ELSEVIER

Engineering Analysis with Boundary Elements 26 (2002) 315–327

ENGINEERING ANALYSIS *with* BOUNDARY ELEMENTS

www.elsevier.com/locate/enganabound

Optimization algorithms for identification inverse problems with the boundary element method

Guillermo Rus, Rafael Gallego*

Department of Structural Mechanics, University of Granada, Edificio Politecnico de Fuentenueva, E-18071 Granada, Spain

Received 21 September 2001; revised 2 January 2002; accepted 4 January 2002

Abstract

In this paper the most suitable algorithms for unconstrained optimization now available applied to an identification inverse problem in elasticity using the boundary element method (BEM) are compared. Advantage is taken of the analytical derivative of the whole integral equation of the BEM with respect to the variation of the geometry, direct differentiation, which can be used to obtain the gradient of the cost function to be optimized. © 2002 Elsevier Science Ltd. All rights reserved.

Keywords: Inverse problems; Optimization algorithms; Identification of defects; Analytical derivative; Sensitivity equations

1. Introduction

Identification inverse problems (IIPs) arise in the search of defects in materials and structures (subsurface flaws, inclusions or cracks), or geological prospections (water, oil, configuration of layers), among other applications. They can be tackled with elastic excitation (static displacements, or dynamic displacements in the form of a study of the propagation of sound or seismic waves), or with electric, thermal or other flow measurements, described by similar potential equations.

The identification can be based either on propagation phenomena (such as elastic deformation, wave propagation, acoustics, etc.), governed by partial differential equations, or based on radiation (X-rays tomography, reconstruction by photography, etc.). In another classification, it is possible to measure static response, steady state, transient response and eigenmodes and eigenfrequencies. One may also distinguish between acoustic response and acoustic emission depending on the source of the excitation. The problems tackled here belong to the class of static elastic field.

Nowadays, most methods used for the solution of the IIPs are qualitative, requiring either approximations to get analytic solutions or interpretation of the measured data by an experienced person.

1.1. Inverse problems

A generic inverse problem can be defined as a counterpart to the definition of a direct one. If the direct problem is stated as the calculation of the response (certain field v, like the displacements, and stress vectors q) in a specific body defined by its geometry Ω, mechanical properties (k), behaviour model (operator L) and boundary conditions (some known values of u and q)

$$L(k)v = q \text{ on } \Omega$$

the nature of the unknown yields the following classification of inverse problems by Kubo [13]:

- IIP: a part of the geometry (Ω). This is the problem we are dealing with in this paper.
- modelization: the mathematical equations that govern the behaviour (L).
- reconstruction: the boundary or initial conditions.
- external actions: q.
- material properties: some parameters characterizing the material (k).

In order to find this data, supplementary information has to be provided, in the form of some extra measurements of v or q made on an accessible area of the specimen.

A sensible way to tackle this problem is to define the geometry, the boundary of the domain, by a discrete or continuous design variable(s) z. So, the problem can now be stated as the search of z so that the implicit equation

* Corresponding author. Fax: +34-958-24-9512.
E-mail address: gallego@ugr.es (R. Gallego).

PII: S0955-7997(02)00008-5

$L(z,k)v = q$ holds with v ideally equal to the experimental values v^{ex}.

The selection of the design variables z and the simplifications that can be made are a subject closely related to the so-called regularization methods. They are mainly aimed at increasing the conditioning of the problem and making the ensuing algorithms more convergent, when not convergent at all. The principle of the regularization is the addition of information about the variable z in some form (reduction of unknowns by a parameterization, addition of conditions in the relationship between parameters or in the geometry, etc.). The study of the regularization methods is a very broad and ongoing research area.

The explicit relationship between the primary variable v and the design variable z will generally not be available as $F(z) = v$. However, for a given value z, the variable v will be computed by a suitable (numerical) method, here the boundary element method (BEM). Hence for the real values of z, say $\bar{z}$, $F(z)$ will ideally be equal to v^{ex} at the measurement points.

At this point we can see the main difficulties of inverse problem:

- $F(z)$ may be a set of highly non-linear functions, making the inverse problem ill posed, and the algorithms to solve it, unstable and ill conditioned.
- The implication only holds in one direction, so the solution of z may not be unique. The existence of multiple solutions is a consequence of the non-linearity of F, but can be increased by an incorrect choice of the design variable or its parameterization.
- In real cases when the model is not accurately defined and since the measurements involve unavoidable errors, the equality $F(\bar{z}) = v^{ex}$ will be unattainable, i.e. the real value of v is not v^{ex} due to experimental errors, and the real relationship between v and z is not F due to inaccuracies/errors in the model.

The tools to reduce these effects are known as regularization techniques.

1.2. Solution strategies

Two set-ups of the IIP can be considered in order to find a solution:

- solving directly the observation equation

$$F(z) = v^{ex}$$

- minimize a suitably defined cost function $f(z)$, for instance,

$$\min_{z\in\mathcal{R}^n} f(z) = \min_{z\in\mathcal{R}^n} \frac{1}{2}\|F(z) - v^{ex}\|^2.$$

The exact solution of the observation equation may be impossible since the equations may have no or non-unique solution. On the other hand, some methods to find optimal solutions, in some projective sense, are shown to be closely related to the Gauss–Newton method.

In the second and more general approach, the definition adopted for the cost functional as the L^2 norm of the residual simply means that the deviation of the predicted response from the measured one will be minimal (this has some statistical interpretations as well, see Refs. [1,15,20]).

There are several methods that can be applied to solve inverse problems. A partial classification of them depending on the scope of the convergence is shown in Fig. 1.

The ideal algorithm should cover all the scope in order to start from a completely unknown configuration and end up with any required approximation (1: accuracy). The way to achieve this at an affordable computational cost (2: efficiency) and a good likelihood to attain a solution (3: convergence) is through several stages ranging from global to local methods successively.

The methods tested in this paper are the optimization algorithms, and in particular secant and least squares methods since they are the most used in the literature.

2. The direct problem

The BEM provides clear advantages in comparison with the finite element and other methods to tackle this kind of

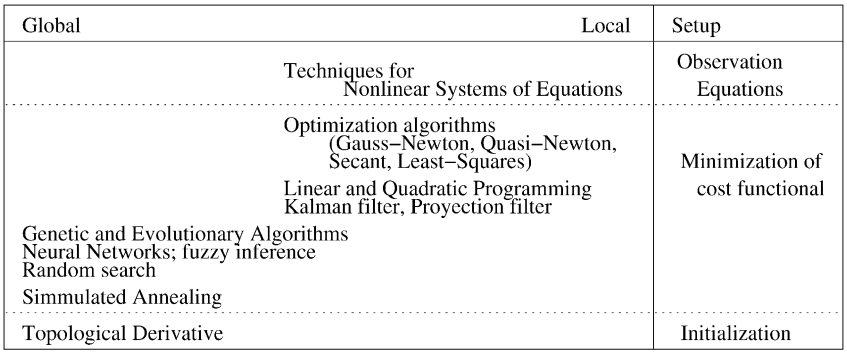

Fig. 1. A classification of inverse problem strategies.

problem:

- It does not require a remeshing of the domain at each iteration. This reduces both the computational effort and eliminates the important perturbations due to changes in the mesh.
- The application of these methods to real problems may require many iterations as well as high precision in the intermediate solutions, so the use of finite elements would be more expensive.

