

## SOFTWARE TOOLS FOR STRUCTURAL DAMAGE IDENTIFICATION

Tomasz G. Zieliński, Andrzej Świercz

Institute of Fundamental Technological Research  
ul. Świętokrzyska 21, 00-049 Warsaw, Poland  
e-mails: [tzielins@ippt.gov.pl](mailto:tzielins@ippt.gov.pl), [aswiercz@ippt.gov.pl](mailto:aswiercz@ippt.gov.pl)  
web page: <http://smart.ippt.gov.pl/>

**Key words:** Virtual Distortion Method, structural remodelling, inverse dynamic analysis, damage identification.

**Abstract.** *The principal purpose of this paper is to present recently implemented in the Java programming language software tools for the so-called Virtual Distortion Method (VDM), which is the method used to deal with some problems of structural analysis, control and design. The developed package of software includes also tools for the recently developed Impulse Virtual Distortion Method (IVDM), which is dedicated for structures under dynamic load. This version of the method was used to develop a new approach to the damage identification problem based on the analysis of perturbation of elastic wave propagation. The paper describes briefly the foundations of VDM and IVDM, then some general features of the software are presented, and finally a simple numerical example to test some of its algorithms is given. The example regards the problem of damage identification.*

## 1 INTRODUCTION

The paper presents software tools for the so-called *Virtual Distortion Method* (VDM). In his monograph [1] Holnicki-Szulc comprehensively described this method and its application to structural analysis, design and control of static structures. The developed package of software includes also tools for recently developed *Impulse Virtual Distortion Method* (IVDM), which is the method dedicated for structures under dynamic load. Preliminary descriptions of this approach together with the idea of application to the damage identification problem based on perturbation of elastic wave propagation were given by Holnicki-Szulc and Zieliński [2]. Thorough and complete description of the method gave Zieliński in [3]. In this work some new aspects of the classical VDM together with the description of the object-oriented implementation of distortion methods were given. This paper briefly discusses some basics of this subject.

## 2 VIRTUAL DISTORTION METHOD (VDM)

### 2.1 Some basic notions used by the VDM

The following notions play a very substantial role in the Virtual Distortion Method (here we give their brief definitions):

- the *virtual distortion* – the initial strain introduced in a structural element (its effect is analogous to the result of non-homogenous heating or geometric imperfection);
- the *unit-distortion* – the virtual distortion that would cause a unit-strain (of some kind) in an unconstrained element;
- the *compensating load* – the self-equilibrated load applied to the nodes of an element equivalent to the unit-distortion effect;
- the *influence matrix* – generalized strains (or, in general case, arbitral static linear responses of the structure) obtained for unit-distortions imposed successively on some of structural elements.

It is important to remark here that the virtual distortions,  $\hat{\varepsilon}_i$ , are to be used to model modifications of design parameters,  $p_s$ , that involves modification of structural stiffness,  $k_i$ .

### 2.2 Influence matrices

We shall see that the most fundamental for VDM computations is so-called *influence matrix*. It groups static responses obtained for unit-distortions imposed successively on some of the structural elements. In practice, every column of the matrix is calculated for the adequate compensating load.

Two kinds of the influence matrix were distinguished by Zieliński [3]. The first one is the *general influence matrix*,  $\check{D}_{\alpha i}$ . The type of response grouped in this matrix is

quite arbitral, i.e., depends on what is “in our interest”. The only requirement is that it should be linearly dependent on  $\hat{\varepsilon}_i$  (in practice, the response is just a linear combination of the generalized displacements of the finite element model degrees of freedom). Thus, having this matrix computed and knowing the distortions (which, for instance, model some structural modifications), we are able to calculate quickly the updated result (without any modification of the original structure):

$$f_\alpha = \overset{L}{f}_\alpha + \sum_i \check{D}_{\alpha i} \hat{\varepsilon}_i. \quad (1)$$

Here,  $\overset{L}{f}_\alpha$  is the original response to the load (i.e., linear term), whilst the second term (where the general influence matrix is used) describes the effect of structural modifications. This second term is non-linear since distortions non-linearly depend on modifications.

