Damage Identification by the Static Virtual Distortion Method

Przemysław Kołakowski
SMART-TECH Centre
Institute of Fundamental Technological Research
Polish Academy of Sciences
Swietokrzyska 21, 00-049 Warsaw, Poland
e-mail: pkolak@ippt.gov.pl

Key words:

Damage identification, inverse static analysis, structural optimisation

Abstract

This paper describes the application of a structural reanalysis technique called the Virtual Distortion Method, coupled with an optimisation technique called the Gradient Projection Method, to the damage identification problem for the case of the static structural analysis. The outline of the Virtual Distortion Method is provided and the fundamentals of the Gradient Projection Method are explained. Subsequently the damage identification problem for the static load case is posed. Numerical example of a truss structure is demonstrated. Conclusions are presented with the emphasis on advantages of the employed methods in numerical computations.

1. Introduction

The problem of damage identification in structural analysis is usually posed as a dynamic task of wave propagation. An excitation signal is applied and the resulting dynamic response is examined. A lot of research has been done on efficient signal processing methods analysing perturbations to the original signal due to structural damage, e.g. [1] - [4]. However, the currently used methods often encounter problems with identifying the damage properly and the related numerical cost may be considerable. A novel approach for solving the inverse dynamic problem has been proposed by Holnicki-Szulc and Zielinski [5].

The paper is a first-stage study of the damage identification problem tackled in the framework of the Virtual Distortion Method (VDM) and treated as a static task. For solving the optimisation problem involved, a constrained optimisation technique called the Gradient Projection Method (GPM) has been successfully employed. Advantage has also been taken of the Singular Value Decomposition (SVD), which is a matrix qualitative analysis method.

The continuation of the paper (cf. [6]) will deal with the VDM-based approach to the inverse dynamic problem, which may have practical applications to damage identification in various engineering structures (an experimental verification of feasibility of the VDM-based damage identification concept will be provided). The problem formulations in this paper and paper [6] are analogous, although the optimisation methods employed are different.

The Gradient Projection Method has been originated by Rosen [7], [8] in the early 1960's and then modified by Haug and Arora [9] in the late 1970's. Concise description of the method applied to structural analysis, given in Haftka and Gurdal [10], has been the basis for the Sec. 3 of the paper.

Most problems in structural analysis are constrained optimisation problems. The GPM is one of the techniques, which is able to minimise an objective function $f(x)$ subject to equality $h(x)=0$ and/or inequality constraints $g(x)\geq 0$. The inequality constraints divide the design space into two domains – the feasible domain, where the constraints are satisfied and the infeasible domain, where the constraints are violated. In many structural analysis problems, the objective function minimum is found on the boundary between the feasible and infeasible domains. The constraints, which belong to the boundary $g(x)=0$, are called active constraints and have great influence on the optimal solution. The constrains, which belong to the feasible domain $g(x)>0$ are inactive and could actually be removed without affecting the solution. The essence of the GPM is to find the optimum in the subspace tangent to the active constraints. Therefore the notion of the active constraints is so important here.

In the VDM-based damage identification problem we minimise certain function subject to inequality constraints. As we will see later on, the less damaged the structure is, the more active constraints are expected at the optimum. This justifies the choice of the GPM, which operates on active constraints in the optimisation process. In the engineering practice we often look at structures to which high technical requirements apply, e.g. pipelines. If such a structure is severely damaged, it is no longer operational. So we should rather expect a small number of structural defects and consequently, a large number of active constraints at our optimal solution of the damage identification problem.

2. Outline of the Virtual Distortion Method

2.1. Principal postulates

The Virtual Distortion Method has been developed for over 20 years. It may be simply classified as a fast reanalysis technique, according to Akgun, Garcelon and Haftka [11]. This method is very efficient if we know an original response of the structure and then want to introduce some modifications to its behaviour without repeating the whole analysis. With the VDM we are able to solve various problems of structural mechanics e.g. progressive collapse, structural remodelling, damage identification, damping of vibration, adaptive structure design and other.

Let us consider introducing a field of initial strains $\boldsymbol{\varepsilon}^{\mathbf{o}}$ (called *virtual distortions*) into a structure. This action will induce residual strains and stresses in the structure, expressed as follows (cf. [12]-[15]):

$$\varepsilon^{R} = D\varepsilon^{o}, \tag{2.1}$$

$$\sigma^{R} = E(D-I)\varepsilon^{o}, \tag{2.2}$$

where D denotes the so-called influence matrix telling how the whole structure responds to a unit virtual distortion $\varepsilon^{o}=1$ imposed in a chosen location, E denotes the constitutive matrix and I – the identity matrix. Assume that application of external load to the structure provokes linear elastic response ε^{L}, σ^{L}, which will be superposed over the residual response ε^{R}, σ^{R}. Thus in view of (2.1), (2.2), we get:

$$\varepsilon = \varepsilon^{L} + \varepsilon^{R} = \varepsilon^{L} + D\varepsilon^{o}, \tag{2.3}$$

$$\sigma = \sigma^{L} + \sigma^{R} = E\varepsilon^{L} + E(D-I)\varepsilon^{o} = E(\varepsilon - \varepsilon^{o}). \tag{2.4}$$

Virtual distortion field introduced in the structure may be twofold. We shall distinguish between the purely virtual distortions ε^{o} (having no physical meaning) used for modelling the structural geometry modifications (e.g. changes of cross-sectional area) and plastic-like distortions β^{o} used for simulating physical non-linearities in the structure. The plastic-like distortions are identified with plastic strains:

$$\beta^{o} \equiv \varepsilon^{pl}. \tag{2.5}$$

Thus in the elasto-plastic range of material behaviour, formulas (2.3), (2.4) have the following form:

$$\varepsilon = \varepsilon^{L} + D(\varepsilon^{o} + \beta^{o}), \tag{2.6}$$

$$\sigma = E\varepsilon^{L} + E(D-I)(\varepsilon^{o} + \beta^{o}) = E(\varepsilon - \varepsilon^{o} - \beta^{o}). \tag{2.7}$$