2.1. Governing equations

The elastic behaviour of a body under static loads is governed by the equilibrium, compatibility and constitutive equations. These equations can be summarized in a boundary integral equation

$$c_k^i(\mathbf{y})u_k(\mathbf{y}) + \int_\Gamma \Big[q_k^i(\mathbf{x};\mathbf{y})u_k(\mathbf{x}) - u_k^i(\mathbf{x};\mathbf{y})q_k(\mathbf{x})\Big]\mathrm{d}\Gamma(\mathbf{x}) = 0 \tag{1}$$

where Γ is the boundary of the body, q_i the traction vector at a boundary point whose outward normal is n_i, and u_i is the displacement vector, $c_k^i(y)$ is the free term, which depends on the position of the collocation point $\mathbf{y}$ ($c_k^i = 0$ if $\mathbf{y} \notin \Omega \bigcup \Gamma$; $c_k^i = 1$ if $\mathbf{y} \in \Omega$ and its value depends on the local geometry of the boundary if $\mathbf{y} \in \Gamma$); u_k^i and q_k^i are the displacement and traction vectors of the fundamental solution, respectively (see Ref. [6] for further details). The boundary conditions,

$$u_i(\mathbf{x}) = \bar{u}_i \ \text{ on } \Gamma_u, \qquad q_i(\mathbf{x}) = \bar{q}_i \ \text{ on } \Gamma_q$$

complete the set of equations to be solved, where Γ_u and Γ_q are non-overlapping partitions of the boundary Γ.

3. Sensitivity computation

Due to changes in the geometry, the variation of the cost functional (δf) is necessary in order to improve and accelerate the convergence of the local minimization algorithms. To do so, the sensitivity of the primary variables, δu_k and δq_k, to the variation in the geometry, has to be computed. A classification of the different approaches to do this calculation, with respect to the computational cost, is the following:

- Finite differences (FD): this approach has a very high computational cost since one additional direct problem solution (or two for central schemes) at a finite distance from the original one is required for each design variable. This method has been widely used, however, due to its simplicity [4,17].
- The adjoint variable method (AV): it is based on the computation of an auxiliary problem, the adjoint one, such that the sensitivities of the primary variables is not needed in order to compute δf. It implies the computation of an adjoint problem for each cost function, plus some fast calculations to obtain each derivative with respect to each design variable. This approach was used in Refs. [2,5,16,7].
- Direct differentiation (DD): the sensitivity of the primary variables is first computed and then the sensitivity of any cost functional. A direct problem for each design variable has to be solved. The basis for the direct derivative came from the variation formulation used directly for inverse problem in Refs. [21,22], and for the first time with success in Refs. [18,19]. Mellings and Aliabadi [14] successfully used DD of the discrete equations for crack identification.

In the present paper the sensitivity boundary integral equations are obtained not by the formulas of material differentiation of integrals, as in Ref. [5], but by an alternative procedure based on series expansion. Although it has not been rigorously established, both procedures should lead to the same or equivalent integral equations. The effect of the exact (DD) versus the approximate (FD) computation of the gradient is considered and thoroughly tested for the chosen algorithms.

3.1. Sensitivity computation by direct differentiation

In order to compute the sensitivity of the primary variables with respect to the shape of a domain, the variation of the boundary integral equation, δBIE, was properly derived by Suarez and Gallego [19] for potential problems and Rus and Gallego [18] for elasticity.

The shape sensitivity of a magnitude $v(\mathbf{x})$ can be understood as its derivative with respect to a change in the geometry. In this sense, that sensitivity can be defined as:

$$\delta v(\mathbf{x}) = \lim_{h\to 0} \frac{v(\mathbf{x} + h\,\delta\mathbf{x}) - v(\mathbf{x})}{h}.$$

For this reason, the responses of two problems, an original one $v(\mathbf{x})$ and one corresponding to a geometry perturbed by an infinitesimal magnitude $\delta\mathbf{x}$, namely $v(\mathbf{x} + \delta\mathbf{x})$, are compared. The second one is obtained from the first, with the help of a series expansion of each term arising in the boundary integral equation.

Consider a perturbed problem where the shape of the domain and therefore its boundary is modified by a infinitesimal variation of the design variable. In the sequel $\tilde{\Omega}$ and $\tilde{\Gamma}$ denote the modified domain and its boundary, whereas Ω and Γ refer to the same ones for the actual one. In general a tilde over a variable denotes its value for the modified configuration.

Eq. (1) can be written for an interior collocation point $\mathbf{y}$ in the assumed domain,

$$u_k(\mathbf{y},\Gamma) + \int_\Gamma \Big[q_k^i(\mathbf{x};\mathbf{y})u_k(\mathbf{x},\Gamma) - u_k^i(\mathbf{x};\mathbf{y})q_k(\mathbf{x},\Gamma)\Big]\mathrm{d}\Gamma(\mathbf{x}) = 0 \tag{2}$$

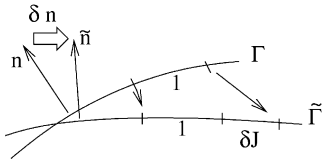

Fig. 2. Graphical explanation of δn_i and δJ.

and the same equation can be set at the corresponding point $\tilde{\mathbf{y}}$ in the modified domain,

$$u_i(\tilde{\mathbf{y}},\tilde{\Gamma}) + \int_{\tilde{\Gamma}} \left[q_k^i(\tilde{\mathbf{x}};\tilde{\mathbf{y}}) u_k(\tilde{\mathbf{x}},\tilde{\Gamma}) - u_k^i(\tilde{\mathbf{x}};\tilde{\mathbf{y}}) q_k(\tilde{\mathbf{x}},\tilde{\Gamma}) \right] \mathrm{d}\tilde{\Gamma}(\tilde{\mathbf{x}}) = 0. \tag{3}$$

The extra parameter, the boundary, in the primary variables has been added to stress their dependence on the shape of the domain, which is the unknown variable for the IIP.

3.2. Series expansion

The variation of the needed data and geometrical values are defined as follows, and explained in Fig. 2.

$$\delta x_i = \tilde{x}_i - x_i, \qquad \delta y_i = \tilde{y}_i - y_i, \qquad \delta n_i = \tilde{n}_i(\tilde{\mathbf{y}}) - n_i(\mathbf{y}),$$
$$\mathrm{d}\tilde{\Gamma}(\tilde{\mathbf{y}}) = (1 + \delta J)\mathrm{d}\Gamma(\mathbf{y}), \qquad \delta u_i = u_i(\tilde{\mathbf{y}},\tilde{\Gamma}) - u_i(\mathbf{y},\Gamma),$$
$$\delta q_i = q_i(\tilde{\mathbf{y}},\tilde{\Gamma}) - q_i(\mathbf{y},\Gamma). \tag{4}$$

Note that the variation of the displacements and tractions are material variations since the change from the assumed configuration to the actual one implies the change of both the boundary and the calculation point.

The variables and quantities in Eq. (3) can be written in terms of the same variables and quantities of the actual domain and their variations. Likewise, the kernels in this integral equation can be expanded in series using the variations defined in Eq. (4).

$$u_k^i(\tilde{\mathbf{x}};\tilde{\mathbf{y}}) = u_k^i(\mathbf{x};\mathbf{y}) + u_{k,m}^i(\mathbf{x};\mathbf{y})\delta r_m + \text{h.o.t.}$$
$$q_k^i(\tilde{\mathbf{x}},\tilde{\mathbf{y}}) = \sigma_{jk}^i(\tilde{\mathbf{x}};\tilde{\mathbf{y}}) n_j(\tilde{\mathbf{x}}) = q_k^i(\mathbf{x};\mathbf{y}) + \sigma_{jk}^i(\mathbf{x};\mathbf{y})\delta n_j(\mathbf{x}) + \sigma_{jk,m}^i(\mathbf{x};\mathbf{y})\delta r_m n_j(\mathbf{x}) + \text{h.o.t.} \tag{5}$$

where h.o.t. stands for higher order terms and $\delta r_m = \delta x_m - \delta y_m = \delta(x_m - y_m)$.

