CAS 3211
1 August 2003 Disk used
ARTICLE IN PRESS
No. of Pages 15, DTD = 4.3.1

Available online at www.sciencedirect.com
SCIENCE DIRECT®

PERGAMON

Computers and Structures xxx (2003) xxx–xxx

Computers & Structures

www.elsevier.com/locate/compstruc

Global optimization by coupled local minimizers and its application to FE model updating

Anne Teughels [a,*], Guido De Roeck [a], Johan A.K. Suykens [b]

[a] *Department of Civil Engineering, Division of Structural Mechanics, Katholieke Universiteit Leuven, Kasteelpark Arenberg 40, B-3001 Heverlee, Belgium*

[b] *Department of Electrical Engineering, Katholieke Universiteit Leuven, ESAT-SISTA, Kasteelpark Arenberg 10, B-3001 Heverlee, Belgium*

Received 2 August 2002; accepted 3 July 2003

Abstract

Coupled local minimizers (CLM) is a new method applicable to global optimization of functions with multiple local minima. In CLM a cooperative search mechanism is set up using a population of local optimizers, which are coupled during the search process by synchronization constraints. CLM is characterised by a relative fast convergence since the local optimizers are gradient-based. The combination of both, the coupled parallel strategy and the fast convergence, offers an efficient global optimization algorithm. In the paper the CLM method is described and is illustrated with a test function. Due to the simultaneous and coupled search of a whole population of optimizers, CLM is able to find the global minimum of the test function. Next, CLM is successfully applied to FE model updating using experimental modal data. In an example the damage pattern of a reinforced concrete beam is identified.

Keywords: Global optimization; Coupled local minimizers; FE model updating; Damage assessment; Modal parameters

1. Introduction

FE models are widely used to predict the dynamic properties of structures. However, the results obtained from a FE model often differ from the experimental results obtained from a vibration test. This discrepancy can be caused by both, errors in the experimental data and errors in the analytical model. Despite the presence of experimental errors, it is generally assumed that the experimental data are a better representation of how the structure behaves than are the predictions from the initial FE model. Consequently, the FE model is corrected in a FE model updating procedure, in which the uncertain model properties are adjusted such that the numerical predictions correspond as closely as possible to the measured data [1,2]. In FE model updating using experimental modal data, an optimization problem is solved with an objective function defined by the discrepancies between the numerical and experimental modal parameters (natural frequencies and mode shapes). The function can be quite irregular and can contain several local minima. The updating variables are the correction factors of the uncertain model properties.

The success of the application of the updating method depends on the accuracy of the numerical FE model, the quality of the modal test, the definition of the optimization problem and the mathematical capabilities of the optimization algorithm. Conventional gradient-based mathematical programming (MP) methods have a satisfactory convergence rate, but they may get stuck into any local minimum depending on the starting point [3–

* Corresponding author. Tel.: +32-16-321-665; fax: +32-16-321-988.
E-mail address: anne.teughels@bwk.kuleuven.ac.be (A. Teughels).
URL: http://www.bwk.kuleuven.ac.be/bwm.

0045-7949/$ - see front matter © 2003 Published by Elsevier Ltd.
doi:10.1016/S0045-7949(03)00313-4

5]. The basic MP method is the Newton method which makes use of the local curvature of the original function to build an approximate quadratic model function. This model function is calculated in each point of the iterative process and minimized to obtain the consecutive point. The process ends when the minimum is reached. Other local optimization methods are quasi-Newton, conjugate gradient, sequential quadratic programming, augmented Lagrangian method, etc.

The global search methods, such as genetic algorithms (GA) [6] and simulated annealing (SA) [7], are in general more robust, i.e. the choice of the starting position has little influence on the final results, and they present a better global behaviour [8]. However, both algorithms share the disadvantage of requiring a large number of function evaluations since they are based on probabilistic searching without the use of any gradient information. They are both derived from analogies with natural phenomena: GA with natural evolution and SA with a thermodynamic cooling process.

Recently, a method of coupled local minimizers (CLM) has been proposed by Suykens et al. [9,10]. Within the framework of optimization problems the CLM method can be used for global optimization problems. The method couples multiple local optimization runs in order to create interaction and information exchange between the search points. A relative fast convergence is maintained, due to the derivative information used in the local algorithms. Furthermore the global minimum is expected to be found more easily, since multiple search points are used simultaneously.

This paper deals with the CLM algorithm, which is originally developed as a continuous-time optimization method in the framework of neural networks [9,10]. This paper proposes a new implementation of the algorithm, such that it can be used as a numerical, iterative, global optimization method that generates discrete steps in the design space instead of continuous-time variations of the design variables. The theoretical background of CLM and its implementation are described in the paper and the method is illustrated with a test function containing multiple local minima. The advantages of CLM over conventional local optimization algorithms (multistart local optimization) are shown. Next, CLM is applied to FE model updating, used for the damage identification of a reinforced concrete beam. The damage identification is performed in two updating steps in order to adjust the FE model to the reference and the damaged state of the beam respectively. The damage pattern of the concrete beam is identified successfully with the CLM method.

This paper is organized as follows. The global search methods, GA and SA, are briefly reviewed in Section 2. In Section 3 we present the theory of the CLM algorithm and its implementation. In Section 4 we illustrate the CLM algorithm with a test function. In Section 5 CLM is applied to FE model updating. In Section 6 conclusions are made.

An explicit comparison between the CLM method and the GA or SA methods for the FE model updating application, could be an interesting topic of a benchmark study.

2. Global search methods: genetic algorithms and simulated annealing

The basic GA was suggested by Holland [6]. It is based on natural evolution and its concept of survival of the fittest. The algorithm acts on a population of chromosomes, defined by binary strings. Each chromosome is a representation of a design vector and its fitness value is given by the objective function. The GA consists of generating a new population of chromosomes from the old population using three randomized operators that mimic those of natural evolution: selection, crossover and mutation [8,11]. In the first operation, a number of chromosomes are selected such that those with greater fitness have a higher probability of selection. A very fit individual may have several changes to be selected. Some of the selected chromosomes are then randomly paired together. Both chromosomes in each pair swap information beyond a crossover point which is randomly chosen along the binary string. This operator has the potential to join successful genetic fragments together to form fitter individuals. Mutation randomly flips some of the bits in a single chromosome, meant to reintroduce genetic information that has been lost from the population. The average fitness of the generation successively increases and the process is stopped by a suitable convergence criterion. The capability of finding the global minimum is mainly due to the simultaneous search by a whole population of design points using randomized operators, such that the search space is widely explored. Moreover, the information exchange between selected pairs directs the process towards the optimal point.

Kirkpatrick et al. [7] proposed SA as a powerful global search method. The method gets its name from the physical process whereby the temperature of a physical system is raised to a melting point and then slowly and discretely lowered. The substance attains its lowest energy provided that it acquires the least possible energy at each temperature during the successive cooling process. This concept of thermal equilibrium is mimicked in SA [12,13] by reducing the objective function to a reasonably low value correlated with the 'temperature' at each state of the optimization process. Global optimum is reached through a search within randomly generated configurations in the neighbourhood of a single design state. If the new point has a smaller value for the objective function (downhill move), this point is accepted and replaces the old one. However, in the opposite case (uphill move), the candidate design may either be rejected or accepted depending on a control parameter (similar to temperature in the annealing process) which is reduced slowly so as not to get trapped in a local minimum. At initial stages of optimization (at high temperatures), the probability of accepting uphill moves is higher. Later on (at low temperatures), it becomes smaller so that in the end the designs having higher cost are almost never accepted. Various implementations of SA exist, based on different cooling schedules and neighbourhood functions [8]. The success of SA lies in the fact that a random choice of a candidate point and the occasional acceptance of uphill moves, avoid getting stuck in a local minimum.

Both GA and SA are frequently used in structural optimization problems [13–18].

3. Coupled local minimizers

In the method of CLM [9,10] a cooperative search mechanism is set up, which combines the advantage of the local gradient-based algorithms (fast convergence) with the global approach of GA (parallel strategy and information exchange). A population of search points is used, initially spread over the search space. The derivative information in each of these points directs the global search process. Instead of performing separate, independent searches from each of these points (which is the case in multistart local optimization [1]), the local optimizers are coupled during the search process by constraints that enforce the global search process to converge towards one point. In this way a cooperative search mechanism is set up that aims to perform better than multistart local optimization (Fig. 1).

The method is implemented as a minimization problem in which the average objective function value—i.e. the function value averaged over all the search points—is minimized. The whole population of search points look for the minimum of this average function using derivative information. And in order to couple the (local) search runs, the search points are subjected to pairwise synchronization constraints that enforce them to end in the same final point. In this way the constraints realize an information exchange within the population.