Relation between nodal forces f and generalised stresses σ for an element is known via the geometry matrix G, which also links generalised strains ε and nodal displacements q:

$$f = G^T \sigma , \tag{2.8}$$
$$\varepsilon = Gq . \tag{2.9}$$

Let us now take into account the structural geometry modifications exemplified by changes of Young's modulus. In a general case this means the analysis of a modified constitutive matrix E_m. In view of (2.7) and (2.8) we can express internal forces in the *original structure* with introduced virtual distortion field (called *distorted structure*) and in the *modified structure*, as follows:

$$f = G^T E\left(\varepsilon - \varepsilon^o - \beta^o\right), \tag{2.10}$$
$$f_m = G^T E_m\left(\varepsilon_m - \beta^o_m\right). \tag{2.11}$$

The main postulate of the VDM in structural remodelling requires that strains (including plastic strains) and forces in the distorted and modified structure should be equal:

$$\varepsilon = \varepsilon_m , \quad \beta^o = \beta^o_m , \quad f = f_m . \tag{2.12}$$

This postulate leads to the following relation:

$$G^T E\left(\varepsilon - \varepsilon^o - \beta^o\right) = G^T E_m\left(\varepsilon - \beta^o\right). \tag{2.13}$$

The above formulation applies to any structure, which is (e.g. truss) or can be (e.g. plate) made discrete in the sense of the Finite Element Method.

Let us confine our considerations in this paper to truss structures in the elastic range. If so, the geometry matrix G becomes identity and the plastic-like distortions β^o do not occur. Equation (2.13) provides the coefficient of the stiffness (Young's modulus) change for each truss element *i* as the ratio of the modified parameter to the initial one:

$$\mu_i = \frac{\left(E_m\right)_i}{E_i} = \frac{\left(A_m\right)_i}{A_i} = \frac{\varepsilon_i - \varepsilon^o_i}{\varepsilon_i} \; . \tag{2.14}$$

This coefficient is identical for the change of cross-sectional area of a truss element. The variation of the coefficient in the range $0 \le \mu \le 1$ may be considered as a measure of structural damage in the element. Substituting (2.3) into (2.14) we get a set of equations for ε^o, which must be solved to model an arbitrary damage of the structure with the coefficient μ. Indices *i*, *j* run through the damaged locations.

$$\sum_j \left(D_{ij} - \frac{\delta_{ij}}{1-\mu_i} \right) \varepsilon^o_j = -\varepsilon^L_i \; . \tag{2.15}$$

2.2. Generation of the influence matrix

The influence matrix introduced in the previous section forms a numerical basis for the VDM. It has been mentioned that component D_{ij} of the matrix determines the strain in structural element i, caused by unit distortion (initial strain) applied to element j. Thus, the influence matrix is (generally) non-symmetric. For truss structures with n elements, its dimension is n x n.

To compute one column of the influence matrix we must calculate responses in all structural elements caused by a local load properly applied to one of them. The so-called equivalent load (a pair of axial forces in case of truss structures) must correspond to the unit strain of the unconstrained element (see the single diagonal element in Fig. 2.1 after applying a pair of forces). The response of the structure to the imposition of the unit virtual distortion $\varepsilon^{o}_{4}=1$ is depicted by the "skewed" configuration in Fig. 2.1.

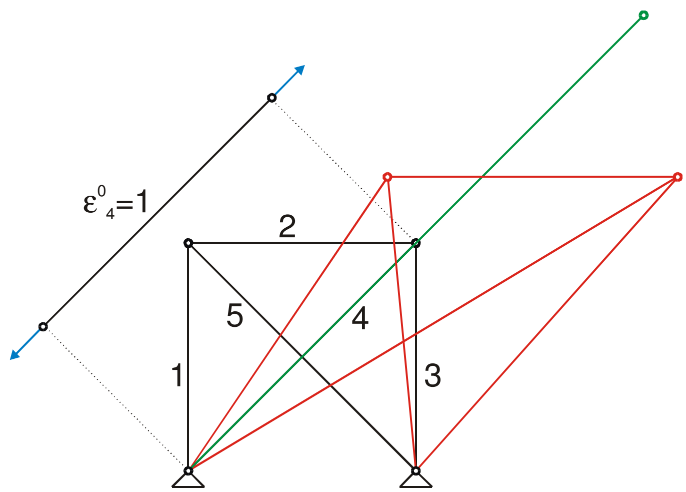

Fig. 2.1 Influence of the unit distortion applied in one element

Note that the static influence matrix for statically determinate structures becomes identity (zero redundancy means no inter-relations between the members) and the VDM loses its major tool.

If we assume that the virtual distortion depends on time (as well as for example the corresponding load, which is used to realise the distortion) then we can make use of the VDM in solving dynamic problems (cf. [6]). Consequently, the corresponding influence matrix will also be time-dependent, so it will be given another, third dimension.

3. Fundamentals of the Gradient Projection Method

3.1. Linear constraints

The Gradient Projection Method is based on the idea of projecting the direction of improvement (i.e. the direction in which the objective function value decreases) into the subspace tangent to the active constraints. For the case of linear constraints the optimisation problem can be posed in the following way:

$$\min\ f(x) \tag{3.1}$$

$$\text{subject to:}\quad g_j(x) = \sum_{i=1}^{m} n_{ij} x_i - b_j \geq 0, \quad j = 1,\dots,m_g\,, \tag{3.2}$$

where $n_{ij} = \dfrac{\partial g_j}{\partial x_i}$ i.e. the gradients of the constraints are stored column-wise. Subscript *i* runs through the number of design variables *m* whereas subscript *j* runs through the number of constraints m_g.