A certain particular case of the general influence matrix will be the second distinguished type – we shall call it the *strain influence matrix*,  $D_{ij}$ . The responses grouped in this matrix are strains obtained in some distortion locations for successive unit-distortions imposed in these locations. Therefore, this is a square matrix, and all elements on its diagonal (i.e.,  $D_{ii}$ ) are in the interval  $[0, 1]$ . Knowing this matrix we may calculate the general stresses and strains in some members of the structure:

$$S_i = k_i \overset{L}{\varepsilon}_i + k_i \sum_j (D_{ij} - \delta_{ij}) \hat{\varepsilon}_j, \quad (2)$$

$$\varepsilon_i = \overset{L}{\varepsilon}_i + \sum_j D_{ij} \hat{\varepsilon}_j. \quad (3)$$

Here, the first terms  $\overset{L}{S}_i$  and  $\overset{L}{\varepsilon}_i$  are the linear response to the load, whereas the second terms of both expressions are the residual responses for the imposed distortions that may model modifications and thus they are non-linear. However, the used above strain influence matrix tends to play more substantial role, since (as it will be shown below) it is used to determine these distortions.

### 2.3 Distortion states in finite elements on the example of the well-known 3D-beam finite element

Let us consider the well-known finite element of three-dimensional Bernoulli beam. Solving the eigenvalue problem for the stiffness matrix,  $\mathbf{K}_{(12 \times 12)}^{(e)}$ , of this element provide us with twelve eigenvectors and corresponding twelve eigenvalues. Six of the eigenvalues are equal zero, and their eigenvectors describe the six rigid motions of 3D-beam element: three translations and three rotations. Thus, the remaining six eigenstates specify the complete *orthogonal basis of deformations*. These deformations should be assumed as the distortion states for this element. The form of these deformations and the compensating loads equivalent to the unit-distortions are presented in Fig. 1. The analogous approach

of defining unit-distortion states as orthogonal, non-rigid eigenstates is proposed for other finite elements.

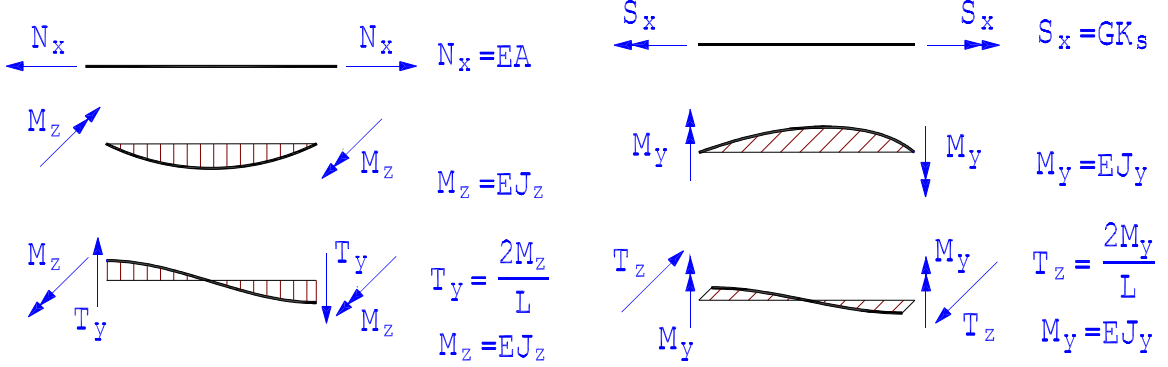


Figure 1: Orthogonal deformations (eigenstates) of the 3D-beam element assumed as the states of unit-distortions, and their compensating loads.