In this paper [2] the CLM technique is implemented with the augmented Lagrangian method, which is a MP method for constrained optimization [3,5]. The augmented Lagrangian function $\mathscr{L}_\mathrm{A}$ is defined by the average objective function of the population together with the synchronization constraints between the individual local minimizers. A standard unconstrained optimization algorithm minimizes $\mathscr{L}_\mathrm{A}$.

3.1. Augmented Lagrangian method

Consider the minimization of an objective function $f(\boldsymbol{x})$ with equality constraints $h_i(\boldsymbol{x})$ with $\boldsymbol{x} \in \mathbb{R}^n$. The augmented Lagrangian function is defined as [3,5]

$$\mathscr{L}_\mathrm{A}(\boldsymbol{x}, \boldsymbol{\lambda}) = f(\boldsymbol{x}) + \sum_i \lambda_i h_i(\boldsymbol{x}) + \frac{\gamma}{2} \sum_i h_i^2(\boldsymbol{x}), \tag{1}$$

where λ_i and γ are the Lagrange multiplier estimates and the penalty parameter respectively. The different terms in $\mathscr{L}_\mathrm{A}$ are the objective function, the hard and the soft constraints respectively.

In each main iteration k, the function $\mathscr{L}_\mathrm{A}(\boldsymbol{x}, \boldsymbol{\lambda}_k)$ is minimized with respect to $\boldsymbol{x}$ to find $\boldsymbol{x}_k^*$. The values of $\boldsymbol{\lambda}_k = (\lambda_1, \lambda_2, \ldots, \lambda_i, \ldots)_k$ are then updated to start the next main iteration. The update formula for each λ_i is [3,5]

$$(\lambda_i)_{k+1} = (\lambda_i)_k + \gamma h_i(\boldsymbol{x}_k^*). \tag{2}$$

The process continues until the optimal $\boldsymbol{\lambda}^*$ are found, which are the Lagrange multipliers at $\boldsymbol{x}^*$.

3.2. Coupled local minimizers method

Consider now the unconstrained minimization of the objective function $f(\boldsymbol{x})$. In CLM, a population is used consisting of q local minimizers, whose average cost is defined as

$$\langle f \rangle = \frac{1}{q} \sum_{i=1}^{q} f(\boldsymbol{x}^{(i)}). \tag{3}$$

Pairwise synchronization constraints are applied to the design vectors $\boldsymbol{x}^{(i)}$ (= vectors of variables), resulting in a constrained minimization problem:

$$\min_{\boldsymbol{x}^{(i)} \in \mathbb{R}^n} \langle f \rangle \text{ such that } \boldsymbol{x}^{(i)} - \boldsymbol{x}^{(i+1)} = 0 \tag{4}$$

for $i = 1, 2, \ldots, q$ and with boundary condition $\boldsymbol{x}^{(q+1)} = \boldsymbol{x}^{(1)}$.

One defines the augmented Lagrangian function:

$$\mathscr{L}_\mathrm{A}(\mathbf{x}, \boldsymbol{\Lambda}) = \frac{\eta}{q} \sum_{i=1}^{q} f(\boldsymbol{x}^{(i)}) + \sum_{i=1}^{q} \langle \boldsymbol{\lambda}^{(i)}, [\boldsymbol{x}^{(i)} - \boldsymbol{x}^{(i+1)}] \rangle + \frac{\gamma}{2} \sum_{i=1}^{q} \| \boldsymbol{x}^{(i)} - \boldsymbol{x}^{(i+1)} \|^2 \tag{5}$$

with $\mathbf{x} = [\boldsymbol{x}^{(1)}; \ldots; \boldsymbol{x}^{(q)}]$ and $\boldsymbol{\Lambda} = [\boldsymbol{\lambda}^{(1)}; \ldots; \boldsymbol{\lambda}^{(q)}]$, $(\boldsymbol{x}^{(i)}, \boldsymbol{\lambda}^{(i)} \in \mathbb{R}^n)$. $\langle \cdot, \cdot \rangle$ denotes the inner product (for the

[1] Multistart local optimization consists in performing a number of local optimization runs, each starting from another point, but sequentially, so without any coupling.

[2] CLM is originally developed in the area of neural networks. In [9,10] a Lagrange programming network is therefore considered for the continuous-time optimization.

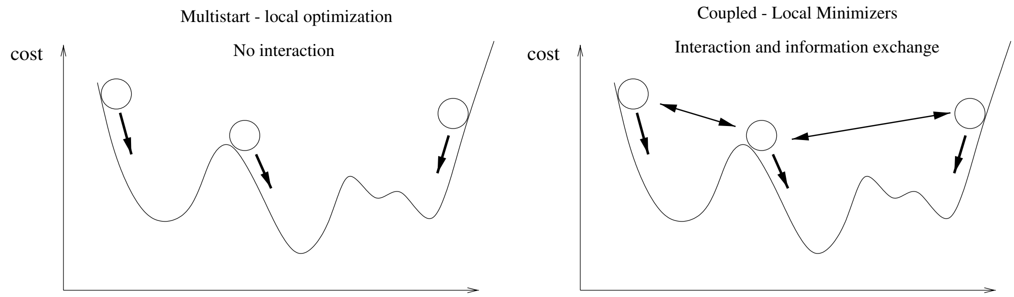

Fig. 1. Instead of independent runs in multistart local optimization, local minimizers are coupled in CLM.

hard constraints) and $\|.\|$ the Euclidean norm of a vector (for the soft constraints). η is a weighting factor of the average objective function.

The main idea is to impose upon the multiple design vectors to reach the same final position. When the initial states of the design vectors are located in different valleys, they are enforced to take a decision about which valley to choose. If the parameters η and γ are chosen appropriately, an improved solution is obtained, which is usually the global minimum.

The number of q needed to achieve a good performance, depends on the complex shape of the surface or typically on the number of local minima per volume in the search space.

3.3. Implementation of CLM

In this paper [3] we implement the CLM algorithm with a standard Trust Region Newton method [5] for minimizing $\mathscr{L}_{\mathrm{A}}(\mathbf{x}, \boldsymbol{\Lambda}_k)$ with respect to $\mathbf{x}$. In each sub-iteration s, a quadratic approximation $m(\mathbf{p})$ of $\mathscr{L}_{\mathrm{A}}$ at the current population $\mathbf{x}_s$ has to be minimized within a trust region Δ_s. The quadratic model $m(\mathbf{p})$ is defined by the truncated Taylor series of $\mathscr{L}_{\mathrm{A}}$:

$$\min_{\mathbf{p}}\ m(\mathbf{p}) = \mathscr{L}_{\mathrm{A}} + [\nabla\mathscr{L}_{\mathrm{A}}]^{\mathrm{T}}\mathbf{p} + \tfrac{1}{2}\mathbf{p}^{\mathrm{T}}[\nabla^2\mathscr{L}_{\mathrm{A}}]\mathbf{p}, \quad \text{such that } \|\mathbf{p}\| \leqslant \Delta, \tag{6}$$

where $\mathbf{p}$ denotes a step-vector from $\mathbf{x}_s$ and where $\mathscr{L}_{\mathrm{A}}$, $\nabla\mathscr{L}_{\mathrm{A}}$ and $\nabla^2\mathscr{L}_{\mathrm{A}}$ are the values of the function, the gradient and the Hessian of $\mathscr{L}_{\mathrm{A}}$ at $\mathbf{x}_s$ respectively.

Since we assume that each local minimizer is independent of the values of the other minimizers, we have: (for $i = 1, \ldots, q$)

$$\nabla_{x^{(i)}}\mathscr{L}_{\mathrm{A}} = \frac{\eta}{q}\nabla_{x^{(i)}} f(\boldsymbol{x}^{(i)}) - \boldsymbol{\lambda}^{(i-1)} + \boldsymbol{\lambda}^{(i)} - \gamma[\boldsymbol{x}^{(i-1)} - \boldsymbol{x}^{(i)}] + \gamma[\boldsymbol{x}^{(i)} - \boldsymbol{x}^{(i+1)}], \tag{7}$$

$$\nabla^2_{x^{(i)}}\mathscr{L}_{\mathrm{A}} = \frac{\eta}{q}\nabla^2_{x^{(i)}} f(\boldsymbol{x}^{(i)}) + 2\gamma I, \tag{8}$$

$$\nabla^2_{x^{(i)}x^{(i-1)}}\mathscr{L}_{\mathrm{A}} = -\gamma I, \tag{9}$$

$$\nabla^2_{x^{(i)}x^{(i+1)}}\mathscr{L}_{\mathrm{A}} = -\gamma I, \tag{10}$$

to be included in the gradient vector or the band-structured Hessian matrix. I denotes the identity matrix $(n \times n)$. The boundary constraints are: $\boldsymbol{x}^{(0)} = \boldsymbol{x}^{(q)}$, $\boldsymbol{x}^{(q+1)} = \boldsymbol{x}^{(1)}$.