If we select only the m_a active constraints then the constraints (3.2) may be written as follows:

$$g_a = N^T x - b = 0\,, \tag{3.3}$$

where the matrix N stores gradients of the constraints in columns.

We minimise the objective function value by determining the direction of improvement s and looking for a current design variable vector x in an iterative way:

$$x_{k+1} = x_k + \alpha s\,, \tag{3.4}$$

where α denotes the step length.

The basic assumption of the GPM is that the design variable vector x lies in the subspace tangent to the active constraints, i.e. both the former design point x_k and the current one x_{k+1} satisfy equation (3.3). This assumption implies the following condition:

$$N^T s = 0\,. \tag{3.5}$$

In order to find the direction of improvement s let us now apply the steepest descent approach in minimising the objective function, i.e. let us look for a direction with the most negative directional derivative:

$$\min\ s^T \nabla f\,, \tag{3.6}$$

satisfying the equation (3.5) and being additionally normalised:

$$s^T s = 1. \tag{3.7}$$

We shall now construct the Lagrange function with the multipliers λ and η in order to get rid of the constraints (3.5) and (3.7) and make the optimisation problem unconstrained:

$$L(s,\lambda,\eta) = s^T \nabla f - s^T N\lambda - \eta\left(s^T s - 1\right). \tag{3.8}$$

In order to determine the stationary point of the Lagrange function we must zero the first derivative of L with respect to s:

$$\frac{\partial L}{\partial s} = \nabla f - N\lambda - 2\eta s = 0. \tag{3.9}$$

Pre-multiplying (3.9) by N^T and making use of (3.5) we can determine the Lagrange multipliers λ as:

$$\lambda = \left(N^T N\right)^{-1} N^T \nabla f. \tag{3.10}$$

The Kuhn-Tucker necessary criteria for optimality must be satisfied, so only non-negative Lagrange multipliers $\lambda \geq 0$ are of interest in the optimisation process. Constraints corresponding to negative values of λ are eliminated.

Substituting (3.10) into (3.9) we obtain:

$$s = \frac{1}{2\eta} P\nabla f, \tag{3.11}$$

where P is the so-called projection matrix expressed in terms of the matrix N as follows:

$$P = I - N\left(N^T N\right)^{-1} N^T. \tag{3.12}$$

To show that P has indeed the projection property it is enough to choose an arbitrary vector w and check that the vector Pw lies in the subspace tangent to the active constraints, i.e. the requirement (3.5) is met

$$N^T P w = 0, \tag{3.13}$$

which is easy to verify by taking into account the definition of P given in (3.12). Note that P=0 if N is a non-singular, square matrix. Then the tangent subspace reduces to a point.

The factor $1/2\eta$ is not significant as s defines only the direction of improvement, so instead of (3.11) we use the steepest descent direction in the tangent subspace:

$$s = -P\nabla f. \tag{3.14}$$

Substituting (3.12) into (3.14) and making use of the definition (3.10) we obtain an equivalent, useful formula for s, which will be used later in the algorithm for constructing the so-called projection move:

$$s = -(\nabla f - N\lambda). \tag{3.15}$$

3.2. Non-linear constraints

For the case of non-linear constraints we base on linear approximation of (3.2) in the form:

$$g_j = g_j(x_i) + \nabla g_j^T(\overline{x}_i - x_i). \tag{3.16}$$

Except for the projection move we also look now for the so-called restoration move $\overline{x}_i - x_i$ in the tangent subspace (i.e. $P(\overline{x}_i - x_i) = 0$), which reduces g_j to zero. It can be checked that the desired restoration move is:

$$x_{RESTORATION} = \overline{x}_i - x_i = -N(N^T N)^{-1} g_a(x_i), \tag{3.17}$$

where g_a is the vector of active constraints (cf. (3.3)).

In the modification of the GPM by Haug and Arora [9] it is proposed to specify a desired reduction γ ($0 \le \gamma \le 1$) in the objective function value, so that:

$$f(x_k) - f(x_{k+1}) \approx \gamma f(x_k). \tag{3.18}$$

Using a linear approximation with (3.4) and the assumption (3.18), we get the following projection move:

$$x_{PROJECTION} = \alpha s = -\frac{\gamma f(x_i)}{s^T \nabla f} s. \tag{3.19}$$

Haug and Arora's procedure is then a combination of the projection and restoration moves, as:

$$x_{k+1} = x_k + x_{PROJECTION} + x_{RESTORATION}, \tag{3.20}$$

where (3.4), (3.17) and (3.19) are used.

4. GPM in VDM-based damage identification for static problems

4.1. Problem formulation

We shall now pose the optimisation problem of structural damage identification in the framework of the Virtual Distortion Method. Let us minimise the following function:

$$\min \sum_A \left(\varepsilon_A^M - \varepsilon_A\right)^2 , \tag{4.1}$$

which can be interpreted as an average departure of the total strain ε_A from the experimentally measured strain $\varepsilon_A{}^M$ in locations *A*, capable of identifying the structural damage (called *sensors* hereinafter). Taking advantage of the VDM formulation (cf. (2.3)) we can decompose the strain ε_A into two parts:

$$\varepsilon_A = \varepsilon_A^L + \varepsilon_A^R = \varepsilon_A^L + \sum_A D_{Ai} \varepsilon_i^o , \tag{4.2}$$

where $\varepsilon_A{}^L$ denotes the response of undamaged structure, D is the influence matrix and ε^o is the virtual distortion vector (see Sec. 2). As the component $\varepsilon_A{}^L$ is constant for a given external static load, the so-called residual strain component $\varepsilon_A{}^R$ may only be varying in the optimisation process with the virtual distortion ε^o as the design variable.