## 2.4 Distortion-based modelling of modifications of structures under static load

Let  $\mathbf{p} = [p_s]$  be the *vector of structural parameters* and  $\hat{\mathbf{p}} = [\hat{p}_s]$  – its modification. We demand that these parameters affect stiffness parameters, i.e.:  $k_i = k_i(\mathbf{p})$ , so after the modification we have:  $\hat{k}_i = k_i(\hat{\mathbf{p}})$ . We define the *parameters of stiffness modification* as:  $\mu_i = \hat{k}_i/k_i$ . Introducing the *vector of structural modification*:  $\boldsymbol{\lambda} = [\lambda_s]$ , where  $\lambda_s = \hat{p}_s/p_s$ , we may write that  $\mu_i = \mu_i(\boldsymbol{\lambda})$ . Now, let us consider the structure modified in some locations and the original structure but with some distortions imposed in these locations. We postulate that:

*the structure modelled by distortions and the modified structure are identical in the sense of equality of their fields of general strains and stresses.*

General stresses and strains of the structure modelled by distortions are expressed by Eqs. (2) and (3). Thus, we substitute these equations to the constitutive relationship of the modified structure (under static load):  $S_i = \hat{k}_i \varepsilon_i$ , and after division by  $k_i \neq 0$  and having used the definition for  $\mu_i$ , we obtain:

$$\sum_j A_{ij} \hat{\varepsilon}_j = (1 - \mu_i(\boldsymbol{\lambda})) \varepsilon_i \quad (4a)$$

where

$$A_{ij} = \delta_{ij} - (1 - \mu_i(\boldsymbol{\lambda})) D_{ij}. \quad (4b)$$

The above system of equations (4) allows us to compute the distortions that model the modifications.

## 2.5 VDM applied to the sensitivity analysis of structures under static load

We want to determine how the specified response  $f_\alpha$  is sensitive to structural modifications. This means that we want to be able to calculate the following gradient:  $\frac{\partial f_\alpha}{\partial \lambda_s}(\boldsymbol{\lambda})$ . From Eq.(4) it is evident that distortions  $\hat{\varepsilon}_i$  depend on  $\boldsymbol{\lambda}$ . Terms  $f_\alpha^L$  are the responses of the original structure (i.e., without any modifications) to the predetermined load and thus they don't depend on  $\boldsymbol{\lambda}$ . Therefore, we have that

$$\frac{\partial f_\alpha}{\partial \lambda_s}(\boldsymbol{\lambda}) = \sum_i \check{D}_{\alpha i} \frac{\partial \hat{\varepsilon}_i}{\partial \lambda_s}(\boldsymbol{\lambda}). \quad (5)$$

The above relationship means that to compute the stiffness sensitivity we need to calculate the gradient of distortions,  $\frac{\partial \hat{\varepsilon}_i}{\partial \lambda_s}(\boldsymbol{\lambda})$ . To find this gradient we differentiate (4) with respect to  $\lambda_s$ , and as result we get the following linear system of equations:

$$\sum_j A_{ij} \frac{\partial \hat{\varepsilon}_j}{\partial \lambda_s} = -\frac{\partial \mu_i(\boldsymbol{\lambda})}{\partial \lambda_s} \left[ \hat{\varepsilon}_i^L + \sum_j D_{ij} \hat{\varepsilon}_j(\boldsymbol{\lambda}) \right]. \quad (6)$$

Here we should notice that before solving the above system of equations, we need to determine the distortions  $\hat{\varepsilon}_i$  that model the modifications  $\boldsymbol{\lambda}$ . To this end we must solve the system (4). However, let us notice that the governing matrices for the both systems, (4) and (6), are the same and expressed by Eq. (4b).

## 3 IMPULSE VIRTUAL DISTORTION METHOD (IVDM)

### 3.1 Impulse of distortion and impulse influence matrix

For dynamic problems we need to introduce the time factor into the VDM. Thus, we assume that the virtual distortions depend on time. Therefore, the influence matrix is also time-dependent, and so now it will be three-dimensional matrix. The Impulse VDM is based on the so-called *method of impulse response function*. Thus, for IVDM we have the so-called *impulse influence matrix*, since its every column is computed (using Newmark's method) for a unit-distortion applied as Dirac-like *impulse* in the initial instant,  $\tau = 0$ . In practice this impulse loads we perform in the initial conditions of Newmark's integrations of the homogenous equations of motion. This distortion-based approach proposed for structures under predetermined dynamic load allows to model the modifications of parameters that involve change of stiffness (modelling of mass modification is not yet implemented).