Since a Newton-based method is used, the search process in CLM is carried out with a high convergence speed. Furthermore, the convergence is enforced by the use of a Trust Region strategy.

Additionally, bound constraints on the design vectors $\boldsymbol{x}^{(i)}$ can be added. Although these constraints are not really necessary, because of the proper restriction of the trust region, they can be desirable in order to impose specific limitations.

The CLM algorithm is implemented in the optimization toolbox of MATLAB [19]. The Trust Region Newton method, used for the minimization of $\mathscr{L}_{\mathrm{A}}$ with respect to $\mathbf{x}$, is applied by means of the command *fmincon*, for which the 'Trust Region' option is chosen.

3.4. Choice of η, γ—normalization

Since the tuning parameters η and γ are problem dependent, it is difficult to determine them a priori or in a general way. Moreover, they enable the analyzer of each particular problem to direct the process, just by adjusting them (see Section 4.1). The difficulty of selection of values for these parameters is typical for global search methods (e.g. GA and SA), but at the same time they provide the capability of finding the global minimum. This is in contrast with local MP methods, which are fully determined but can only find local minima.

[3] In [9,10] a steepest descent method is used for solving the Lagrange programming network. Our implementation with the Trust Region Newton method is meant for realizing a faster convergence and for obtaining a robust optimization process.

However, in order to generalize the CLM method as much as possible, the objective function and the synchronization constraints ($\Delta x_j^{(i)} = x_j^{(i)} - x_j^{(i+1)}$) in $\mathscr{L}_\mathrm{A}$ are normalized:

$$f_\mathrm{n} = \frac{f+t}{sc_\mathrm{f}} \Rightarrow 0 \leqslant f_\mathrm{n} \leqslant 1, \tag{11}$$

$$\Delta x_{j_n}^{(i)} = \frac{\Delta x_j^{(i)}}{sc_{\mathrm{c}j}} \Rightarrow 0 \leqslant |\Delta x_{j_n}^{(i)}| \leqslant 1. \tag{12}$$

The inequalities in Eqs. (11) and (12) should hold only on that part of the search space that will be tried out during the process. Consequently, the translation value t and the factors $sc_\mathrm{f}, sc_{\mathrm{c}j}$ are not unique and can only be estimated. The following expressions can be used when choosing the normalization parameters: $t = |\min(0, f_{\min})|$; $sc_\mathrm{f} = f_{\max} + t$; $sc_{\mathrm{c}j} = |x_{j,\mathrm{upper}} - x_{j,\mathrm{lower}}|$; with $f_{\min}, f_{\max}$ denoting the minimum, maximum function value encountered during the process and $x_{j,\mathrm{upper}}, x_{j,\mathrm{lower}}$ the upper and lower boundary of design variable x_j. With this approach a normalized objective function f_n is minimized but still with respect to the unscaled design vector $\boldsymbol{x}$. The formulas for $\mathscr{L}_\mathrm{A}$, $\nabla\mathscr{L}_\mathrm{A}$ and $\nabla^2\mathscr{L}_\mathrm{A}$ in Eqs. (6)–(8) and the update formula for $\boldsymbol{\lambda}^{(j)}$ are accordingly adjusted. Due to the normalization the relative weights of the different terms in $\mathscr{L}_\mathrm{A}$ are less dependent on the characteristics of each particular minimization problem.

4. Test function

To illustrate the CLM method, a two-dimensional test function is minimized:

$$f(\boldsymbol{x}) = \sum_{j=1}^{2} 0.01\left((x_j + 0.5)^4 - 30x_j^2 - 20x_j\right) \quad \text{with } -6 \leqslant x_j \leqslant 6. \tag{13}$$

In Fig. 2 the test function is visualized. There are four local minima. One of them is the global minimum, located at (−4.454;−4.454).

The applied normalization parameters are: $t = 6$; $sc_\mathrm{f} = 21$; $sc_{\mathrm{c}1} = sc_{\mathrm{c}2} = 12$.

A CLM run is carried out with a population of 8 ($= q$) local minimizers. Their initial values are randomly spread in the search space (Fig. 3a). The initial values in $\boldsymbol{\lambda}^{(i)}$ are randomly chosen in the interval [−1;1], for reason of generality. The tuning parameters are: $\eta = 3$ and $\gamma = 0.3$. All the eight minimizers end up in the global minimum (Fig. 3b). Even if all the minimizers are initially situated in the valley of a local minimum (Fig. 4a, $q = 5$), the CLM method finds the global minimum (Fig. 4b). For this case also five independent local optimization runs were carried out starting from each point separately and they all ended up in the same local minimum, different from the global minimum. This illustrates that instead of multistart local optimization consisting of independent runs, the search process is clearly improved with CLM by coupling the local optimizers during the process. Furthermore, in CLM a Trust Region approach is used in order to be able to minimize a nonconvex function. This is essential, since a nonconvex augmented Lagrangian function makes it possible to escape from a local minimum, as it is the case in Fig. 4.

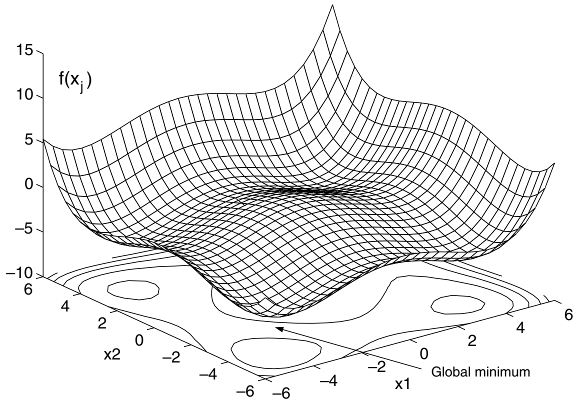

Fig. 2. Test function with four local minima, one of which is the global minimum (surface plot above a contour plot).

4.1. Influence of η, γ

In order to detect the global minimum, the search process can be influenced by the tuning parameters η and γ. Fig. 3c shows the search path corresponding with the parameter values: $\eta = 3$ and $\gamma = 0.3$, as used in previous paragraph. About 70 iterations are performed before converging to the global minimum. By increasing γ, more weight is given to the soft constraints in $\mathscr{L}_\mathrm{A}$ and consequently the convergence rate is improved. But one should be careful to choose γ not too high since in this case the CLM run would end up in the local minimum that is closest to the geometrical center of gravity of the population (Fig. 5a, 20 iterations). A low γ value on the other hand leads to more exploration in the search domain, but consequently decreases the speed of convergence. Many iterations are necessary before the convergence criterion is satisfied (Fig. 5b, 450 iterations). By reducing γ and η, still much exploration is carried out, but now the soft constraints are relatively more stringent than in previous case, which results in fewer—but still many—iterations (Fig. 5c, 350 iterations).

Appropriate values for the tuning parameters are problem dependent. Since in real optimization problems, the global minimum is not known beforehand and

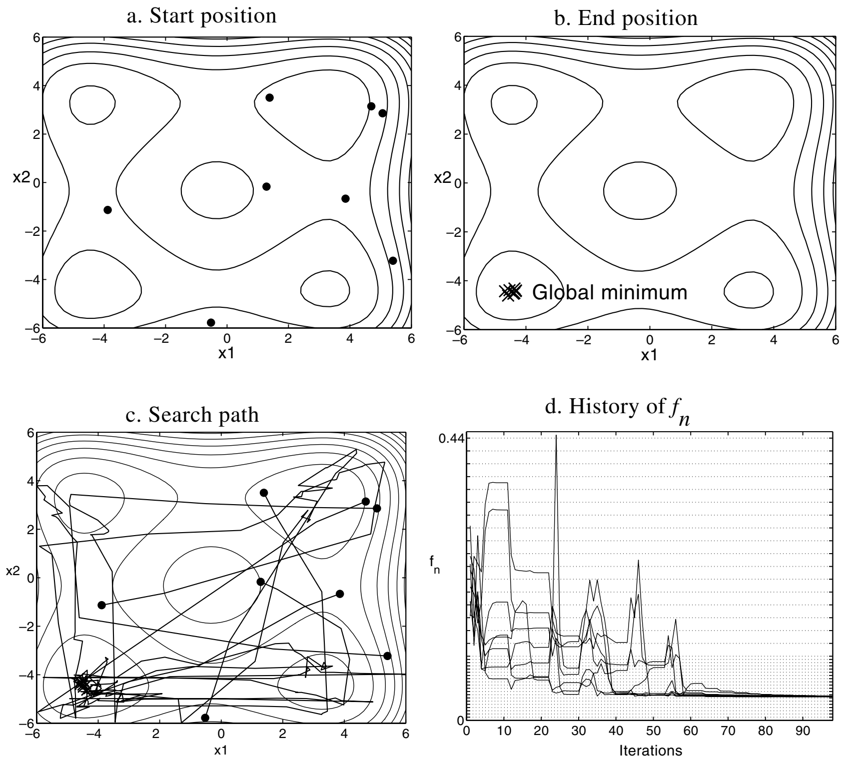

Fig. 3. A CLM run with a population of eight searching points, which are initially randomly spread over the whole search space (a) and end up in the global minimum (b). The search path of all local searchers is plotted in (c) on the contour plot of the normalized function f_n (●: start point, ×: end point). In (d) the history of the f_n-values evaluated by each local searcher is shown.