We shall measure the structural damage in each element *i* with the help of the coefficient μ_i introduced by formula (2.14) i.e. with the ratio of cross-sectional area of a damaged element to the undamaged one. Consequently, we have to impose appropriate constraints on this coefficient. As we examine the physical process of deterioration of the element cross-section (e.g. due to corrosion), we are interested in such vector μ_i, which complies with the constraints allowing only for reduction of the cross-sectional area. On the other hand, only positive values of the vector μ_i may be considered in view of its definition (cf. (2.14)). Thus the constraints take the following form:

$$0 \le \mu_i \le 1 \quad \text{i.e.} \quad 0 \le \frac{\varepsilon_i - \varepsilon_i^o}{\varepsilon_i} \le 1, \quad i = 1,\ldots,m . \tag{4.3}$$

Each coefficient μ_i depends non-linearly upon the virtual distortion vector $\varepsilon^o{}_i$, so the constraints (4.3) are non-linear.

The gradients of the objective function and the right-hand side constraints (4.3) are expressed in terms of the design variable ε^o as follows:

$$\nabla f = \frac{\partial f}{\partial \varepsilon_i^o} = -2 \sum_A D_{Ai} \left(\varepsilon_A^M - \varepsilon_A\right) \quad \text{and} \quad N = n_{ij} = \frac{\partial g_j}{\partial \varepsilon_i^o} = \frac{\delta_{ji}\varepsilon_j - D_{ji}\varepsilon_j^o}{\left(\varepsilon_j\right)^2} . \tag{4.4}$$

We assume that every element of the structure may be subject to damage, so the index *i* runs through all structural elements *m*, the index *j* runs through the number of active constraints m_a, whereas the index *A* runs through the number of sensors m_s.

4.2. Numerical algorithm

In order to solve the damage identification problem posed by (4.1) and (4.3) we will use the Gradient Projection Method described in the Sec. 3. All symbols used beneath are compatible with those used previously.

Determination of the original projection matrix P (cf. (3.12)) may cause numerical problems because of the necessity of finding the inverse of the matrix N^TN. To avoid the inconvenience, an original algorithm has been developed, employing the GPM and making use of the Singular Value Decomposition (SVD) [16] for solving sets of equations. The algorithm is nested in the VDM programming environment (the influence matrix D is pre-computed) and performs the following steps:

1) Determine the matrix N and the vector ∇f on the basis of (4.4)
2) Compute the matrix N^TN and the vector $N^T\nabla f$
3) Solve the system

$$\left(N^T N\right)\lambda = N^T \nabla f \tag{4.5}$$

   for Lagrange multipliers λ using the SVD method (instead of using (3.10) directly)
4) If λ<0, eliminate the corresponding constraints, go back to 1) and redefine the matrix of active constraint gradients N, else proceed
5) Compute the projection move (cf. (3.15), (3.19)) as

$$x_{\mathrm{PROJECTION}} = -\frac{\gamma f\left(x_i\right)}{\nabla f^T\left(\nabla f - N\lambda\right)}\left(\nabla f - N\lambda\right) \tag{4.6}$$

   with the desired reduction of the objective function, e.g. γ=0.3
6) Determine the active constraints vector g_a on the basis of (3.3)
7) Solve the system

$$\left(N^T N\right)\lambda' = g_a \tag{4.7}$$

   for auxiliary multipliers λ' using the SVD method (instead of using (3.17) directly)
8) Compute the restoration move (cf. (3.17)) as

$$x_{\mathrm{RESTORATION}} = -N\lambda' \tag{4.8}$$

9) Calculate the design variable in the next iteration as a combination of the projection and restoration moves according to (3.20)
10) Go back to 1), proceed until the termination criterion is met

4.3. *Heuristic amendments to the algorithm*

The algorithm described in 4.2 has been implemented in the Fortran code named DAMIS after DAMage Identification for Statics [17].

Some heuristic amendments have been added to the algorithm to improve its efficiency. These are:

- To avoid singularity or ill-conditioning of the matrix N^TN (a situation in which the condition number of the matrix exceeds the assumed limit value, e.g. 10^3) in solving the set (40) with the SVD method, the constraints corresponding to "corrupted equations" are eliminated from the set (similarly to the negative λ's in Point 4 of the algorithm). In this way the SVD provides a reliable solution of an acceptably low residual. This problem occurs when some elements of the statically loaded structure are not strained at all or their strains are by several orders of magnitude lower than the strains in other elements.

- Instead of performing the proper restoration move according to (4.7) and (4.8), which requires solving a set of equations by SVD, it is equivalent to zero the design variables (virtual distortions ε^o) for the constraints classified as active.

- As we expect the optimum with a great number of active constraints μ=1 (undamaged elements), the formally imposed left-hand side constraint $0 \leq \mu$ has been disregarded (cf. (4.3)). Consequently the matrix N has become smaller (cf. (4.4)). However, if the constraint $0 \leq \mu$ is violated (the fact observed very rarely), the design variables are zeroed, i.e. the condition μ=1 is imposed.

5. Numerical example of a truss structure

5.1. Presentation of results

A 40-element cantilever truss structure, shown in Fig. 5.1a, consisting of 8 repeatable 5 m x 5 m segments, has been analysed to test the DAMIS code. The structure was statically loaded with the vertical force P=200 kN. All elements have the same tubular cross-sectional area A=201 cm^2 (65 cm diameter, 1 cm thickness) and Young's modulus E=210 GPa. The assumed geometry was meant to reflect the dimensions of real engineering structures (pipelines).