3.3. Variation boundary integral equation (δBIE)

Substituting Eqs. (4) and (5) in Eq. (3), neglecting higher order terms, and subtracting Eq. (2), an integral equation valid for an interior point $\mathbf{x}$ is obtained. A final limit to a smooth boundary point leads to the variation boundary integral equation (δBIE),

$$\frac{1}{2}\delta u_k(\mathbf{y}) + \int_\Gamma \left(q_k^i\, \delta u_k - u_k^i\, \delta q_k \right) \mathrm{d}\Gamma = \int_\Gamma \left\{ \left(u_{k,m}^i q_k - \sigma_{jk,m}^i n_j u_k \right)\delta r_m + \left(u_k^i q_k - q_k^i u_k \right)\delta J - \sigma_{jk}^i u_k\, \delta n_j \right\} \mathrm{d}\Gamma. \tag{6}$$

The discretization of the δBIE follows standard boundary element techniques. First, the boundary is divided into a number of elements (Γ_e), and at each one, the variables are interpolated in terms of their value at a series of points (interpolation nodes) using shape functions ψ_n and ϕ_n. The sensitivities are consistently discretized in the same way,

$$u_i(\mathbf{x}) = \sum_{n\in\Gamma_e} \phi_n u_i^{(n)} \qquad q_i(\mathbf{x}) = \sum_{n\in\Gamma_e} \psi_n q_i^{(n)}$$
$$\delta u_i(\mathbf{x}) = \sum_{n\in\Gamma_e} \phi_n\, \delta u_i^{(n)} \qquad \delta q_i(\mathbf{x}) = \sum_{n\in\Gamma_e} \psi_n\, \delta q_i^{(n)}.$$

However, the variation of the geometry of the boundary ($\delta\mathbf{y}$), will be interpolated using a special parameterization, which will be C^∞ regardless of the discretization of the boundary, something which will assure the convergence of the limits to the boundary. For each cavity, the following parameterization will be considered, $\delta\mathbf{y}(\mathbf{y}) = \mathbf{P}\,\delta z$ where,

$$\mathbf{P}(\mathbf{y}) = \begin{bmatrix} 1 & 0 & \left(y_2 - y_2^0\right) & \left(y_1 - y_1^0\right) & \left(y_1 - y_1^0\right) & \left(y_2 - y_2^0\right) \\ 0 & 1 & -\left(y_1 - y_1^0\right) & \left(y_2 - y_2^0\right) & -\left(y_2 - y_2^0\right) & \left(y_1 - y_1^0\right) \end{bmatrix}$$

being (y_1^0, y_2^0) the coordinates of the centroid of the flaw.

The discrete design or geometric variable vector, δz, is a six component vector defined as,

$$\delta z = \begin{bmatrix} \delta y_1^0 \\ \delta y_2^0 \\ \delta\omega \\ \delta\varepsilon_m \\ \delta\varepsilon' \\ \delta\varepsilon_{12} \end{bmatrix} = \begin{bmatrix} \text{Horizontal displacement of the centroid} \\ \text{Vertical displacement of the centroid} \\ \text{Rotation} \\ \text{Dilatation} \\ \text{Elongation} \\ \text{Distortion} \end{bmatrix}.$$

This parameterization represents the variation of the geometry as a movement of the cavity due to a virtual deformation field.

All the variables that depend on the variation of the

geometry can be written in terms of the geometric variables as well. Thus,

$$\delta r_i = (P_{ij}(\mathbf{x}) - P_{ij}(\mathbf{y}))\delta z_j, \quad \delta J = Q_j\,\delta z_j, \quad \delta m_i = R_{ij}\,\delta z_j$$

where t_i is the tangent vector to the boundary and

$$Q_j = \frac{\mathrm{d}y_i\,\mathrm{d}y_k}{\mathrm{d}y_m\,\mathrm{d}y_m}P_{ij,k}, \qquad R_{1j} = P_{2j,1}t_1 - P_{2j,2}t_2,$$

$$R_{2j} = -P_{1j,1}t_1 + P_{1j,2}t_2.$$

3.4. Direct design variable update

After following this discretization procedure the δBIE is cast in an algebraic system of equations,

$$\mathbf{H}\,\delta\boldsymbol{u} = \mathbf{G}\,\delta\mathbf{p} + \boldsymbol{\Delta}\,\delta\mathbf{z}$$

where $\delta\mathbf{u}$ and $\delta\mathbf{p}$ collect the variation of the displacements and tractions at all interpolation nodes on the boundary. Applying the boundary conditions as in the direct BEM, rearranging the system according to the boundary conditions, collecting the unknown sensitivities of displacements and tractions in the vector $\delta\boldsymbol{v}$, and solving the ensuing system of equations, the following the matricial relationship is obtained,

$$\delta\boldsymbol{v} = \mathbf{A}\,\delta\boldsymbol{z}. \tag{7}$$

At this point, the matrix $\mathbf{A}$ has several usages. In the framework of optimization algorithms, $\mathbf{A}$ is the Jacobian of $\boldsymbol{v}$ needed in most of the algorithms (named later J), whereas in the observation equation approach, the use is as follows.

If the actual estimate of the design variable, $\boldsymbol{z}_k$ is such that $\boldsymbol{F}(\boldsymbol{z}_k) = \boldsymbol{v}_k \neq \boldsymbol{v}^{\mathrm{ex}}$, one would like the new estimate $\boldsymbol{z}_{k+1}$ to be such that $\boldsymbol{F}(\boldsymbol{z}_{k+1}) = \boldsymbol{v}^{\mathrm{ex}}$, or $\boldsymbol{F}(\boldsymbol{z}_{k+1}) - \boldsymbol{F}(\boldsymbol{z}_k) = \boldsymbol{v}^{\mathrm{ex}} - \boldsymbol{v}_k = \delta\boldsymbol{v}_k$. The linear version of the previous equation is given by the system of equations in Eq. (7). This overdetermined system of equation is solved by least squares, leading to

$$\delta\boldsymbol{z} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\,\delta\boldsymbol{v}. \tag{8}$$

Then, the design variables are updated from its previous estimate by

$$\boldsymbol{z}_{k+1} = \boldsymbol{z}_k + \delta\boldsymbol{z}$$

and the whole procedure is repeated until a suitable convergence criterion is fulfilled. This is the simplest procedure to solve the observation equations. In Section 4 a brief review of the most suitable algorithms for solving the inverse problem cast as an optimization problem are presented.

4. Algorithms for unconstrained optimization

A good survey on the methods for local unconstrained optimization was carried out by Dennis and Schnabel [12], and others [11].

The unconstrained optimization problem can be simply stated as

$$\min_{x\in\mathcal{R}^n} f(x) : \mathcal{R}^n \rightarrow \mathcal{R}.$$

The methods tested in this paper are the most used in the literature to solve IIP: BFGS, Gauss–Newton, damped Gauss–Newton and Levenberg–Marquardt method. In the following paragraphs a brief summary of them is presented.

4.1. Newton's method

By a multivariate Taylor series expansion to the second term, a model of $f(x)$ can be defined as,

$$m_c(x_c + p) = f(x_c) + g(x_c)^T p + \frac{1}{2}p^T H(x_c)p$$

where $g_i(x_c) = \partial f/\partial x_i$ is the gradient, $H_{ij}(x_c) = \partial^2 f/\partial x_i\,\partial x_j$ is the Hessian.

Newton's method is an iterative procedure that follows the following steps,

1. k = 0, initialize x_k
2. Compute $g_k = g(x_k)$ and $H_k = H(x_k)$
3. Solve $H_k s_k = -g_k$
4. Update $x_{k+1} = x_k + s_k$
5. Repeat 2–5 until convergence.

The interesting property of this approach is that converges q-quadratically ($|x_{k+1} - x_k| \leq c_k|x_k - x_*|^2$).