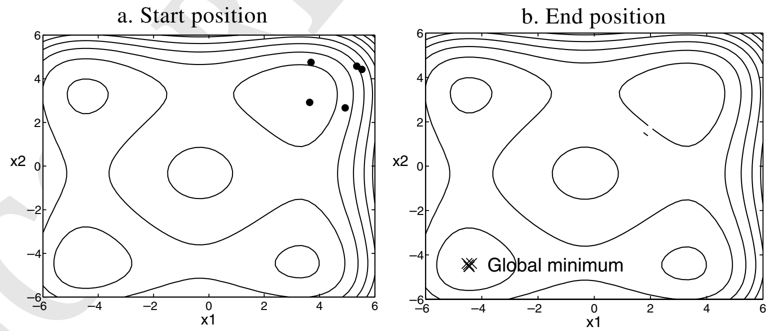

Fig. 4. A CLM run with a population of five searching points, which are initially localized close to a local minimum (a) and end up in the global minimum (b).

therefore it is not sure whether the result is a local or the global minimum, one should look at the history of the (normalized) objective function f_n evaluated by each search point. Fig. 3d shows that the final objective function value, i.e. the one evaluated at the final solution, is the least one of all values encountered during the process, i.e. when the search points explored the search space. If, on the other hand, lower f_n values would appear during the history, the analyzer knows that he has to adjust the tuning parameters until the final f_n value is the least one.

5. FE model updating

In this section we discuss the application of CLM to FE model updating using measured modal data. First the general updating procedure is explained. Next, we

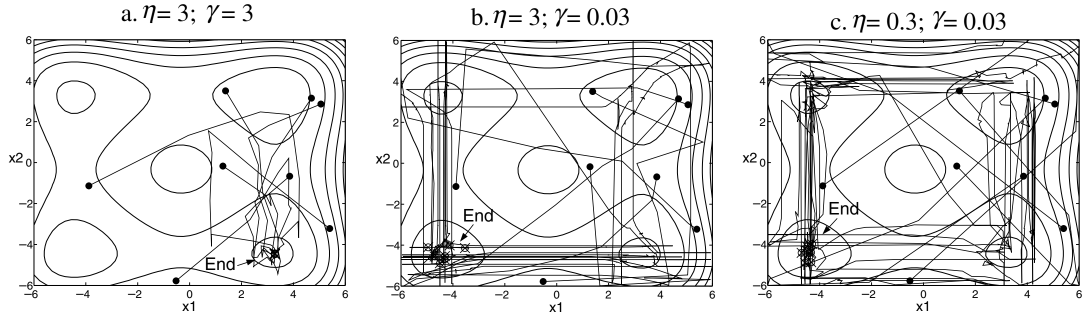

Fig. 5. Search paths of three CLM optimization runs (a–c) with different values for η and γ, each drawn on the contour plot of the test function.

will illustrate the CLM application by identifying the damage pattern of a reinforced concrete beam.

5.1. General procedure

In FE model updating one aims to identify the uncertain properties of a structure by minimizing the discrepancies between the experimental vibration data, extracted from a dynamic test on the structure, and those computed with the numerical FE model. Therefore, an optimization problem is solved in which the objective function contains the differences between the experimental and numerical modal data (natural frequencies and mode shapes) [1,2]. The updating variables are the uncertain model properties.

The cost function is stated as a nonlinear least-squares problem [20]:

$$f(\boldsymbol{a}) = \frac{1}{2}\|\boldsymbol{r}(\boldsymbol{a})\|^2 = \frac{1}{2}\left\| \begin{matrix} \boldsymbol{r}_{\mathrm{f}}(\boldsymbol{a}) \\ \boldsymbol{r}_{\mathrm{s}}(\boldsymbol{a}) \end{matrix} \right\|^2 \tag{14}$$

with

$$r_{\mathrm{f}}(\boldsymbol{a}) = \frac{\omega_j^2(\boldsymbol{a}) - \tilde{\omega}_j^2}{\tilde{\omega}_j^2} \quad \text{with } \omega_j = 2\pi\nu_j, \tag{15}$$

$$r_{\mathrm{s}}(\boldsymbol{a}) = \frac{\phi_j^l(\boldsymbol{a})}{\phi_j^r(\boldsymbol{a})} - \frac{\tilde{\phi}_j^l}{\tilde{\phi}_j^r}. \tag{16}$$

The residual vector $\boldsymbol{r}: \mathbb{R}^n \to \mathbb{R}^m$ contains the discrepancies in eigenfrequencies ν_j (Eq. (15)) and in mode shapes $\boldsymbol{\phi}_j$ (Eq. (16)). l and r denote an arbitrary and a reference degree of freedom (DOF) respectively. The vector $\boldsymbol{a} \in \mathbb{R}^n$ represents the set of uncertain model properties. [4] The experimental modal parameters, $\tilde{\nu}_j$ and $\tilde{\boldsymbol{\phi}}_j$, are obtained from a modal test. Only the translation DOFs of the mode shapes can be measured.

Relative differences are taken in r_{f} in order to obtain a similar weight for each frequency residual. In r_{s} the mode shapes are scaled to one in a reference point r, since the numerical and experimental mode shapes can be scaled differently. As in civil engineering, measurements are often conducted in operational conditions, which means that the exciting forces (coming from wind, traffic,...) are unknown, an absolute scaling of the mode shapes is not possible. The reference point r should be chosen at the DOF with the largest magnitude, or at least at one with a large magnitude.

The updating parameters are the uncertain physical properties of the numerical model, determined on elemental level. Instead of the absolute value of each uncertain variable X^{e}, a fractional correction factor a^{e} is used, with respect to the initial value X_0^{e}:

$$a^{\mathrm{e}} = -\frac{X^{\mathrm{e}} - X_0^{\mathrm{e}}}{X_0^{\mathrm{e}}} \Rightarrow X^{\mathrm{e}} = X_0^{\mathrm{e}}(1 - a^{\mathrm{e}}). \tag{17}$$

The gradient and the Hessian of $f(\boldsymbol{a})$ are:

$$\nabla f(\boldsymbol{a}) = \sum_{j=1}^{m} r_j(\boldsymbol{a})\nabla r_j(\boldsymbol{a}) = \mathbf{J}_a(\boldsymbol{a})^{\mathrm{T}}\boldsymbol{r}(\boldsymbol{a}), \tag{18}$$

$$\begin{aligned}\nabla^2 f(\boldsymbol{a}) &= \mathbf{J}_a(\boldsymbol{a})^{\mathrm{T}}\mathbf{J}_a(\boldsymbol{a}) + \sum_{j=1}^{m} r_j(\boldsymbol{a})\nabla^2 r_j(\boldsymbol{a}) \\ &\approx \mathbf{J}_a(\boldsymbol{a})^{\mathrm{T}}\mathbf{J}_a(\boldsymbol{a})\end{aligned} \tag{19}$$

with $\mathbf{J}_a$ the Jacobian matrix, containing the first partial derivatives of the residuals r_j $(= r_{\mathrm{f}}$ and $r_{\mathrm{s}})$ with respect to $\boldsymbol{a}$. The Hessian is approximated with the first order term in Eq. (15) as it is the case in most nonlinear least-squares methods [5]. The approximation is equivalent with a linearization of the residual functions in $\boldsymbol{a}$.

The first partial derivatives of each frequency residual r_{f} (Eq. (15)) and mode shape residual r_{s} (Eq. (16)) with respect to the correction parameters $\boldsymbol{a}$ are:

[4] Note that the symbol $\boldsymbol{a}$ is used in this section on FE model updating, whereas the symbol $\boldsymbol{x}$ is used in the general mathematical formulations of previous sections.

$$\frac{\partial r_{\mathrm{f}}}{\partial a^{\mathrm{e}}} = \frac{1}{\tilde{\omega}_j^2}\frac{\partial \omega_j^2}{\partial a^{\mathrm{e}}}, \tag{20}$$

$$\frac{\partial r_{\mathrm{s}}}{\partial a^{\mathrm{e}}} = \frac{1}{\phi_j^r}\frac{\partial \phi_j^l}{\partial a^{\mathrm{e}}} - \frac{\phi_j^l}{(\phi_j^r)^2}\frac{\partial \phi_j^r}{\partial a^{\mathrm{e}}}. \tag{21}$$

The modal sensitivities in Eqs. (20) and (21) are calculated using the formulas of Fox and Kapoor [21]. If only stiffness parameters have to be corrected, these formulas are simplified to

$$\frac{\partial \omega_j^2}{\partial a^{\mathrm{e}}} = \boldsymbol{\phi}_j^{\mathrm{T}}\frac{\partial \mathbf{K}}{\partial a^{\mathrm{e}}}\boldsymbol{\phi}_j, \tag{22}$$

$$\frac{\partial \boldsymbol{\phi}_j}{\partial a^{\mathrm{e}}} = \sum_{q=1;\, q\neq j}^{d} \frac{\boldsymbol{\phi}_q}{\omega_j^2 - \omega_q^2}\left(\boldsymbol{\phi}_q^{\mathrm{T}}\frac{\partial \mathbf{K}}{\partial a^{\mathrm{e}}}\boldsymbol{\phi}_j\right). \tag{23}$$

K represents the stiffness matrix of the FE model. Instead of the complete base (d is the analytical model order) a truncated base is used.