### 3.2 Structural remodelling of structures under determined dynamic load

Thanks to the impulse influence matrices,  $\check{D}_{\alpha i}(t)$  and  $D_{ij}(t)$ , knowing distortion functions  $\hat{\varepsilon}_i(t)$  we may compute, respectively: the actual dynamic response

$$f_\alpha(t) = f_\alpha^L(t) + \sum_{\tau=0}^t \sum_i \check{D}_{\alpha i}(t - \tau) \hat{\varepsilon}_i(t), \quad (7)$$

and the time-varying strain functions in distortion locations

$$\varepsilon_i(t) = \overset{L}{\varepsilon}_i(t) + \sum_{\tau=0}^t \sum_j D_{ij}(t-\tau) \hat{\varepsilon}_j(\tau). \quad (8)$$

However, it should be emphasized here that the *strain impulse influence matrix*,  $D_{ij}(t)$ , is mostly needed when we determine the distortion functions which model structural modifications  $\boldsymbol{\lambda}$  that involve change of stiffness. To this end we solve, for every successive instant  $t$ , the following system of equations:

$$\sum_j A_{ij}^0 \hat{\varepsilon}_j(t) = \begin{cases} (1 - \mu_i(\boldsymbol{\lambda})) \overset{L}{\varepsilon}_i(0) & \text{for } t = 0, \\ (1 - \mu_i(\boldsymbol{\lambda})) \left[ \overset{L}{\varepsilon}_i(t) + \sum_{\tau=0}^{t-1} \sum_j D_{ij}(t-\tau) \hat{\varepsilon}_j(\tau) \right] & \text{for } t > 0, \end{cases} \quad (9a)$$

where, fortunately, the governing matrix does not depend on  $t$ :

$$A_{ij}^0 = \delta_{ij} - (1 - \mu_i(\boldsymbol{\lambda})) D_{ij}(0), \quad (9b)$$

and thus is identical for all the systems. Please, notice that this matrix is in a way similar with the matrix (4b) used for determining static distortions.

### 3.3 Sensitivity analysis of structures under dynamic load

The stiffness sensitivity of dynamic response to modification of structural parameters consists in determining the following gradient:

$$\frac{\partial f_\alpha}{\partial \lambda_s}(t, \boldsymbol{\lambda}) = \sum_{\tau=0}^t \sum_i \check{D}_{\alpha i}(t-\tau) \frac{\partial \hat{\varepsilon}_i}{\partial \lambda_s}(t, \boldsymbol{\lambda}), \quad (10)$$

for which we need to compute the distortion gradient by solving, successively, for every instant  $t$ , the following systems of equations (for every  $s$ ):

$$\sum_j A_{ij}^0 \frac{\partial \hat{\varepsilon}_j}{\partial \lambda_s}(t) = \begin{cases} -\frac{\partial \mu_i(\boldsymbol{\lambda})}{\partial \lambda_s} \varepsilon_i(0, \boldsymbol{\lambda}) & \text{for } t = 0, \\ -\frac{\partial \mu_i(\boldsymbol{\lambda})}{\partial \lambda_s} \varepsilon_i(t, \boldsymbol{\lambda}) + (1 - \mu_i(\boldsymbol{\lambda})) \sum_{\tau=0}^{t-1} \sum_j D_{ij}(t-\tau) \frac{\partial \hat{\varepsilon}_j(\tau, \boldsymbol{\lambda})}{\partial \lambda_s} & \text{for } t > 0. \end{cases} \quad (11)$$

The right-hand sides of these systems depend on the strain functions (8), and thus they depend on the distortions that are to be first determined from (9). Fortunately, the governing matrix for all the systems (11) is the same and equal (9b), so it is identical with the governing matrix for the system of equations (9a). Thus, we need to perform the LU-decomposition only once.