If the gradient and the Hessian are not available, they can be calculated with the FD approximations,

$$g_i(x_c) \simeq \frac{f(x_c + h_i e_i) - f(x_c - h_i e_i)}{2h_i} \quad \text{error} \leq \frac{\gamma}{6}h_i^2 \tag{9}$$

$$H_{ij}(x_c) \simeq \frac{f(x_c + h_i e_i + h_j e_j) - f(x_c - h_i e_i) - f(x_c - h_j e_j) + f(x_c)}{h_i h_j}$$

$$\text{error} \leq \frac{5\gamma h}{6}. \tag{10}$$

It is very interesting to remark that with a properly chosen FD increment the convergence properties of Newton's method are preserved, therefore reducing the number of function evaluations. The increment should be $h_i \approx 10^{-\mathrm{DIGITS/n}}$ max $\{|\mathrm{x_i}|$, typical $x_\mathrm{i}\}$ sign (x_i) where $n = 2$ to compute the gradient, and $n = 3$ for the Hessian, and DIGITS is the number of reliable base 10 digits in $f(x)$.

4.2. Modified Newton's methods

Applying Newton's method far form the optimum can lead a quadratic model that does not properly represent the non-linear function $f(x)$, or even lead to a non-positive definite Hessian matrix, therefore invalidating the convex quadratic model. In such cases a full Newton's step $s_k = -H_k^{-1}g_k$ can lead to an inadequate estimate $x_{k+1} = x_k + s_k$. Two families of strategies, termed globally convergent modifications of Newton's method have been devised (in

the sense of assuring the convergence from almost any starting point, but the convergence to the absolute optimizer is not assured) to deal with this problem: line search and model trust region.

Line search. The idea is very simple: given a descent direction, p_k, take a step in that direction that yields an acceptable $x_{k+1} = x_k + \lambda p_k$. The acceptable step, λ, is obtained backtracking form the full Newton's step, if necessary, until a new estimate is obtained. It is not necessary that λ minimizes the function $r(\lambda) = f(x_k + \lambda p_k)$, but that the step fulfils the following conditions:

$$r(\lambda) \leq r(0) + \alpha\lambda r'(0) \tag{11}$$

$$r'(\lambda) \geq \beta r'(0), \quad \beta > \alpha \tag{12}$$

where α and β are properly chosen parameters. These conditions guarantee that both the sequence of estimates is monotonically decreasing, and converges to a minimizer. There are scores of different algorithms, which use quadratic, cubic or mixed models of the line, when not bisection or any other one-dimensional technique [11].

Model-trust region. The idea in this family of global strategies is to define a ball of radius δ_c around the actual estimate x_c where the quadratic function $m(x_c + s)$ can be trusted to model adequately the function $f(x_c + s)$. A step outside this trust region is not allowed and therefore the step is defined by a constrained minimization problem:

$$\min_s m_c(x_c + s) = f(x_c) + g(x_c)^T s + \tfrac{1}{2} s^T H(x_c) s$$

subject to $\|s\|_2 \leq \delta_c$.

This problem reduces to an unconstrained minimization one:

$$\min_s m_c^{\mu}(x_c + s) = f(x_c) + g(x_c)^T s + \frac{1}{2} s^T (H(x_c) + \mu I) s$$

whose solution is given by an augmentation of the Hessian

$$s(\mu) = -(H(x_c) + \mu I)^{-1} g(x_c)$$

where $\mu \geq 0$ is such that $\|s(\mu)\| = \delta_c$, unless $\|s(0)\| \leq \delta_c$, in which case $s(0) = H(x_c)^{-1} g(x_c)$ is the appropriate step.

Different approaches to solve for μ the equation $\|s(\mu)\| - \delta_c = 0$ leads to variants of the model trust region, such as the locally constrained optimal ('hook') step and the double dog-leg step. The trust radius is updated by backtracking if the first estimation does not provide a satisfactory step, and by heuristic rules when a proper step is attained.

4.3. Secant methods

The use of Newton methods requires the computation of the Hessian of the non-linear function $f(x)$. In IIP the Hessian is not easily available and its computation by FDs is expensive. The secant methods is a class of algorithms that use cheaper ways of approximating the Hessian, usually updating the approximate Hessian in the previous estimate.

The best Hessian update is provided by the positive definite secant update or BFGS (obtained independently by Broyden, Fletcher, Goldfarb and Shanno in 1970). The update is given by,

$$H_k = H_{k-1} + \frac{y_{k-1} y_{k-1}^T}{y_{k-1}^T s_{k-1}} + \frac{H_{k-1} s_{k-1} s_{k-1}^T H_{k-1}}{s_{k-1}^T H_{k-1} s_{k-1}}$$

where $y_{k-1} = g_k - g_{k-1}$ and $s_{k-1} = x_k - x_{k-1}$; $g_k = \nabla f(x_k)$ is the gradient.

The algorithm can be initialized computing the Hessian, $H_0 = \nabla^2 f(x_0)$ by FDs, or, if computationally expensive, simply by $H_0 = I$.

Save for the calculation of the Hessian by updating, the rest of the minimization algorithm remains as in Newton's method, so the globally convergent modifications are equally applicable for the secant methods.

The practical interesting property of the method is that it converges q-superlinearly ($\|x_{k+1} - x_k\| \leq c_k \|x_k - x_*\|$).

There are other updates, such as the Powell–Symmetric–Broyden update, the Davidon–Fletcher–Powell update, and the Inverse–Positive–Definite–Secant update. But the practice has stated the Broyden–Fletcher–Goldfarb–Shanno (BFGS) or Positive Definite Secant update as the best option.

4.4. Non-linear least squares problem

Newton's methods and the secant updates can be extended to the non-linear least square problem, taking advantage of its special structure. This problem is defined by,

$$\min_{x \in \mathcal{R}^n} f(x) = \frac{1}{2} R^T(x) R(x)$$

where $R : \mathcal{R}^n \rightarrow \mathcal{R}^n$ is the non-linear residual function, and $r_i(x)$ and the n components of $R(x)$.

The Jacobian of R is given $J = (\partial r_i / \partial x_j)$, and then $\nabla f(x) = g(x) = J^T(x) R(x)$; $\nabla^2 f(x) = H(x) = J^T(x) J(x) + \sum_i r_i(x) \nabla^2 r_i(x)$.

4.4.1. Gauss–Newton method

This method is based on approximating the Hessian of $f(x)$ by

$$H_k = J_k^T J_k$$

i.e. neglecting the contribution $S(x) = \sum_i r_i(x) \nabla^2 r_i(x)$, and then applying the basic algorithm in Section 4.1.

The rationale of this approximation is that the term $S(x)$ will be small when the solution is a zero-residual one, or if the residual function is not too non-linear. In these cases this algorithm converges q-linearly and even q-quadratically. The disadvantage is that it is not necessarily globally convergent, and that it is not well defined if J does not have full column rank.

It can be shown that Gauss–Newton's step,

$$s_k = -\left(J_k^T J_k\right)^{-1} J^T R_k$$

is a descent direction. In this case the globally convergent modifications mentioned in Section 4.2 can be applied, leading to the damped Gauss–Newton method and Levenberg–Marquardt method.

4.4.2. Damped Gauss–Newton method

In this method the next estimate is computed by,

$$x_{k+1} = x_k - \lambda\left(J_k^T J_k\right)^{-1} J_k^T R_k$$

where λ is obtained by a one-dimensional line search. This method is globally convergent although it may be very slow. Besides, the method is not well defined for non-full column rank Jacobian $J(x)$.

4.4.3. Levenberg–Marquardt method

This is the Gauss–Newton method modified by the model trust region approach. Then, the new estimate is obtained by

$$x_{k+1} = x_k + s_k(\mu)$$

where $s_k(\mu) = -(J_k^T J_k + \mu I)^{-1} J_k^T R_k$, and μ is obtained solving the equation $\|s_k(\mu)\| - \delta_k = 0$, except if $s_k(0) \leq \delta_c$, when $\mu = 0$.