5.2. Example: damaged RC beam

The FE model updating method can be used for damage assessment (damage localisation and quantification) of civil structures. In this paper the damage pattern of a reinforced concrete beam, which was damaged artificially in a laboratory test program, will be identified by updating the FE model of the beam.

5.3. Laboratory test program

The beam has a length of 6 m. Its section is plotted in Fig. 6. In the test program damage is induced by subjecting the beam to a static point load of 25 kN to produce cracks. The load is applied at 4 m of the left end of the beam (Fig. 7). Before and after applying this load, an experimental modal analysis is carried out to obtain the modal parameters of the reference and damaged state respectively. The modal test is performed on the beam with free–free boundary conditions, which are established by using very flexible springs supporting the beam (Fig. 8). Accelerometers are placed each 20 cm at both longitudinal edges of the upper side of the beam (62 measurement points in total, which are averaged to 31 values). The stochastic subspace identification technique [22] is applied to the dynamic response signals to extract the modal parameters. The first four bending modes are identified. The corresponding eigenfrequencies are given in Tables 1 and 2 for the reference and the damaged state respectively.

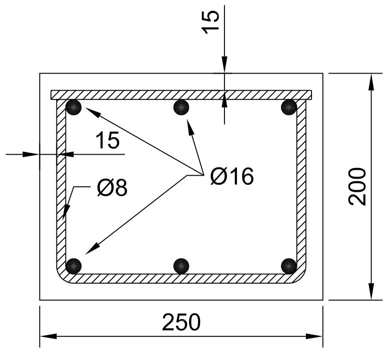

Fig. 6. Cross section of the beam.

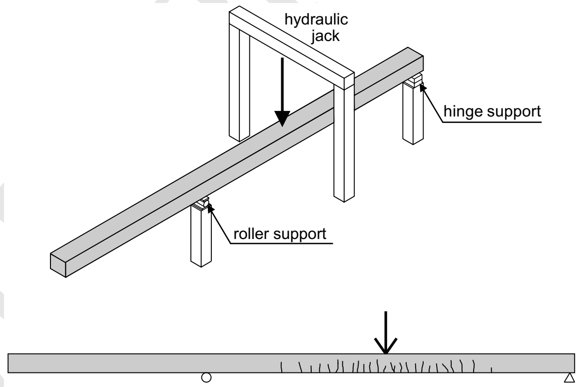

Fig. 7. A static point load is applied, at 4 m of the left beam end, in order to produce cracks.

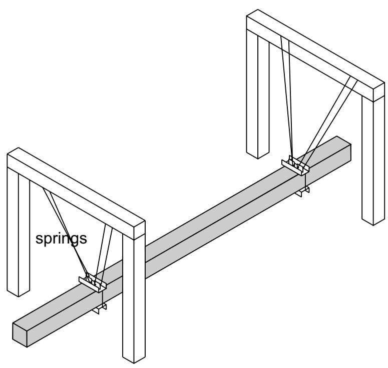

Fig. 8. A modal test is performed on the (reference and damaged) beam with free–free boundary conditions, established by using very flexible springs.

5.4. FE model updating

The beam is modelled with 30 beam elements in ANSYS [23] (Fig. 9a). The initial model for the undamaged state is characterised with a Young's modulus

Table 1
Reference state: eigenfrequencies and correlation values

Reference	Experiment	Initial FE model		Updated FE model	
Mode n^0	$\tilde{\nu}$ [Hz]	$\frac{\nu-\tilde{\nu}}{\tilde{\nu}}$ [%]	MAC [%]	$\frac{\nu-\tilde{\nu}}{\tilde{\nu}}$ [%]	MAC [%]
1	22.02	9.12	99.82	1.66	99.85
2	63.44	3.54	99.91	1.19	99.91
3	123.27	3.21	99.81	−0.05	99.90
4	201.92	2.55	99.81	−0.64	99.91

Table 2
Damaged state: eigenfrequencies and correlation values

Damaged	Experiment	Reference FE model		Updated FE model	
Mode n^0	$\tilde{\nu}$ [Hz]	$\frac{\nu-\tilde{\nu}}{\tilde{\nu}}$ [%]	MAC [%]	$\frac{\nu-\tilde{\nu}}{\tilde{\nu}}$ [%]	MAC [%]
1	19.35	15.69	99.35	1.96	99.81
2	56.90	12.82	99.11	1.52	99.91
3	111.64	10.36	98.15	−0.33	99.90
4	185.22	8.32	97.29	−1.27	99.94

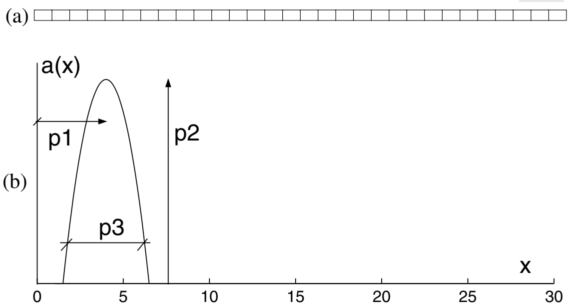

Fig. 9. FE model of the beam (a) and parabolic damage function (b).

of $E_0 = 37.5$ GPa and a moment of inertia of $I = 1.93 \times 10^{-4}$ m^4.

The structural damage is represented by a reduction factor on Young's modulus E^e of each beam element $\left(a^e = -\frac{E^e - E_0}{E_0}\right)$. Instead of modifying all 30 elements separately, a parabolic damage function is used to determine the damage pattern (Fig. 9b and Eq. (24)). The parabola is characterised by three parameters $\{p_1, p_2, p_3\}$ determining the position (element n^0: x), the height (relative stiffness reduction: a [–]) and the width (number of elements) of the damage pattern respectively. They all vary continuously in the process. Each set $\{p_1, p_2, p_3\}$ determines the corresponding vector of correction parameters $\boldsymbol{a}$ in a unique sense, by discretising the con-

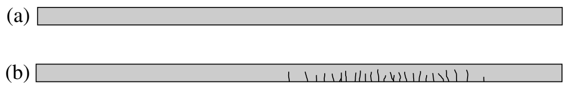

Fig. 10. Reference (a) and damaged state (b) of the beam.

tinuous distribution $a(x)$ in the beam elements of the FE model.

$$a(x, \boldsymbol{p}) = \max \begin{cases} -4\frac{p_2}{p_3^2}x^2 + 8\frac{p_1 p_2}{p_3^2}x + p_2 - 4\frac{p_1^2 p_2}{p_3^2}, \\ 0. \end{cases} \tag{24}$$

The Jacobian matrix $\mathbf{J}_a$, containing the sensitivities to $\boldsymbol{a}$, has to be adjusted as follows:

$$[\mathbf{J}_p]_{m\times 3} = [\mathbf{J}_a]_{m\times n} \left[\frac{\partial \boldsymbol{a}}{\partial p_1} \bigg| \frac{\partial \boldsymbol{a}}{\partial p_2} \bigg| \frac{\partial \boldsymbol{a}}{\partial p_3} \right]_{n\times 3}, \tag{25}$$

to obtain the Jacobian matrix $\mathbf{J}_p$ with sensitivities to $\boldsymbol{p}$, which are the variables of the optimization problem.

The damage detection is performed in two updating processes, to identify the reference and the damaged state respectively (Fig. 10).

In order to make the damage identification method successful, it is necessary to build an adequate FE model that predicts well the structural behaviour. Only some uncertainties remain (such as the stiffness of supports, of material or joints) that have to be determined in a first updating process, i.e. one that defines a representative reference FE model. In this process the analyzer chooses appropriate initial values of the uncertain parameters based on its engineering judgement.

The actual damage, however, is unknown to the analyzer and is identified in the second updating process. Since no prior knowledge exists, the initial damage parameters are chosen randomly, however still within physically meaningful limits.