## 4 FEM+VDM = SOFTWARE TOOLS FOR THE VIRTUAL DISTORTION METHODS

### 4.1 Main features

Algorithms based on the approach briefly presented in the preceding sections were thoroughly elaborated and described in [3]. This description was the base for the effective implementation in *Java* – the leading object-oriented programming language. The implementation was strongly focused on the numerical effectiveness; therefore, very ineffective (though handy) generic Java multidimensional arrays were generally not used (two- and three-dimensional arrays were implemented as objects with data stored in an inner one-dimensional array, *à la* Fortran).

We have shown that, generally speaking, both of the Virtual Distortion Methods are used to model some modifications in a structure. These modifications may be caused by some structural defects, by development of zones of plasticity, or cracks, etc. The VDM approach applies the *virtual distortions* to model the modifications and therefore it *does not* modify the model of the structure (i.e. the stiffness matrix remains unchanged). However, the VDM needs at the beginning a Finite Element model of the original (i.e., unmodified) structure. Thus, the important feature of the VDM software is to cooperate with some well-known commercial CAD/CAE systems for Finite Element Analysis like ADINA, ABAQUS, or ANSYS, etc. The user is encouraged to make use of the finite element model prepared under one of those systems (so far some interface procedures are ready to cooperate with the ADINA product, but interfaces compatible with the ABAQUS and ANSYS are to be added soon).

Moreover, the VDM software cooperates with the original and entirely independent Finite Element System for frame and truss structures written in Java (the program is completely object-oriented and it may be easily expanded with some other finite elements).

We have remarked that the presented FEM+VDM software is written in Java. Therefore, the important feature is that it can be used directly from within the well-known and widely-used MATLAB environment. In fact, some procedures and functions covering the visualization aspect were written in MATLAB.

### 4.2 Structure of the program

Figure 2 presents the structure of the FEM+VDM software. We can see that the software consists of three parts:

1. **Java code** together with thorough and handy (PDF, HTML) description in the API standard (Advanced Programmer's Interface). The code consists mainly of three packages:
  - *Basic Linear Algebra System* (BLAS) – classes with useful methods needed by any finite element system (includes, for example, classes for symmetric dense matrix, sparse profile matrix, etc.);

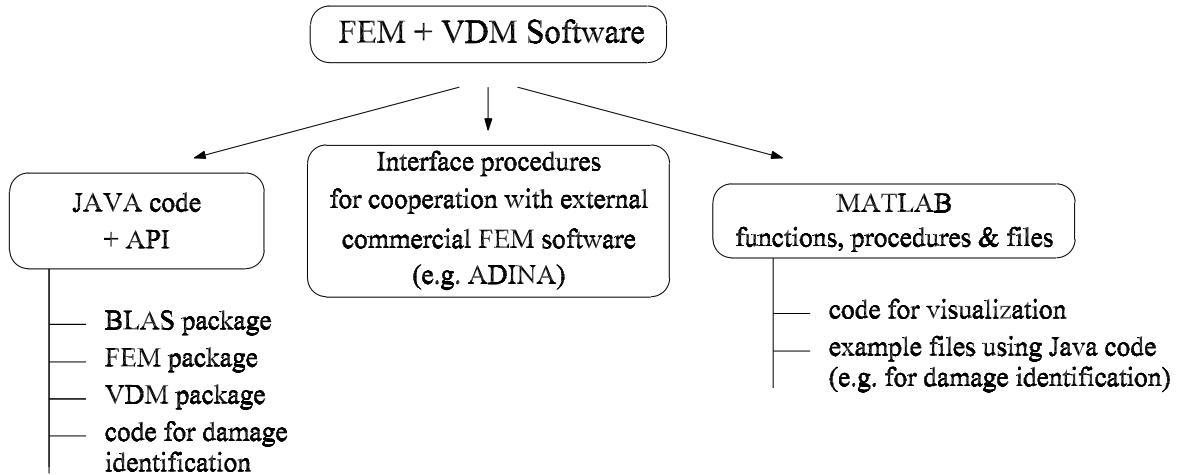


Figure 2: The structure of the FEM+VDM software.