This modification improves the behaviour of the algorithm for J with not full column rank, and for big second terms $S(x)$ in the Hessian. Depending on the strategy to approximate μ different variants of the Levenberg–Marquardt method have been coded.

4.5. Scaling

The different order of magnitude of the variables involved in a problem, and especially inside the design parameters (for example the size of the hole with respect to the total size, or the mixture of displacement and stress measurements) cause important numerical and algorithmic problems. A homogeneization is therefore crucial, and is done by a scaling of the magnitudes.

Whereas Newton's and BFGS methods are unaffected by scaling, the steepest descent and therefore the trust region models are affected. Therefore, the values introduced in the algorithms should previously be modified by a scaling matrix D_x in the form, $\hat{x} = D_x x$.

There is a further effect that one should care about. Too different values may also affect the conditioning of the matrices due to the computer precision, not only in the optimization algorithms, but also in the BEM calculations. The solution is similar to that described before.

5. Equivalence between the solution of the observation equations and the minimization of residual

The procedure summarized in Section 3.4 consists of updating the design variables by computing iteratively their increment through the solution of the so-called variation boundary integral equation (δBIE) [9]. The procedure follows these steps:

1. k = 0, initialize z_k
2. Compute $v_k = F(z_k)$ (direct problem)
3. Compute the matrix $A_k = A(z_k)$ (see Section 3.4)
4. Solve $A_k\,\delta z_k = \delta v_k = v^{\text{ex}} - v_k$ by least squares, i.e. $\delta z_k = (A_k^T A_k)^{-1} A_k^T (v^{\text{ex}} - v_k)$
5. Update $z_{k+1} = z_k + \delta z_k$
6. Repeat 2–6 until convergence.

On the other hand, the IIP can be as well solved by setting a non-linear least squares problem (see Section 1.2),

$$\min_z f(z) = \frac{1}{2}\|F(z) - v^{\text{ex}}\|^2 = \frac{1}{2} R(z)^T R(z)$$

where $R(z) = F(z) - v^{\text{ex}}$. The Gauss–Newton method for this residual function is,

1. k = 0, initialize z_k
2. Compute $R_k = F(z_k) - v^{\text{ex}}$
3. Compute the Jacobian $J_k = \nabla_z R(z_k)$
4. Compute the step $s_k = -(J_k^T J_k)^{-1} J_k^T R_k$
5. Update $z_{k+1} = z_k + s_k$
6. Repeat 2–6 until convergence.

It is easily checked that $s_k = \delta z_k$, since, firstly, $A_k = \nabla_z F(z_k)$ as mentioned at the end of Section 3.4, and therefore $A_k = \nabla_z (F(z_k) - v^{\text{ex}}) = \nabla_z R(z_k) = J_k$; secondly, $v^{\text{ex}} - v_k = v^{\text{ex}} - F(z_k) = -R_k$, and therefore, both steps are the same. In short, the solution of the IIP by the presented direct update approach is equivalent to the solution of a corresponding non-linear least squares problem by the Gauss–Newton method.

Actually the direct update is merely the solution of the observation equations, $F(z) = v^{\text{ex}}$, by Newton's method, but

Table 1
Methods tested

Method	Exact gradient (Gr)	Line search (LS)
Gauss–Newton (G–N)	Yes	No
	Yes	Yes
	No	Yes
Levenberg–Marquardt (L–M)	Yes	No
	Yes	Yes
	No	Yes
BFGS	Yes	Yes
	No	Yes

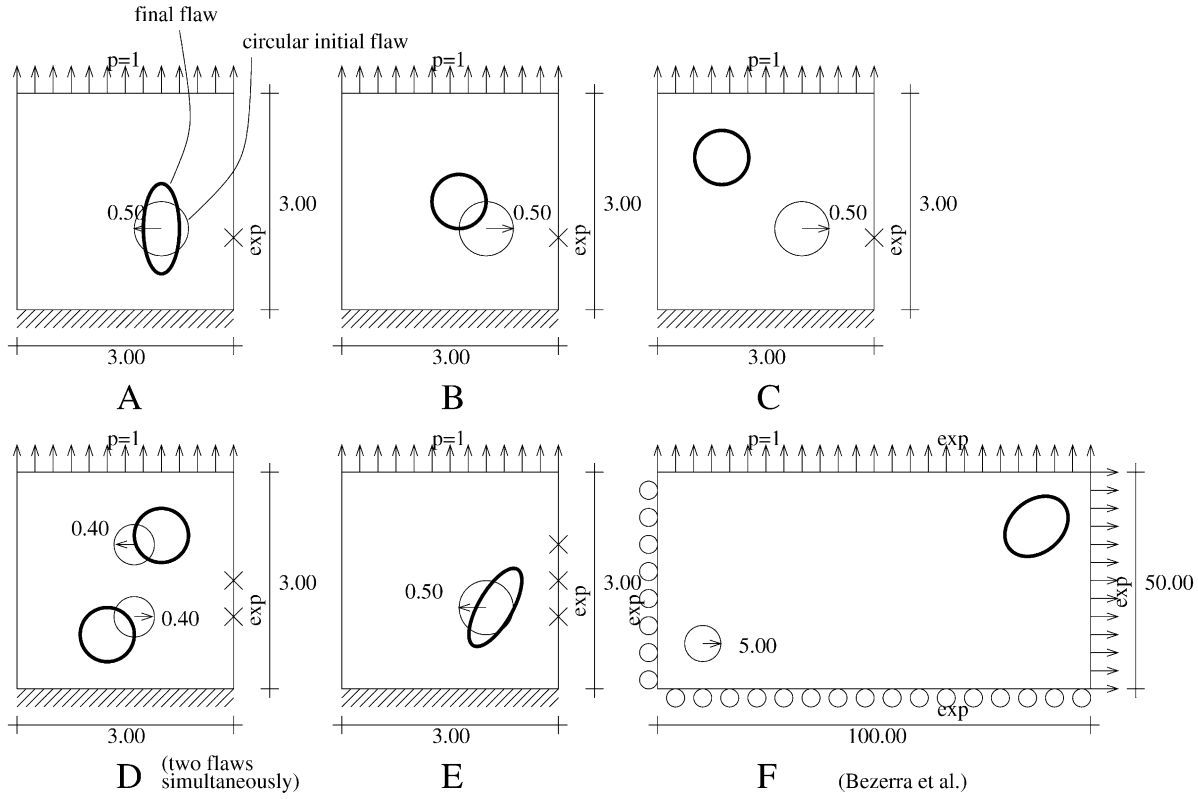

Fig. 3. Problems.

taking into account that these equations form an over-determined system of non-linear equations.

This link between the two approaches may allow for the adaptation of techniques of one into the other, suggesting new procedures. Examples of possible study in this direction could be:

- Use the singular value decomposition of the linearized observation equations onto minimization algorithms in order to damp higher singular values as a regularization technique (see Ref. [15], chapter 7, and Ref. [10], chapter 2).
- Application of theory of factor analysis (see Ref. [15], chapter 10), both after an adaptation to non-linear theory.
- Application of truncated least squares techniques to the definition of the minimization functional (see Ref. [10], chapter 3).

6. Numerical results

6.1. Methodology

The following tests are aimed at finding the advantages and disadvantages of the most suitable optimization algorithms for IIP in static 2D elasticity. Three algorithms are tested, and different variants of them, as shown in Table 1.