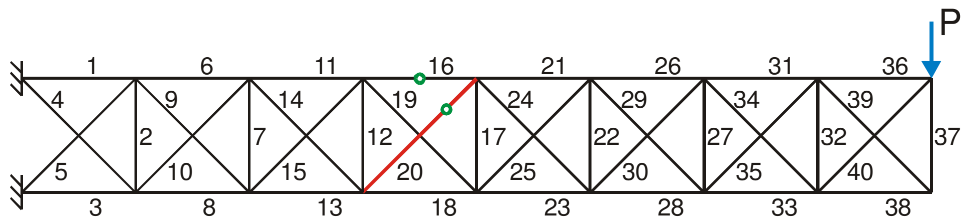

Fig. 5.1a *Two sensors (dots) for precise identification of one damaged element No. 20*

First, only one element No. 20 (see Fig. 5.1a) was chosen as damaged with the corresponding μ_{20}=0.5. The damage identification analysis was carried out assuming just one sensor in the same segment, from which the damaged element was picked up. Such "one by one" damage detection capability was checked and the elements were ranked on the basis of the obtained μ_{20} value in the following order: No. 16 (μ_{20}=0.507), No. 20 (μ_{20}=0.595), No. 19 (μ_{20}=0.783), No. 18 (μ_{20}=0.962).

In order to improve this result, another sensor (second in the ranking of "one sensor – one damage" detection) was added in the element No. 20, so that two sensors in elements Nos. 16 and 20 (marked by dots in Fig. 5.1a) were detecting one damage in the segment. The obtained identification results are shown in Fig. 5.1b, which depicts variations of the coefficient μ for an undamaged member (straight line marked by squares), the coefficient μ_{20} (curve marked by circles) and the objective function value (curve marked by rhombuses) as the optimisation process goes along. We can see that the damage in element No. 20 was precisely detected with all other elements being undamaged.

Subsequently, a pattern of 8 damage locations of various intensities was chosen such that one damage location was examined per segment. The corresponding damage coefficients were as follows: μ_1=0.9, μ_9=0.2, μ_{13}=0.8, μ_{20}=0.3, μ_{21}=0.7, μ_{29}=0.4, μ_{33}=0.6, μ_{40}=0.5 (see Figs. 5.2a and 5.2b). The information on the selected element capability of detecting damage, collected for the one damage case, was utilised. In general however, the trial and error approach was applied in determination of sensor locations. The minimum number of sensors giving the optimal solution turned out to be 12 (see dotted elements Nos. 1, 5, 8, 13, 14, 16, 21, 25, 28, 33, 34, 36 in Fig. 5.2a). The obtained results are depicted in Fig. 5.2b.

Alternatively, the same results were achieved by locating 20 sensors (dotted in Fig. 5.2c) in all 16 horizontal elements plus 4 diagonal elements placed in every other segment (e.g. in compressive diagonal elements Nos. 5, 15, 25, 35).

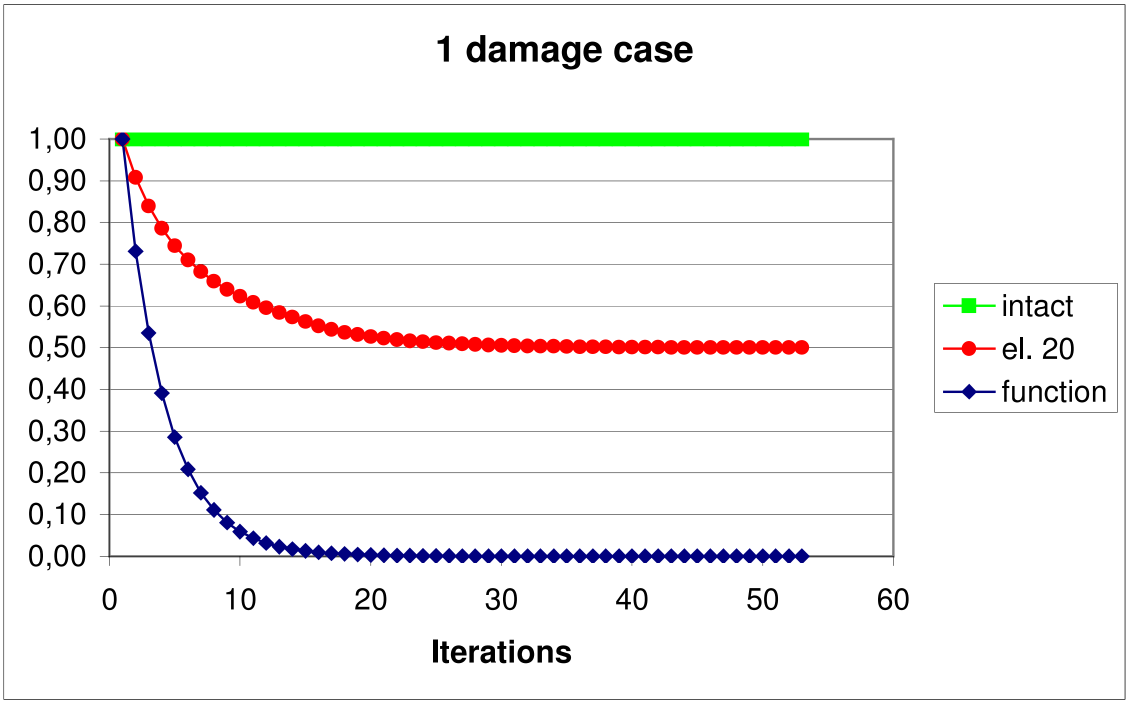

Fig. 5.1b *Results of the identification process for 1 damage case*

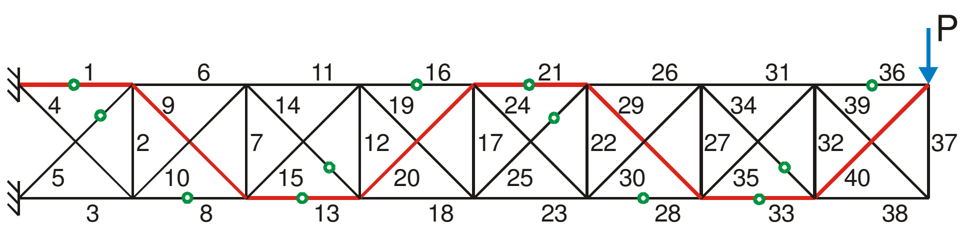

Fig. 5.2a *Minimum number of sensors (12) for identification of 8 damaged elements*