In the reference state of the test beam some initial cracks were already present, [5] probably due to the self weight or the drying process of the fresh concrete.

5.4.1. Reference state

An objective function is set up consisting of four frequency residuals r_f and 104 mode shape residuals r_s corresponding with the major displacements of each of the four modes (Eqs. (15) and (16)). The experimental modal parameters are obtained from the modal test on the undamaged beam. In each iteration step, the MAC-values are calculated $\left(\mathrm{MAC} = \frac{|\boldsymbol{\phi}_j^{\mathrm{T}}\tilde{\boldsymbol{\phi}}_j|^2}{(\boldsymbol{\phi}_j^{\mathrm{T}}\boldsymbol{\phi}_j)(\tilde{\boldsymbol{\phi}}_j^{\mathrm{T}}\tilde{\boldsymbol{\phi}}_j)}\right)$ and used to correlate appropriately the experimental with the numerical modes. The vector of variables contains the three parameters $\{p_1, p_2, p_3\}$ of the parabolic damage function. The correction factors a^e_{ref} for all 30 beam el-

[5] The initial damage is not shown on Fig. 10.

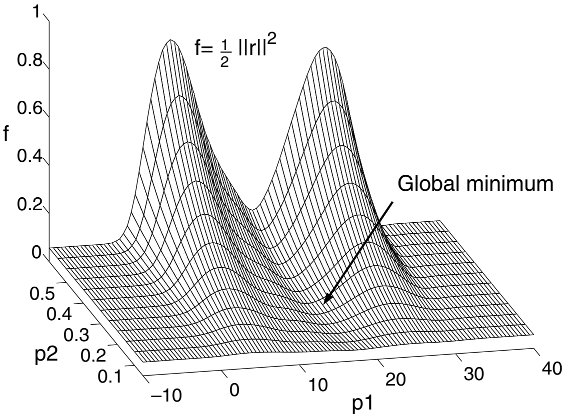

Fig. 11. Surface plot of the objective function with $p_3 = 10$ (reference state).

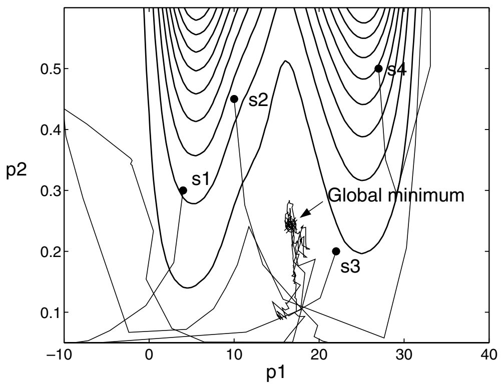

Fig. 12. Search path of a CLM run with four searching points, drawn on the contour plot of the objective function.

ements can be derived using Eq. (24). Note that a positive correction factor a^{e}_{ref} means a stiffness reduction:

$$E^{e} = E_0(1 - a^{e}_{ref}). \tag{26}$$

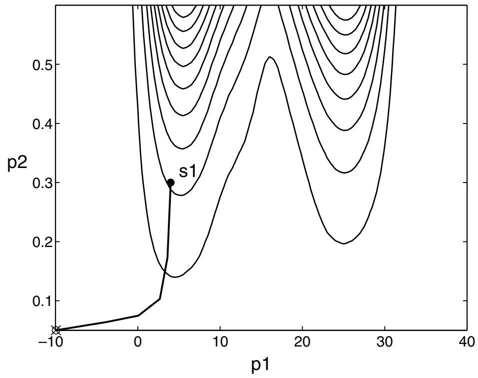

Fig. 13. Search path of a local optimization run (contour plot).

In order to visualize the objective function, the third parameter p_3 is, in a first approach, kept fixed to 10, retaining only two variables p_1 and p_2. This means that the width of the damage pattern is set to 10 elements beforehand and that only the position and the height of the damage have to be determined. The applied bounds are: $-10 \leqslant p_1 \leqslant 40$; $0.05 \leqslant p_2 \leqslant 0.6$.

The function is plotted with respect to p_1 and p_2 in Fig. 11. The surface is characterised by multiple valleys. Therefore, a global minimization method is required to find the global minimum, which is situated at $p_1 = 16.7$; $p_2 = 0.24$ for $p_3 = 10$.

A CLM optimization run is carried out with an initial population consisting of four searching points $\{s_1, s_2, s_3, s_4\}$ (Fig. 12), chosen well-spread in the design space by the analyzer. The normalization factors (Eqs. (11) and (12)) are $sc_f = 0.3$; $sc_{c1} = 30$; $sc_{c2} = 1$. Also the updating variables p_i are scaled to obtain a well-scaled function f. The tuning parameters are set to: $\eta = 3$ and $\gamma = 0.4$. The initial $\boldsymbol{\lambda}^{(i)}$ values are randomly distributed in the interval [−1;1]. The search process ends up in the global minimum (16.7; 0.24) (Fig. 12) after about 90 iterations.

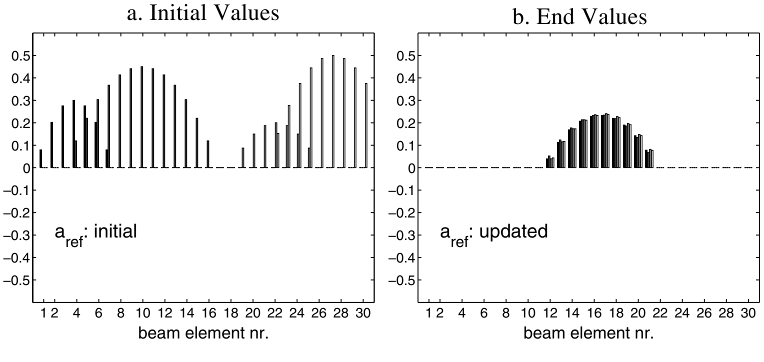

Fig. 14. Initial and updated correction factors a_{ref}, corresponding to the four searching points of the CLM run.

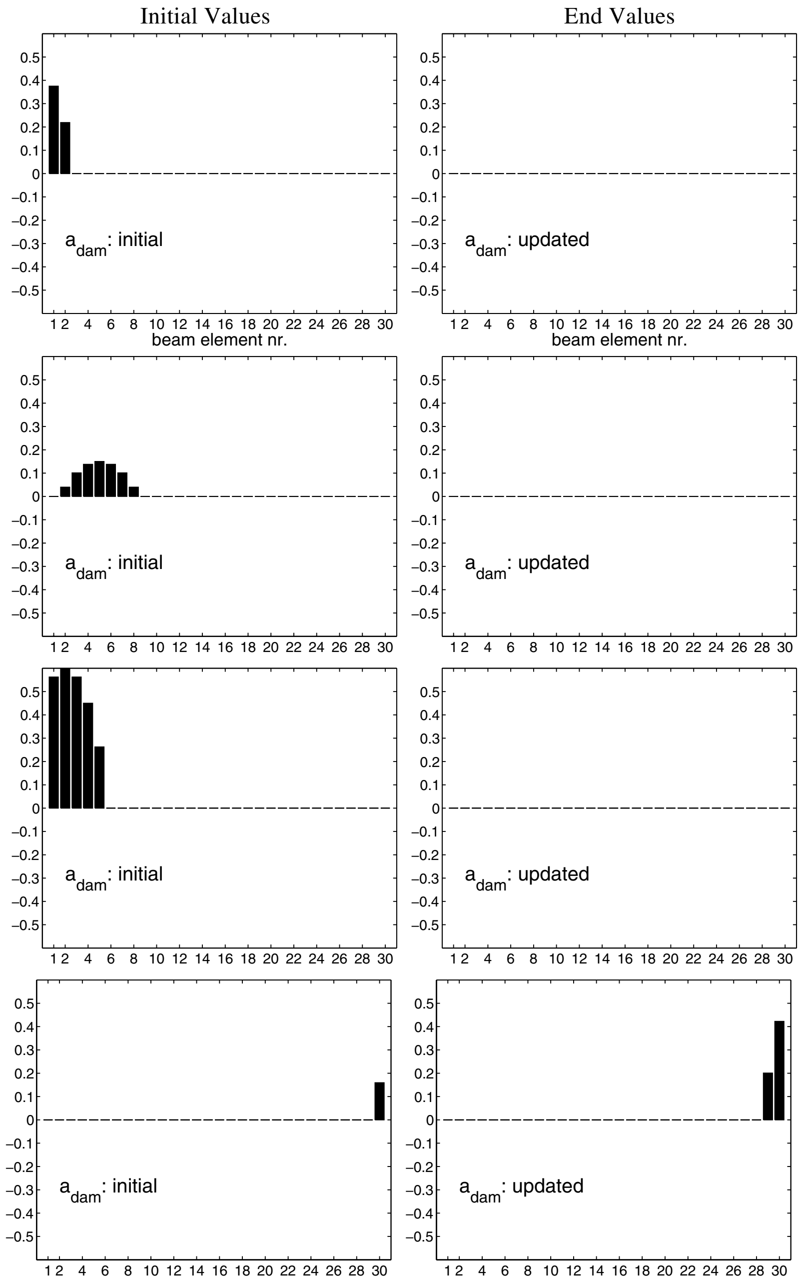

Fig. 15. Initial and updated correction factors a_{dam}, corresponding to four independent local optimization runs.