For each iteration, the problem is divided into three stages:

1. Calculation of a direct problem with the BEM. For this purpose, a simple code for two-dimensional elasticity with standard quadratic elements has been implemented. The details of the formulation can be found in Refs. [6,8]. The code was written in FORTRAN90 and run on an HP700 workstation.
2. Calculation of the sensitivities matrix $A(z)$ that comes from the δBIE. This computation is implemented with the former code in order to take advantage of common calculations. The formulation is detailed in Ref. [18].
3. The optimization algorithm computes the iteration step s_k. This algorithm has been taken from the Matlab Optimization Toolbox, version 5, with the proper modifications to match the needs of the problem.

Table 2
Parameters of the benchmark problems

Problem	A	B	C	D	E	F
Number of exterior elements	12	12	12	12	12	24
Number of flaw elements	4	4	4	8	4	8
Total number of elements	16	16	16	20	16	32
Number of design variables	5	2	2	4	5	2
Number of experimental data	5	2	2	8	5	80

Table 3
Summary of results (number of function calculations/number of gradient requests)

Method	Gr	LS	A	B	C	D	E	F	Mean	Success (%)
G–N	Yes	No	8/8	–/–	–/–	5/5	–/–	31/31	14.7/14.7	50
	Yes	Yes	13/5	–/–	–/–	17/6	–/–	35/–	21.7/5.5	50
	No	Yes	10/0	–/–	–/–	41/0	–/–	–/–	25.5/0.0	33
L–M	Yes	No	7/7	6/6	6/6	7/7	17/17	15/15	9.7/9.7	100
	Yes	Yes	13/5	14/5	23/8	17/6	42/13	–/–	21.8/7.4	83
	No	Yes	10/0	23/0	34/0	41/0	64/0	40/0	42.4/0.0	100
BFGS	Yes	Yes	9/3	20/6	23/7	21/5	61/17	35/–	28.2/7.6	100
	No	Yes	12/0	32/0	37/0	64/0	61/0	–/–	41.2/0.0	83

6.2. Description of the benchmark problems

A set of simple models that enhance different aspects of the IIP each, as well as some from other authors for a comparison have been chosen. They range from relatively easy ones to more challenging problems, although practical problems could become much more complicated. For each problem, all the algorithms in the previous table are tested. The geometry, loads, boundary conditions and experimental data, are sketched in Fig. 3. The number of elements, design variables and measurement data are shown in Table 2.

Problems A to D consist of a 3×3 square with one or two circular flaws. Problem F consists of a 100×50 rectangle with different boundary conditions and an elliptical interior flaw. This problem was solved by Bezerra et al. [3]. Here only two parameters are allowed from a starting circle, allowing horizontal and vertical displacement. The solution converges to a circular configuration close to the real elliptical one. Bezerra et al. solved this problem in 35 iterations.

6.3. Results

In Table 3 the results in terms of function evaluations and gradient evaluations required are summarized. The problems have been sorted from easier to more complex, and (–/–) means no convergence. Note that where no exact gradient is supplied, the given figures are the number of function evaluations, which is higher than the number of iterations (number of function evaluations = number of iterations + number of design variables × number of gradient requests). On the other hand, the exact gradient computation adds some extra computational time, since involves the calculation of matrix $\mathbf{\Delta}$, plus the solution of the system matrix, already factorized however, for each column of that matrix. This extra time does not amount to a complete function evaluation, since this latter involves the setting and solution of a whole new system of equations $\mathbf{Hu} = \mathbf{Gp}$.

The computing time has not been evaluated directly; only the number of function evaluations and exact gradient evaluations, since no particular code optimization has been performed.

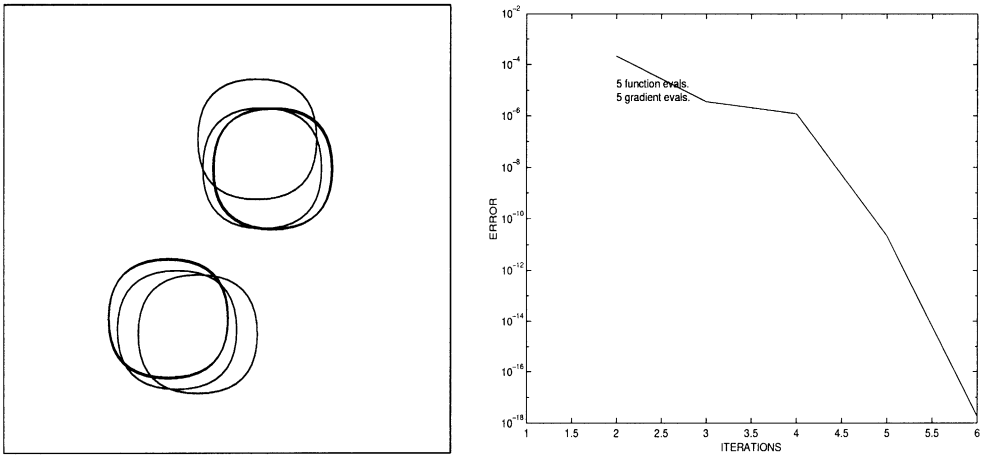

Fig. 4. Example D. Gauss–Newton (exact gradient and no line search). Left: geometries at starting and intermediate iterations. Right: error or cost functional f at each iteration.

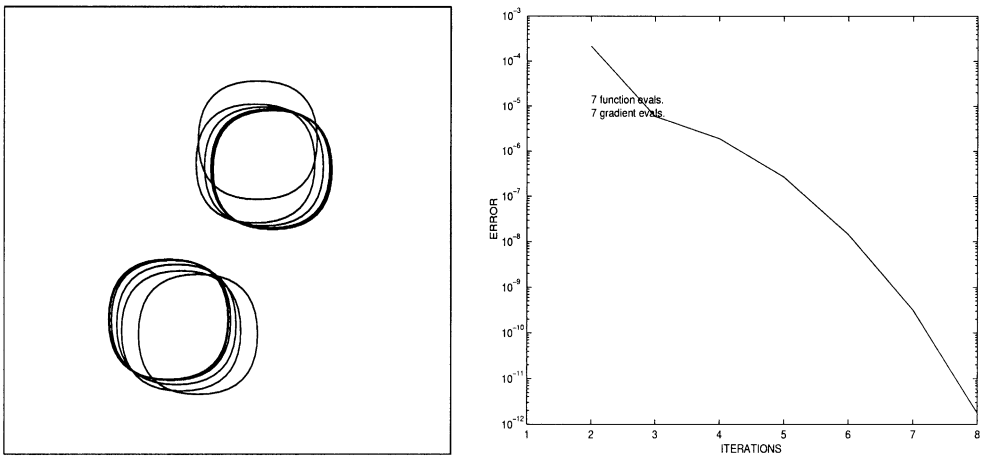

Fig. 5. Example D. Levenberg–Marquardt (exact gradient and no line search).

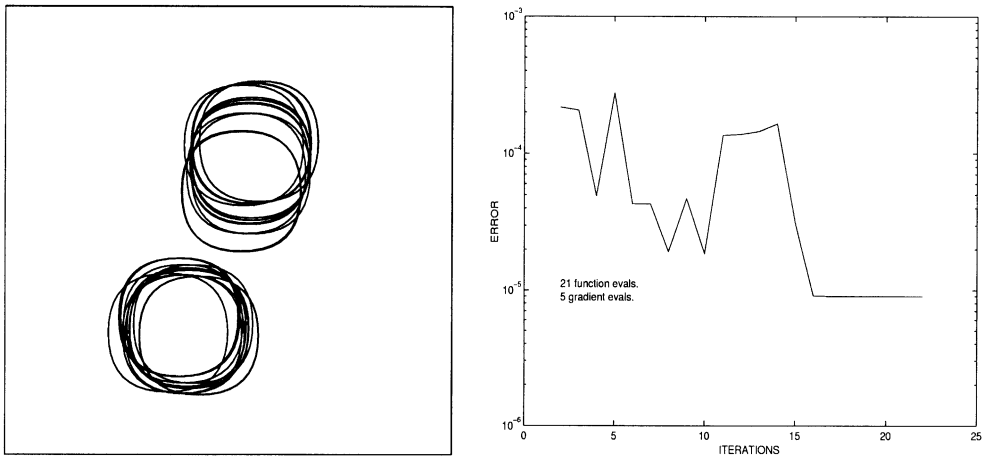

Fig. 6. Example D. BFGS (exact gradient and line search).