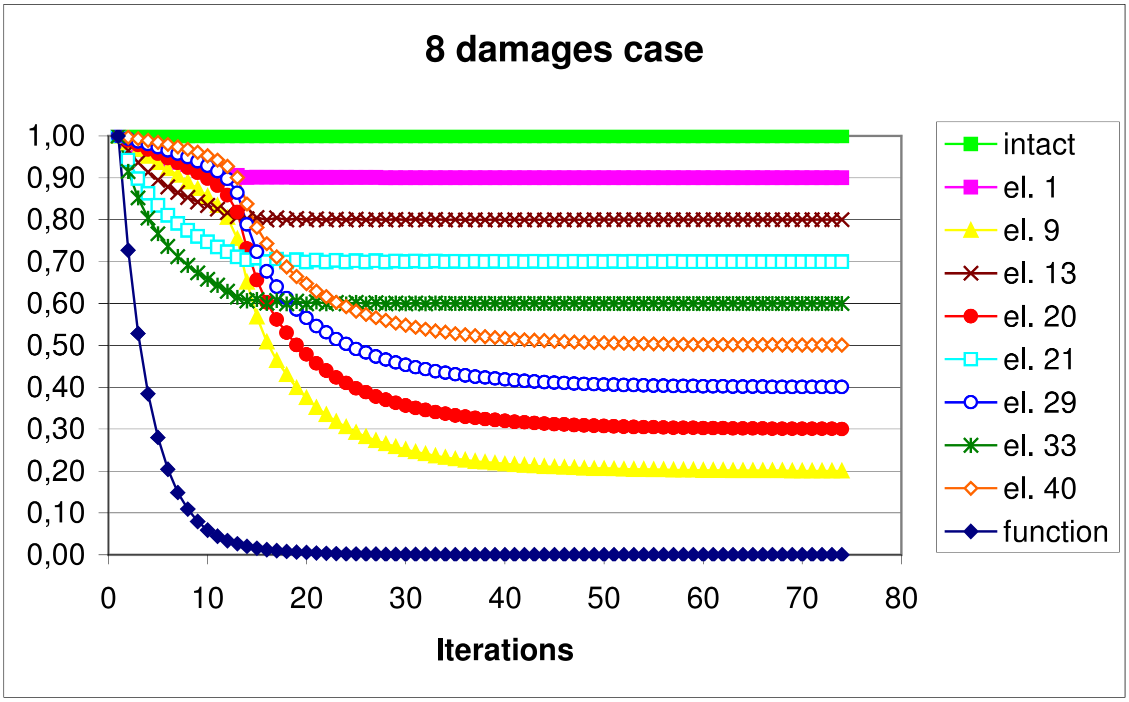

Fig. 5.2b *Results of the identification process for 8 damages case*

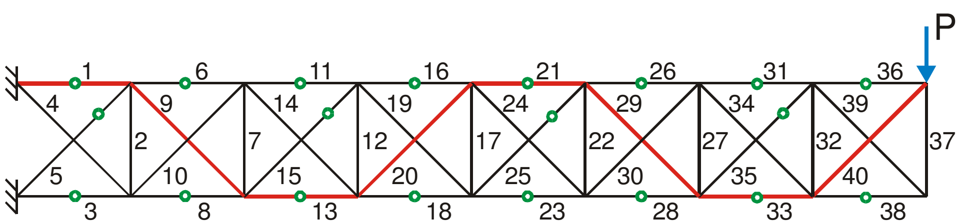

Fig. 5.2c *Configuration of 20 sensors for identification of 8 damaged elements*

Many vertical elements of the truss are lowly strained at the bending-provoking vertical force. As a consequence, the algorithm encounters difficulties in detecting damage in such elements. A way to get rid of the problem is to apply such load to which vertical elements are more sensitive, e.g. the load proposed in Fig. 5.3a. A damage identification was performed for the combined bending and axial load with one damage in the vertical element No. 17 (μ_{17}=0.5). Three sensor locations (in elements Nos. 17, 20, 25, marked with dots in Fig. 5.3a) provided precise identification of the assumed damage. The results are shown in Fig. 5.3b.