As illustration also the search path of a standard local minimization run, i.e. starting from only one point in the search space, is carried out. Fig. 13 shows that this process gets trapped in the nearest valley.

The improvement obtained with the CLM method in comparison to the standard local optimization methods is clear. Since a whole population of points explores the search space, the global minimum is detected with the CLM method, which is not always the case with a local method.

Additionally, the same objective function is also solved by varying all the three parameters $\{p_1, p_2, p_3\}$ of the parabolic damage function. In this case the position, the height and the width of the damage pattern have to be determined. Four local runs are carried out, resulting in different solutions, which indicates the existence of

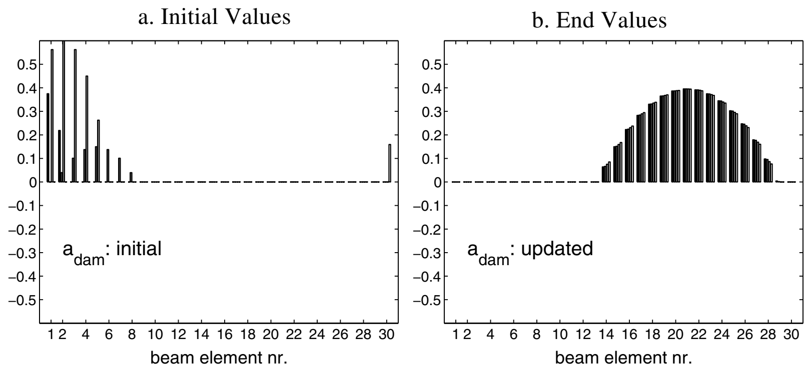

Fig. 16. Initial and updated correction factors a_{dam}, corresponding to one CLM run in which a population of four searching points is used.

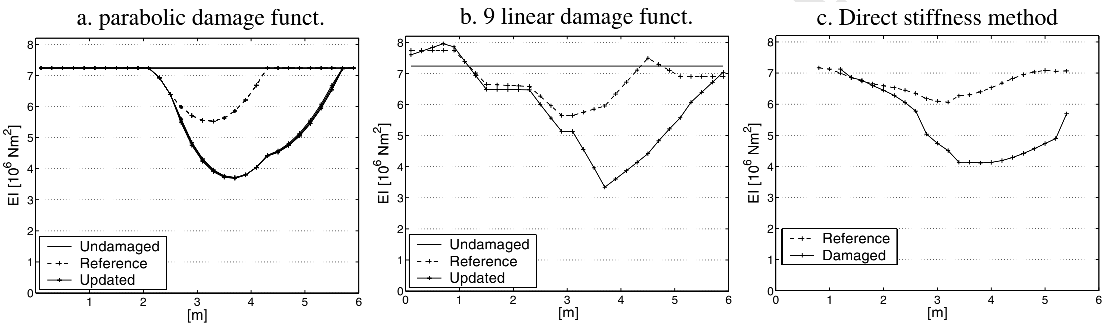

Fig. 17. Comparison of the stiffness distribution *EI* of the reference and damaged state, obtained with different techniques.

multiple local minima. Therefore, again a CLM optimization run is performed with a population consisting of four searching points. The corresponding initial parabolic damage patterns are plotted in Fig. 14a. The same normalization and tuning parameters as previously are used (plus $sc_{c3} = 10$). The third parameter is bounded by $7 \leqslant p_3 \leqslant 16$. The CLM run ends in the global optimum as can be seen in Fig. 14b, showing the damage pattern reached at the end of the optimization process. The obtained values for $\{p_1, p_2, p_3\}$ are:

$$p^*_{1\,\mathrm{ref}} = 16.7, \quad p^*_{2\,\mathrm{ref}} = 0.24, \quad p^*_{3\,\mathrm{ref}} = 10.4 \tag{27}$$

and are found in about 90 iterations. The reference state is characterised by a symmetrical damage pattern with a maximum reduction of 24%. Table 1 lists the relative differences in eigenfrequencies and the MAC-values, both for the undamaged and updated FE model. A clear improvement can be observed, particularly for the frequency differences.

5.4.2. Damaged state

In order to identify the applied damage, a second updating step is carried out in which the correction parameters $\boldsymbol{a}_{\mathrm{dam}}$ are determined with respect to the updated Young's modulus of the previous step:

$$E^{\mathrm{e}} = E^{\mathrm{e}}_{\mathrm{ref}}(1 - a^{\mathrm{e}}_{\mathrm{dam}}) = E_0(1 - a^{\mathrm{e}}_{\mathrm{ref}})(1 - a^{\mathrm{e}}_{\mathrm{dam}}), \tag{28}$$

where $a^{\mathrm{e}}_{\mathrm{ref}}$ is obtained by substituting $p^*_{i\,\mathrm{ref}}$ (Eq. (27)) in Eq. (24). An analogous optimization problem as in the first updating step is solved. The experimental modal parameters are now extracted from the measurements on the damaged beam. The same frequency and mode shape residuals are selected to construct the objective function. In this updating step, $\{p_1, p_2, p_3\}$ determine the correction factors for the damaged state, $a^{\mathrm{e}}_{\mathrm{dam}}$. All the three of them are varied [6] and bounded by $-10 \leqslant p_1 \leqslant 40$; $0.15 \leqslant p_2 \leqslant 0.6$; $7 \leqslant p_3 \leqslant 20$. They are also scaled to form a well-scaled function f.

A CLM optimization run is performed, again with a population consisting of four local minimizers. In order to show the robustness of the method, their initial values are chosen such that four independent local runs,

[6] The optimization with only two parameters is not reported here.

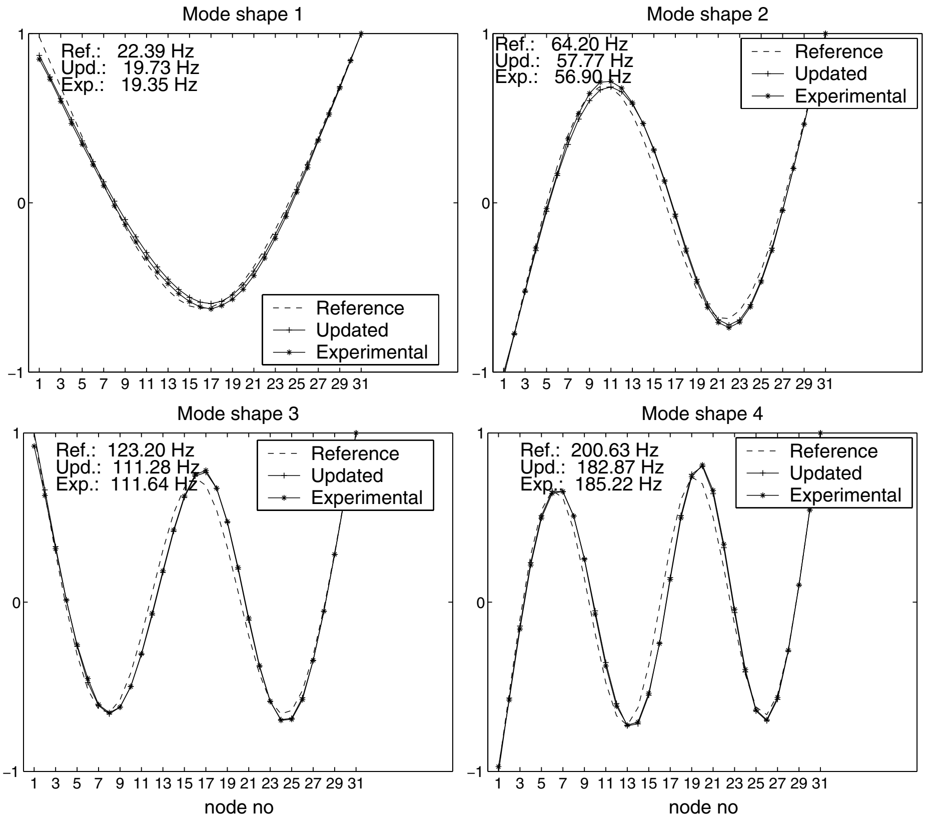

Fig. 18. Experimental and numerical (reference and updated) bending mode shapes (damaged state).

starting from the same four points separately and using a standard local optimization method, all end up in a wrong solution [7] (Fig. 15). Notwithstanding that, the CLM method does find the global minimum (Fig. 16), situated at

$$p^*_{1\,\mathrm{dam}} = 21.1, \quad p^*_{2\,\mathrm{dam}} = 0.4, \quad p^*_{3\,\mathrm{dam}} = 15.7. \tag{29}$$

The tuning parameters used for the optimization are $\eta = 3$ and $\gamma = 0.3$ and the normalization factors are $sc_\mathrm{f} = 1$ and $(sc_{\mathrm{c}1}, sc_{\mathrm{c}2}, sc_{\mathrm{c}3}) = (30, 1, 10)$. The global minimum is identified in about 110 iterations.