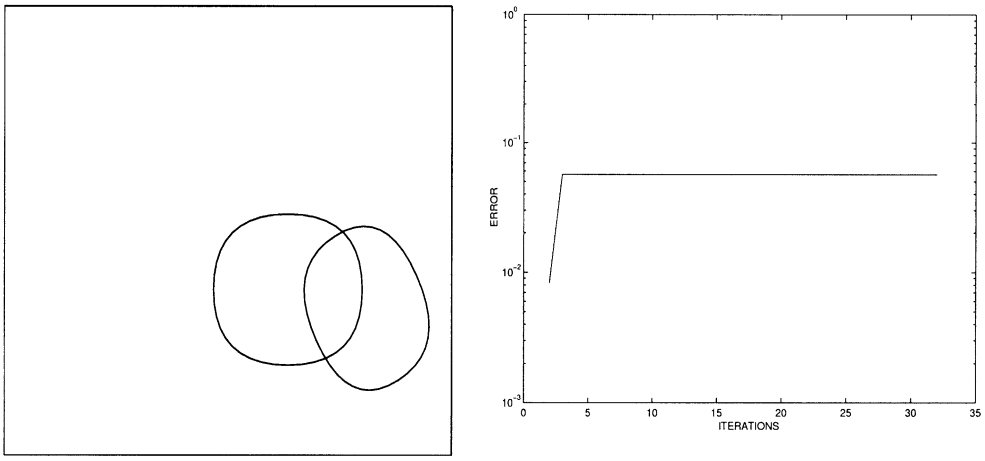

Fig. 7. Example E. Gauss–Newton (exact gradient and no line search).

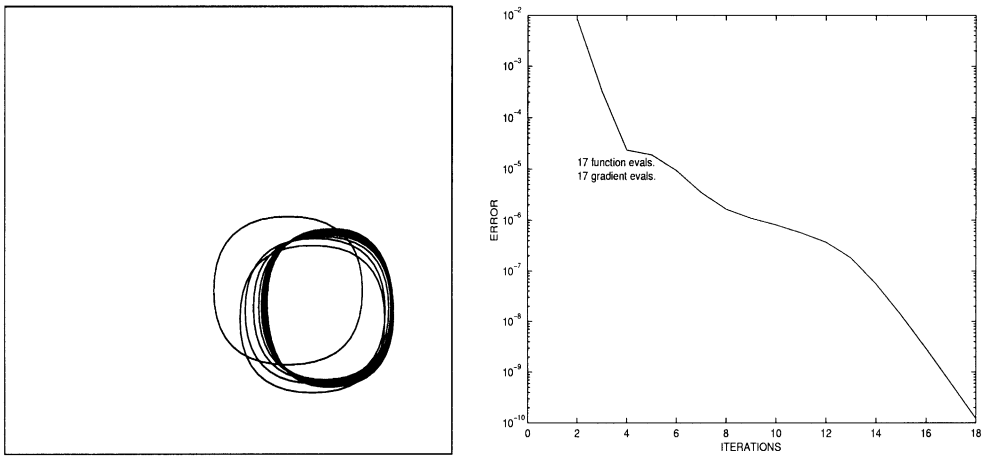

Fig. 8. Example E. Levenberg–Marquardt (exact gradient and no line search).

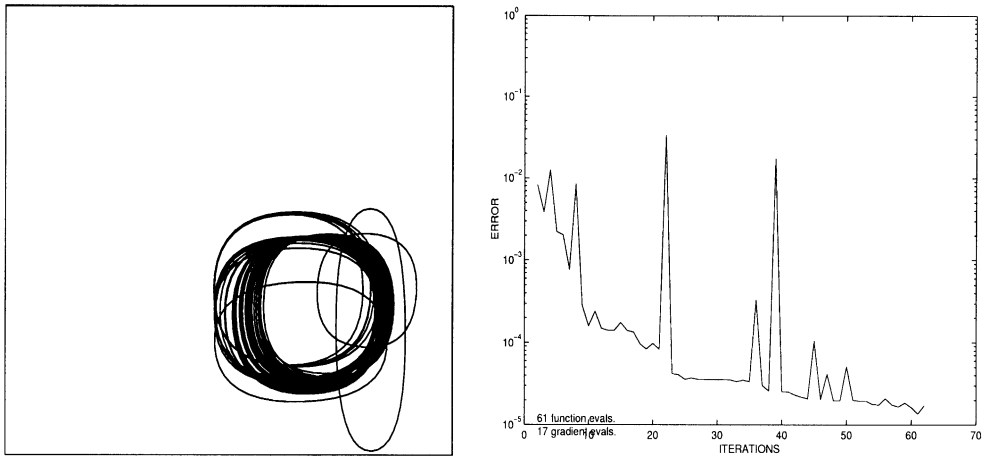

Fig. 9. Example E. BFGS (exact gradient and line search).

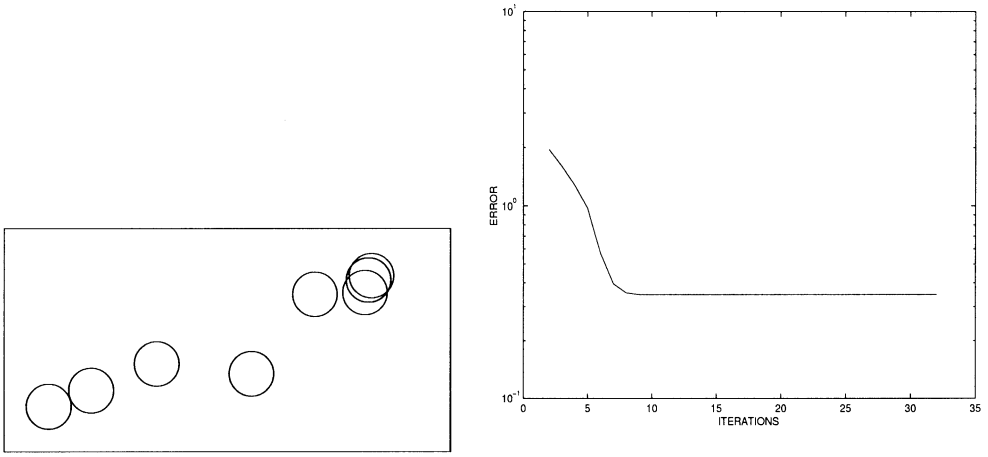

Fig. 10. Example F. Gauss–Newton (exact gradient and no line search).

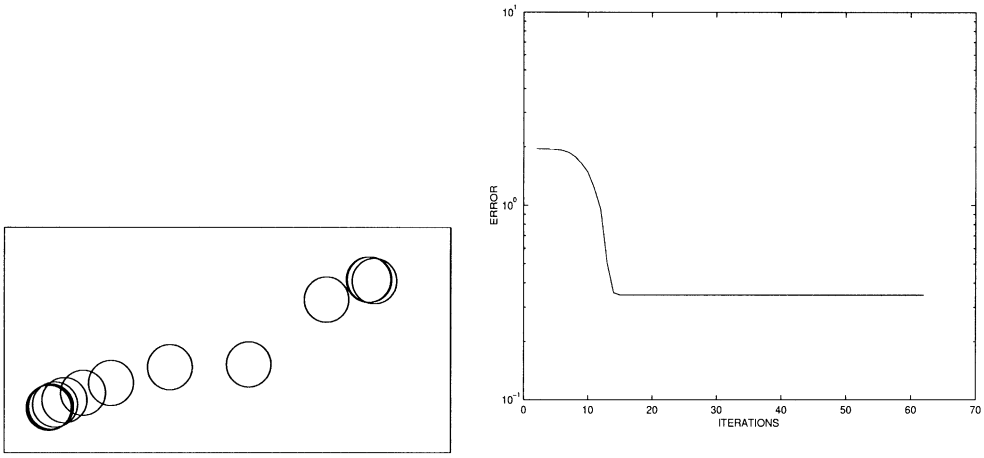

Fig. 11. Example F. Levenberg–Marquardt (exact gradient and no line search).