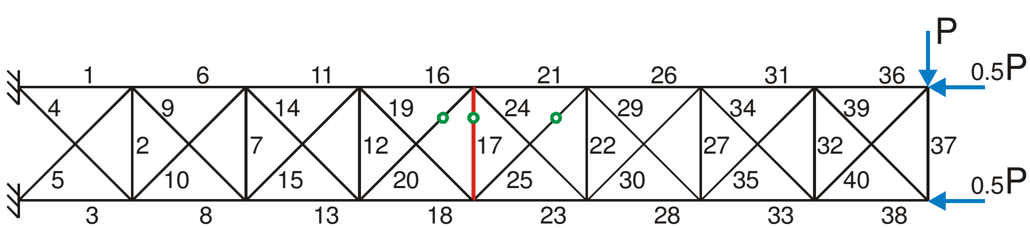

Fig. 5.3a *Three sensors for precise identification of damage in one vertical element*

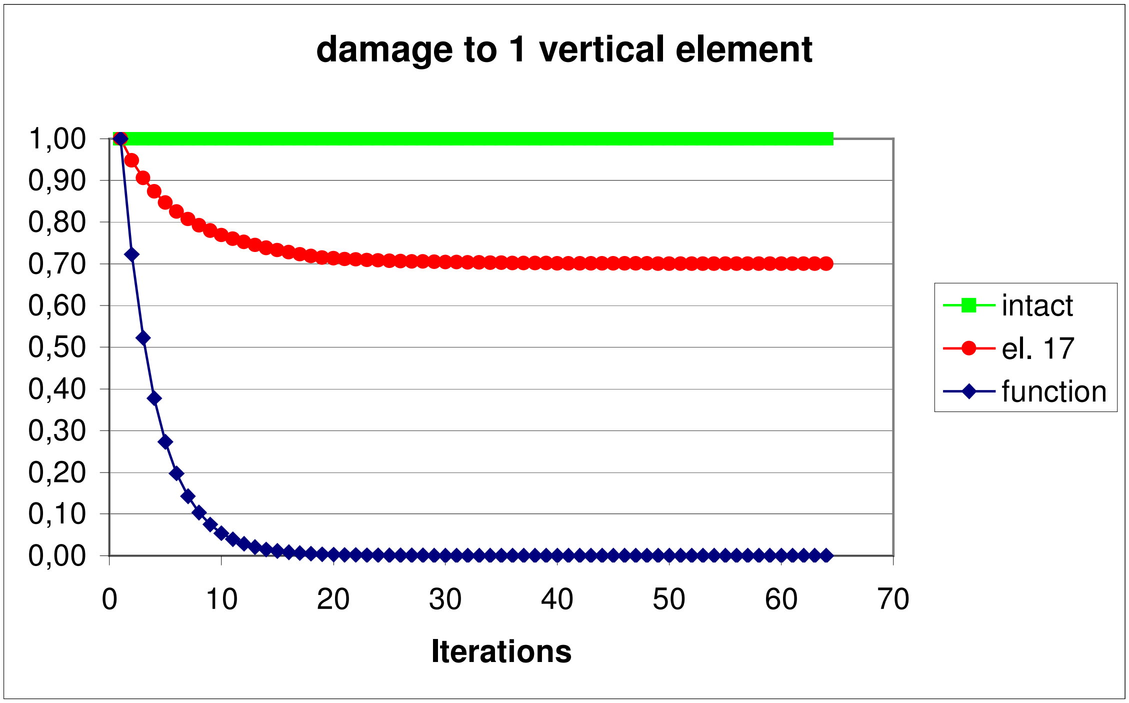

Fig. 5.3b *Results of the identification process for 1 damage in a vertical member*

5.2. Discussion on results and numerical simulation

In order to model various damage intensities in 8 elements, the set of equations (2.15) has to be solved for the assumed vector μ. The resultant distortions ε^o are imposed as an initial strain-type load and together with the external load P produce the damaged structure response ε^M (cf. (4.1)), which should be collected from measurements in real-life identification.

At the start of simulation, the design variables are zeroed (ε^o=0) and all constraints are active with μ=1. Thus the number of linearly independent active constraints is equal to the number of design variables and the tangent subspace reduces to a single point (P=0). Consequently the direction of improvement s cannot be determined. The situation changes when negative Lagrange multipliers corresponding to active constraints are eliminated. Then the matrix N becomes non-square (more rows than columns) and the algorithm may proceed.

A decrease of the objective function value by 7 orders of magnitude was set to be the criterion of computational analysis termination. Slight inaccuracies of the obtained damage coefficient results are due to this choice. More stringent criterion would result in a greater number of iterations.

The number of iterations needed to arrive at the optimal solution was quite satisfactory. With the arbitrarily chosen reduction of the objective function value γ=0.3, the optimum was reached after approximately 70 iterations. The computation took a few seconds with the AMD Athlon 1900+ XP processor.

The minimum number of sensors was determined thanks to the assumption of the damage pattern for numerical modelling. This was a great prompt on how to locate sensors optimally basing on results of the "one by one" damage detection ranking. In real life however, the damage distribution is something to be found. We would not be given any hints on optimal sensor location. Therefore it is practical to devise a certain scheme for locating the sensors, able to identify an arbitrary damage pattern. Such scheme is proposed in Fig. 5.2c, where all 16 horizontal elements plus 4 diagonals placed in every other segment are chosen for sensor locations.

A threshold in damage identification capability was observed. In the analysed example, the resulting virtual distortions corresponding to the detected damaged locations are more or less of the same order of magnitude. However, if they happen to differ by more than 2 orders of magnitude, the damage corresponding to "small" distortions cannot be detected. A "small" distortion means a relatively small damage intensity of the element (say μ=0.98). One should be aware that identification of such a damage would also be a serious problem in experimental measurements.

Sometimes, with a set of sensors being far from optimal, an increase of the objective function value is observed during the analysis (in spite of the desired 30% decrease) and the algorithm is forced to stop. This indicates that the chosen number of sensors is insufficient or their location is inappropriate for the damage identification posed as a static problem. There is simply not enough information for the optimisation algorithm to produce a good solution.

6. Final remarks

A new damage identification method, based on the Virtual Distortion Method as a structural reanalysis tool and on the Gradient Projection Method as an optimisation technique, has been proposed for the static load case. The GPM seems to be very well suited for the damage identification problem posed in Sec. 4.1 as it operates on the active constraints, which have crucial influence on the solution of the problem. In many real-life cases we would expect only local damage (inactive constraints) in the structure with most of the elements left undamaged (active constraints in the framework of the optimisation method used).