- *Finite Element Method* (FEM) – the finite element system for statics and dynamics of small deformations (consists of several subpackages);
- *Virtual Distortion Method* (VDM) – the package of VDM and IVDM algorithms for structural modelling and sensitivity analysis.

There are some other minor utility packages as well as some application code (e.g., for *damage identification*).

2. *Package of interface procedures* to cooperate with the external commercial software for Finite Element Method (e.g., ADINA).
3. *Package of MATLAB functions, procedures and files* – amongst other things for the visualization purposes and for damage identification.

### 4.3 The way of proceeding when using the FEM+VDM software

The most important and extensive part of the software is the *Java code*. As we have mentioned it comprises the package for Finite Element Method, which (for the time being) allows to build frame and truss structures. Therefore, in case of such structures we don't need any of the external, commercial FEM software. However, it often happens that our structure is more complicated or we just want to use the finite element model already prepared and tested under one of the commercial systems. Thus, the general proceeding should be as follows (see also Fig. 3):

- Build a finite element model of the structure. To this end some of the well-known commercial software for Finite Element Modelling may be used. An important fact is that often such a model already exists.



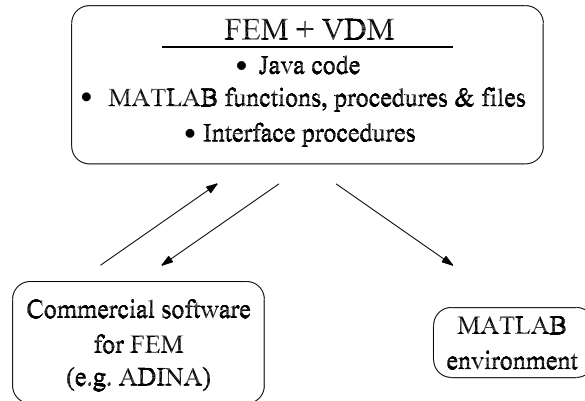


Figure 3: FEM+VDM – cooperation with other programs.

- Decide what parts of the structure are to be considered for virtual modifications. This decision depends strongly on the problem we are dealing with. It may mean, for example:
  - deciding what members of the structure are to be taken into account in the problem of structural remodelling, or
  - which of the structural elements should be considered as possible defect localizations in the damage identification problem, etc.

In this decision the very important for further calculations *set of distortion localizations* is determined. Sometimes, if need, the mesh of the finite element model should be improved to better respond to the examined problem, i.e., to have the optimal choice of the distortion localizations.

- Specify what is the needed structural response (e.g., some displacements or strains, or their linear combinations, readings of sensors, etc.).
- By using interface procedures in regard to the decisions taken in the preceding two points calculate:
  - the strain and general influence matrices (impulse or static),
  - the specified response (e.g., numerical simulations of the readings of sensors) of the structure for the given loading (dynamic or static).
- Now, the algorithms of the VDM software package can be used to compute modelling distortions, updated responses, structural sensitivities for some values of modification parameters.
- Results that are being obtained by the distortion algorithms can be used (controlled iteratively) by the code in Java or MATLAB environment.

## 5 APPLICATION OF IVDM TO THE DAMAGE IDENTIFICATION PROBLEM

### 5.1 Simple numerical experiment

A numerical test for the IVDM sensitivity algorithm was performed regarding the problem of damage identification in the simple truss cantilever shown in Fig. 4. The identification base on the analysis of elastic wave propagation. All elements of the cantilever have the same material and section properties. It was assumed that the truss is excited by using activators at the free tip of the cantilever (elements Nos. 39 and 40), generating sine-wave excitations of identical amplitude and opposite phases. Elements Nos. 21 and 22 have sensors ready to read any change in the longitudinal strain of these bars (Fig. 4).