The graphics of the evolution of the geometry and cost function $f(z) = 1/2R(z)^T R(z)$ are shown in Figs. 4–12.

7. Conclusions

In this paper different methods for the solution of IIPs casting them as unconstrained non-linear optimization problems are compared.

From the point of view of the global convergence, i.e. capacity to arrive to an optimum from a badly chosen initial guess, the Gauss–Newton approach is the worst, whereas there is not very much difference between Levenberg–Marquardt and BFGS (as seen in the number of achievements in Table 3.

The use of a line search does not improve very much the solution. A reason for this is that the problem is actually constrained by the condition of the flaw remaining inside the domain. Therefore, the step is shortened when an impossible configurations is reached.

The convergence rate emphasizes the difference between BFGS and the two first methods. GN and LM are very similar for easy problems, but when the global strategies are necessary (e.g. test F), LM appears to be the fastest. The use of a line search is a critical choice. In easy problems surprisingly it retards very much the solution, but it becomes necessary for complex problems.

The use of analytic gradient instead of FDs reduces the computations as much as one should expect: by 20–50% for complex cases, and by 50–80% when the line search is not necessary.

To sum up, each method has shown some advantages and disadvantages. They are summarized in Tables 3 and 4.

- The Levenberg–Marquardt method provides both robustness and high convergence rate.
- The global strategy is important in complex problems, but it retards the procedures near the solution.
- The use of analytic gradient regularly accelerates the calculation.

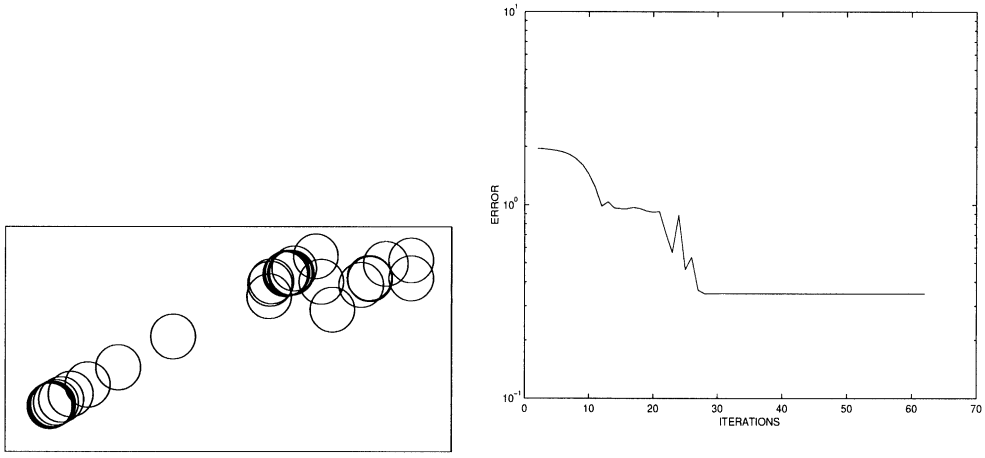

Fig. 12. Example F. BFGS (exact gradient and line search).

Table 4
Characteristics summary of the optimization algorithms

	Gauss–Newton	Levenberg–Marquardt	BFGS
Convergence rate vs. probability of convergence	High convergence rate	Good compromise	High probability of convergence
Particularities	Good for local scope (near the solution)	Good for global scope (far from the solution)	Does not take advantage of the structure of the cost functional

In addition, a relationship between the two main types of inverse problem layouts, the observation equation methods and the minimization methods has been found. In particular, it is proved that the direct solution of the linearized observation equation by least squares is identical to the simple Gauss–Newton method. This may unify the methods allowing for transferring the advantages and techniques from one to the other.

References

[1] Abdallah JB. Inversion gaussienne appliqué à la correction paramétrique de modéles structuraux. PhD thesis, Ecole Polytechnique, 1995.

[2] Aithal R, Saigal S. Shape sensitivity in thermal problems using bem. Engng Anal Boundary Elem 1995;15:115–20.

[3] Bezerra LM, Saigal S. A boundary element formulation for the inverse elastostatics problem (iesp) of flaw detection. Int J Numer Meth Engng 1993;36:2189–202.

[4] Bonnet M. Shape identification using acoustic measurements: a numerical investigation using boundary integral equation and shape differentiation. In: Bui HD, Tanaka M, editors. Inverse problems in engineering mechanics, 1992. p. 191–200.

[5] Bonnet M. Boundary integral equations and material differentiation applied to the formulation of obstacle inverse problems. Engng Anal Boundary Elem 1995;15:121–36.

[6] Brebbia CA, Domínguez J. Boundary elements, an introductory course. New York: CMP, Mc-Graw Hill, 1992.

[7] Burczynski T, Kuhn G, Antes H, Nowakowski M. Boundary element formulation of shape sensitivity analysis for defect identification in free vibration problem. Engng Anal Boundary Elem 1997.

[8] Domínguez J. Boundary elements in dynamics. Amsterdam: Elsevier, CMP, 1993.

[9] Gallego R, Suarez J. Solution of inverse problems by boundary integral equations without residual minimization. Int J Solids Struct 1999;37:5629–52.

[10] Hansen PC. Rank-deficient and discrete ill-posed problems, Numerical aspects of linear inversion. Philadelphia, PA: SIAM, 1997.

[11] MathWorks Inc. Matlab optimization toolbox user's guide.http://www.mathworks.com, 1996.

[12] Dennis Jr. JE, Schnabel RB. Numerical methods for unconstrained optimization and nonlinear equations. Philadelphia, PA: SIAM, 1983–1996.

[13] Kubo S. Classification of inverse problems arising in field problems and their treatments. In: Bui HD, Tanaka M, editors. Inverse problems in engineering mechanics, 1992. p. 51–60.

[14] Mellings SC, Aliabadi MH. Dual boundary element formulation for inverse potential problems in crack identification. Engng Anal Boundary Elem 1993;12:275–83.

[15] Menke W. Geophysical data analysis, discrete inverse theory. New York: Academic Press, 1984.

[16] Meric RA. Differential and integral sensitivity formulations and shape optimization by bem. Engng Anal Boundary Elem 1995;15:181–8.

[17] Nishimura N, Kobayashi S. A boundary integral equation method for an inverse problem related to crack detection. Int J Numer Meth Engng 1991;32:1371–87.

[18] Rus G, Gallego R. Solution of identification inverse problems by a sensitivity boundary integral equation. In: Suárez B, Oñate E, Bugeda G, editors. ECCOMAS2000, Barcelona. September 2000.

[19] Suárez FJ. Aplicatión del Método de los Elementos de Contorno a la Resulución del Problema Inverso en Elastodinámica. PhD thesis, Universidad de Granada, E.T.S.I. Caminos, C.y P. Noviembre 1998.

[20] Suzuki M, Murakami A. Bayesian estimation for nonlinear inverse problems. In: Tanaka M, Dulikravich G, editors. Inverse problems in engineering mechanics II, 2000.

[21] Tanaka M, Masuda Y. Boundary element method applied to some potential inverse problems. Engng Anal 1989;3-3:138–43.

[22] Zeng X, Saigal S. An inverse formulation with boundary elements. Trans ASME 1992;59:835–40.