It has been shown that the GPM & VDM methods linked together produce appealing identification results (<u>several damaged elements with various damage intensities precisely detected</u>) for a properly chosen set of sensors (the elements of the structure, which contribute to the objective function value). There is also the third method behind the optimal solution – the SVD method – which provides very useful qualitative analysis of matrices.

Optimal location of sensors depends upon the external load applied and the corresponding structural response. It seems that the load provoking bending behaviour is more informative as far as damage identification capability of sensors is concerned (cf. Figs. 5.1a, 5.2a). Axial load may be treated as auxiliary in cases when elements are insensitive to the bending-type load (cf. Fig. 5.3a).

The main numerical problem encountered is caused by singularity or ill-conditioning of the matrix N^TN, which is due to unstrained or lowly strained elements (cf. Sec. 4.3). This may be overcome e.g. by applying such an external load that all elements are sufficiently strained (cf. Fig. 5.3a). Another way of avoiding the problem is to set a threshold for the condition number of the matrix we are still pleased with. The latter can be easily done by employing the SVD method in the analysis. Fortran code for the SVD algorithm can be found in [16].

The principal purpose of the presented static case was to prepare the background for dynamic case of VDM-based damage identification. Practical aspects of the concept feasibility for the static case (e.g. magnitude or direction of the applied forces in the presented numerical example) have been disregarded because experimental verification will be provided only for dynamics. Therefore in the static case the optimisation for finding the minimum of the defined objective function was performed assuming only noise-free response ε^M of the structure (modelled numerically instead of measured experimentally).

Further research will be concentrated on the VDM-based inverse dynamic analysis of damage identification utilising the phenomenon of elastic waves propagation (cf. [6]). A Fortran code DAMID for the dynamic case, analogous to the newly created code DAMIS for the static case, will be developed.

Acknowledgement

This work was financially supported by the grant No. KBN 5 T07A 052 22 awarded by the State Committee of Scientific Research, Warsaw, Poland.

References

[1] Kawiecki G. (1998) Feasibility of Applying Distributed Piezotransducers to Structural Damage Detection, *Journal of Intelligent Material Systems and Structures*, vol. 9, pp. 1-9

[2] Uhl T. (2001) Application of Modal Analysis for Monitoring and Diagnostics of Mechanical Structures, *Structural Control and Health Monitoring* edited by J. Holnicki-Szulc, AMAS Lecture Notes 1, IFTR, Warsaw, Poland

[3] Krawczuk M., Palacz M., Ostachowicz W. (2003) The Dynamic Analysis of Cracked Timoshenko Beam by the Spectral Element Method, *Journal of Sound and Vibration* 264: 1139-1153

[4] Ostachowicz W., Krawczuk M. (2003) Damage Detection of Structures using Spectral Element Method, *Proc. of SMART'03 Workshop on Smart Materials and Structures*, Jadwisin, Poland, 2-5 September, by Springer Verlag, to appear

[5] Holnicki-Szulc J., Zielinski T. G. (2001) Damage Identification Method based on Analysis of Perturbation of Elastic Waves Propagation, *Structural Control and Health Monitoring* edited by J. Holnicki-Szulc, AMAS Lecture Notes 1, IFTR, Warsaw, Poland

[6] Kolakowski P., Zielinski T. G., Holnicki-Szulc J. (2003) Damage Identification by the Dynamic Virtual Distortion Method, *Journal of Intelligent Material Systems and Structures*, to appear

[7] Rosen J. B. (1960) The Gradient Projection Method for Nonlinear Programming, Part I – Linear Constraints, *SIAM Journal of Applied Mathematics*, 8(1), pp. 181-217

[8] Rosen J. B. (1961) The Gradient Projection Method for Nonlinear Programming, Part II – Nonlinear Constraints, *SIAM Journal of Applied Mathematics*, 9(4), pp. 514-532

[9] Haug E. J., Arora J. S. (1979) *Applied Optimal Design: Mechanical and Structural Systems*, John Wiley & Sons, New York, U.S.A.

[10] Haftka R. T., Gurdal Z. (1992) *Elements of Structural Optimization* (3rd ed.), Kluwer Academic Publishers, Dordrecht, the Netherlands

[11] Akgun M. A., Garcelon J. H., Haftka R. T. (2001) Fast Exact Linear and Non-linear Structural Reanalysis and the Sherman-Morrison-Woodbury Formulas, *International Journal for Numerical Methods in Engineering*, 50, pp. 1587-1606

[12] Holnicki-Szulc J., Gierlinski J. T. (1995) *Structural Analysis, Design and Control by the Virtual Distortion Method*, John Wiley & Sons, Chichester, U.K.

[13] Kolakowski P., Holnicki-Szulc J. (1998) Sensitivity Analysis of Truss Structures (Virtual Distortion Method Approach), *International Journal for Numerical Methods in Engineering*, vol. 43, pp. 1085-1108

[14] Holnicki-Szulc J., Bielecki T. (2000) Virtual Distortion Method - Review. Applications to Fast Redesign and Sensitivity Analysis, *International Journal of Nonlinear Science and Numerical Simulation*, 1, pp. 71-98

[15] Putresza J. T., Kolakowski P. (2001) Sensitivity Analysis of Frame Structures - Virtual Distortion Method Approach, *International Journal for Numerical Methods in Engineering*, vol. 50, pp. 1307-1329

[16] Vetterling W. T., Teukolsky S. A., Press W. H., Flannery B. P. (1992) *Numerical Recipes in Fortran* (2nd edition), Cambridge University Press, New York, U.S.A.

[17] Kolakowski P., Wiacek D. (2002) DAMIS – DAMage Identification for Static problems, Fortran code, IFTR, Warsaw