The applied damage is identified correctly (Fig. 16b). It is an asymmetrical damage pattern with a maximum value of $a^\mathrm{e}_\mathrm{dam} = 40\%$ of the reference Young's modulus, at the location where the static load was applied, i.e. at 4 m of the left beam end (Fig. 7). The influence of the cracks is spread out over a zone consisting of 16 beam elements.

In Fig. 17a the updated stiffness distribution EI is plotted for the reference and the damaged state, the latter obtained by applying the identified damage of the second updating step to the reference stiffness distribution of the first step (Eq. (28)). The resulting stiffness distribution shows an asymmetrical pattern with a maximum dip of 49% of the initial bending stiffness E_0I at 3.7 m.

It can be compared with the stiffness distribution obtained through FE model updating using nine piecewise linear p-independent damage functions and a standard optimization method [24] (Fig. 17b). The stiffness distribution is also calculated with the direct stiffness calculation (DSC) method [25]. This damage assessment technique calculates the stiffness directly, without using or updating any FE model, and is based on the modal frequencies and curvatures. The method is applied on the same beam using the first three modes and the results are shown in Fig. 17c. [8] A similar pattern is identified for the undamaged as well as for the dam-

[7] An undamaged state is obtained in the first three runs ($p_1 \leqslant -4$) and an almost undamaged state in the last run ($p_1 = 32$).

[8] The calculated values for the bending stiffness at the ends of the beam are omitted.

aged beam for the three figures. Particularly, the amount of induced damage corresponds well.

Table 2 lists the relative differences in eigenfrequencies and the MAC-values with respect to the experimental data of the damaged beam. Again a satisfactory result is obtained.

Additionally, the experimental and numerical (reference and updated) mode shapes of the four bending modes are plotted in Fig. 18. They are all scaled to 1 in the reference node, located at right beam end (at 6 m). The experimental and the updated mode shapes correspond well. Only some minor discrepancies remain, probably due to the initial cracked state, which is not perfectly symmetrical with respect to the longitudinal beam axis and therefore cannot be modelled accurately with the beam model. Also the identified damage pattern is restricted to a parabola, which might differ from the real damage distribution.

6. Conclusions

A new global optimization method is investigated, named coupled local minimizers. In CLM the average objective function value of multiple design vectors is minimized, subjected to pairwise synchronization constraints. This is done with the augmented Lagrangian method, which we have implemented with a Newton-based algorithm, in order to maximize the convergence rate. Furthermore, the Trust Region approach makes it possible to minimize a nonconvex function. In order to generalize the problem, the objective function and the synchronization constraints are normalized.

The CLM method is successfully applied to a test function containing several local minima. We have demonstrated the robustness of CLM, in the sense that the method finds the global minimum of the test function, even if all the search points are initially situated in the valley of a local minimum. The influence of the tuning parameters on the search process is shown. In a second illustration, CLM is used for FE model updating. The correct damage pattern of a beam is identified with the method. In both examples the advantages of CLM over conventional multistart local optimization algorithms are clearly shown.

Acknowledgements

This research work was partially carried out in the framework of the Belgian Programme on Interuniversity Poles of Attraction, initiated by the Belgian State, Prime Minister's Office for Science, Technology and Culture (IUAP P4-02 & IUAP P4-24), the Concerted Action Project MEFISTO of the Flemish Community and the FWO project G.0080.01 *Collective Behaviour and Optimization: an Interdisciplinary Approach*.

The beam tests were carried out within the FKFO-project no. G.0243.96, supported by the FWO—Flanders.

Anne Teughels is a research assistant and Johan Suykens is a postdoctoral researcher, both with the National Fund for Scientific Research FWO—Flanders.

References

[1] Friswell MI, Mottershead JE. Finite element model updating in structural dynamics. Dordrecht, The Netherlands: Kluwer Academic Publishers; 1995.

[2] Maia NMM, Silva JMM, He J. Theoretical and experimental modal analysis. Somerset, England: Research Studies Press; 1997.

[3] Rao SS. Engineering optimization-theory and practice. 3rd ed. New York: John Wiley & Sons; 1996.

[4] Gill PE, Murray W, Wright MH. Practical optimization. 11th ed. San Diego: Academic Press Limited; 1997.

[5] Nocedal J, Wright SJ. Numerical optimization. New York, USA: Springer; 1999.

[6] Holland J. Adaptation in natural and artificial systems. Ann Arbor, MI: University of Michigan Press; 1975.

[7] Kirkpatrick S, Gelatt CD, Vecchi MP. Optimization by simulated annealing. Science 1983;220:671–80.

[8] Levin RI, Lieven NAJ. Dynamic finite element model updating using simulated annealing and genetic algorithms. Mech Syst Signal Process 1998;12(1):91–120.

[9] Suykens JAK, Vandewalle J, De Moor B. Intelligence and cooperative search by coupled local minimizers. Int J Bifurc Chaos 2001;11(8):2133–44.

[10] Suykens JAK, Vandewalle J. Coupled local minimizers: alternative formulations and extensions. In: 2002 World Congress on Computational Intelligence—International Joint Conference on Neural Networks IJCNN 2002, Honolulu, USA, 2002, p. 2039–43.

[11] Dutta VP, Mukherjee S, Kundra TK, Genetic algorithms for optimal structural dynamic modification. In: Proceedings of Imac XIX: A conference on structural dynamics, Kissimmee, Florida, 2001, p. 1682–7.

[12] Gunduz N, Akbulut N, Sonmez FO. Generating optimal 2D structural designs using simulated annealing. In: Proceedings of OPTI 2001: 7th International Conference on Computer Aided Optimum Design of Structures. Bologna, Italy: WIT Press; 2001. p. 347–56.

[13] Hasançebi O, Erbatur F. Layout optimization of trusses using simulated annealing. In: Proceedings of 2nd International Conference on Engineering Computational Technology and 5th International Conference on Computational Structures Technology, vol I. Leuven, Belgium: Civil-Comp Press; 2000. p. 175–90.

[14] Shrestha SM, Ghaboussi J. Evolution of optimum structural shapes using genetic algorithm. J Struct Eng 1998;124(11):1331–8.

[15] Erbatur F, Hasançebi O, Tütüncü I, Kılıç H. Optimal design of planar and space structures with genetic algorithms. Comput Struct 2000;75:209–24.

[16] Nanakorn P, Meesomklin K. An adaptive penalty function in genetic algorithms for structural design optimization. Comput Struct 2001;79(29–30):2527–39.

[17] Lagaros ND, Papadrakakis M, Kokossalakis G. Structural optimization using evolutionary algorithms. Comput Struct 2002;80(7–8):571–89.

[18] Chen T-Y, Su J-J. Improvements of simulated annealing in optimal structural designs. In: Proceedings of 2nd International Conference on Engineering Computational Technology and 5th International Conference on Computational Structures Technology, vol I. Leuven, Belgium: Civil-Comp Press; 2000. p. 169–74.

[19] MATLAB, Matlab optimization toolbox user's guide. Available at: <http://www.mathworks.com/products/optimization> Version 2.1 (Release 12.1). The Mathworks 2000.

[20] Teughels A, De Roeck G. A method for updating finite element models of civil engineering structures, applied on a railway bridge. In: Proceedings of COST F3 International Conference on Structural System Identification, Kassel, Germany, 2001, p. 507–16.

[21] Fox R, Kapoor M. Rate of change of eigenvalues and eigenvectors. AIAA J 1968;6:2426–9.

[22] Peeters B, De Roeck G. Reference-based stochastic subspace identification for output-only modal analysis. Mech Syst Signal Process 1999;6(3):855–78.

[23] ANSYS, Robust simulation and analysis software. Available at: <http://www.ansys.com> Release 5.7.1. ANSYS Incorporated, 2001.

[24] Teughels A, Maeck J, De Roeck G. FEM updating of a reinforced concrete beam using damage functions. In: Proceedings of International Conference on Structural Dynamics Modelling: Test, Analysis, Correlation and Validation, Madeira Island, Portugal, 2002, p. 583–92.

[25] Maeck J, De Roeck G. Damage detection on a prestressed concrete bridge and RC beams using dynamic system identification. In: Proceedings DAMAS 99. Dublin, Ireland: Trans Tech Publications; 1999. p. 320–7.