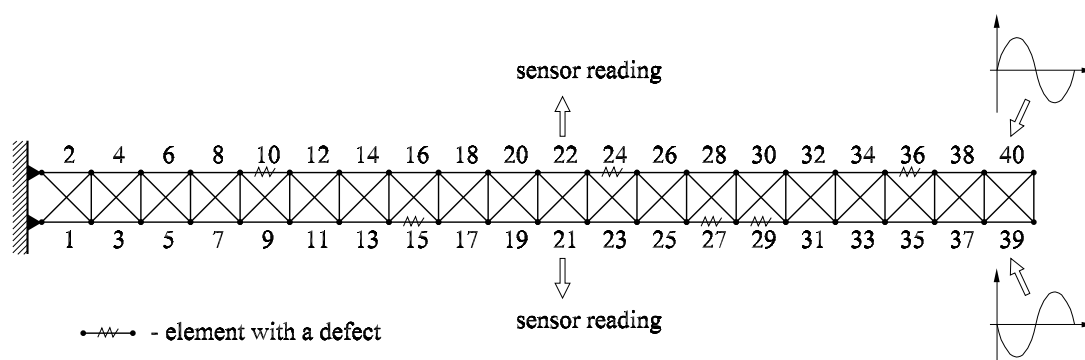


Figure 4: Truss cantilever.

For the assumed excitation signal the responses (i.e., sensor readings, see Fig. 5) were calculated numerically, first, for the case of the original structure (i.e., undamaged), and then for the truss with defects – it was assumed that several bars of the top and bottom flanges have defects reducing their structural stiffness. The defects were considered as thin cracks that reduce the effective area of cross-section but do not significantly affect the mass of element. They were modelled by assigning to every of the defected elements material of adequately reduced Young modulus. Values of defect intensities assumed for the damaged elements are given in Table 1. Localization of the defects is also marked in Fig. 4, while Fig. 5 presents the driven responses “read” from both sensors before and after the damage occurred.

Table 1: Assumed defects intensities in some elements of the truss.

element No.:	$e =$	10	15	24	27	29	36
change of stiffness:	$\mu_{EA}^{(e)} = \frac{\hat{E}^{(e)}}{E^{(e)}} =$	0.8	0.7	0.6	0.6	0.7	0.5
defect intensity:	$1 - \mu_{EA}^{(e)} =$	20%	30%	40%	40%	30%	50%

