sensitivities. AIAA J 43(2):399–405

Mackay DJC (1992a) A practical framework for backprop networks. Neural Comput (4):448–472

Mackay DJC (1992b) Bayesian interpolation. Neural Comput (4):415–447

Menhaj MB, Hagan MT (1994) Training feedforward networks with the marquardt algorithm. IEEE Transac Neural Netw 5(6):989–993

Ong YS, Nair PB, Keane AJ, Wong KW (2004) Surrogate-assisted evolutionary optimization frameworks for high-fidelity engineering design problems. In: Jin Y (ed) Knowledge incorporation in evolutionary computation. Springer, pp 307–332 (Studies in Fuzziness and Soft Computing 167)

Oude Nijhuis MHH (2003) Analysis tools for the design of active structural control systems. PhD dissertation, University of Twente

Ruijter W, Spallino R, Warnet L, Boer A de (2003) Optimization of composite panels using neural networks and genetic algorithms. In: Proceedings of the 2nd MIT conference on computational fluid and solid mechanics, Cambridge, MA

Ruijter W, Spallino R (2003) Feedback-based neural networks in the structural optimization of aerospace structures. In: Proceedings of CC2003, Warsaw, Poland

Simpson T, Peplinski J, Koch P, Allen J (2001) Metamodels for computer-based engineering design: survey and recommendations. Eng Comput 17:129–150

Sobieszczanski J, Loendorf D (1972) A mixed optimization method for automated design of fuselage structures. J Aircr 9(12):805–811

Uri K (2004) Design-oriented analysis of structures, vol 95. E-book. Springer, The Netherlands, pp 187–205

Wind JW (2005) Silent components fast, a global–local optimization method for dynamic problems. Graduate thesis, University of Twente

Wind JW (2006) A global–local optimization method for problems in structural dynamics. In: Proceedings of ICAS 2006, Hamburg, Germany

Yates K, Gürdal, Thangjitham S (1994) Multilevel optimization of space trusses using continuum modelling. Struct Multidisc Optim 7:176–183