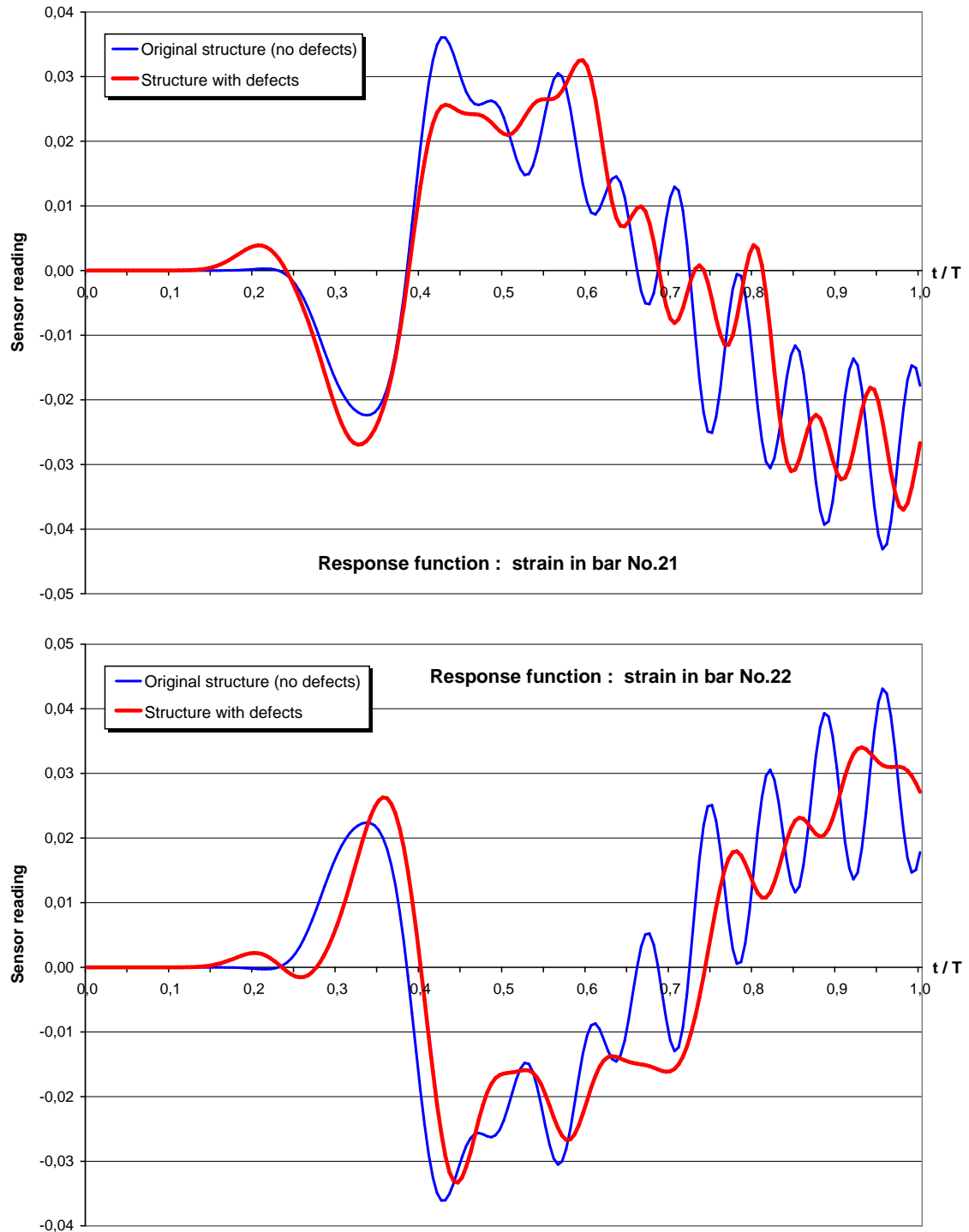


Figure 5: Forced responses of the truss structure with and without defects (i.e., “sensor readings” from elements Nos. 21 and 22).

## 5.2 Results of the identification analysis

In the examined damage identification problem we assumed that the defects are to be sought only in the elements of the both flanges of the truss cantilever but with the exception of the two elements with activators. Thus, we were seeking for defects only in the most probable flange elements Nos. 1 to 38 (see Fig. 4). The identification process was based on gradient approach where the objective function was defined as the squared difference between the responses obtained, respectively, for the original and damaged structure. For gradient calculations the IVDM algorithm for stiffness sensitivity of dynamically loaded structure was used. Actually, for the sake of comparison three independent damage identification processes were performed:

1. the identification using “readings” from the sensor situated in the element No. 21,
2. the identification using “readings” from the sensor situated in the element No. 22,
3. the identification using “readings” from the both sensors simultaneously.

In every of the identification processes calculations were performed in 40 iterations. The results are cumulatively presented in Fig. 6. The analysis of those graphs leads to the following observations and straightforward conclusions:

- Better results were obtained in the identification process using the sensor from the element No. 22 than in the process with the sensor from the element No. 21. However, because of the symmetry of the structure, localization of sensors and excitation, it is obvious that contrary situation shall happen for some other defect distributions.
- Obviously, the best results are from the identification making use of the readings obtained from both sensors, though localization of defects found using only one sensor (in the element No. 22) is sufficiently accurate (especially when considering the defects in the top flange).
- Defect identification (even using both sensors) cannot be fully accurate – the intensities of the identified defects are usually lower than the accurate ones, while some neighbouring elements exhibit false defects (though usually of comparatively small intensities). Thus, the identified defects tend to be fuzzy. This happens even for such a simple numerical experiment. Obviously, these small errors are not very disturbing but in more complicated tasks much important errors shall happen. This is because the solution of the identification problem is, generally speaking, not unique; some systematic errors will play an important role as well (for example, an imperfect FE model, an inappropriate excitation signal, etc.).
- Small defects that cause little loss of stiffness might be difficult to identify especially in the presence of other bigger defects. Moreover, they can be hardly distinguished from the mentioned above fuzziness of identification results.

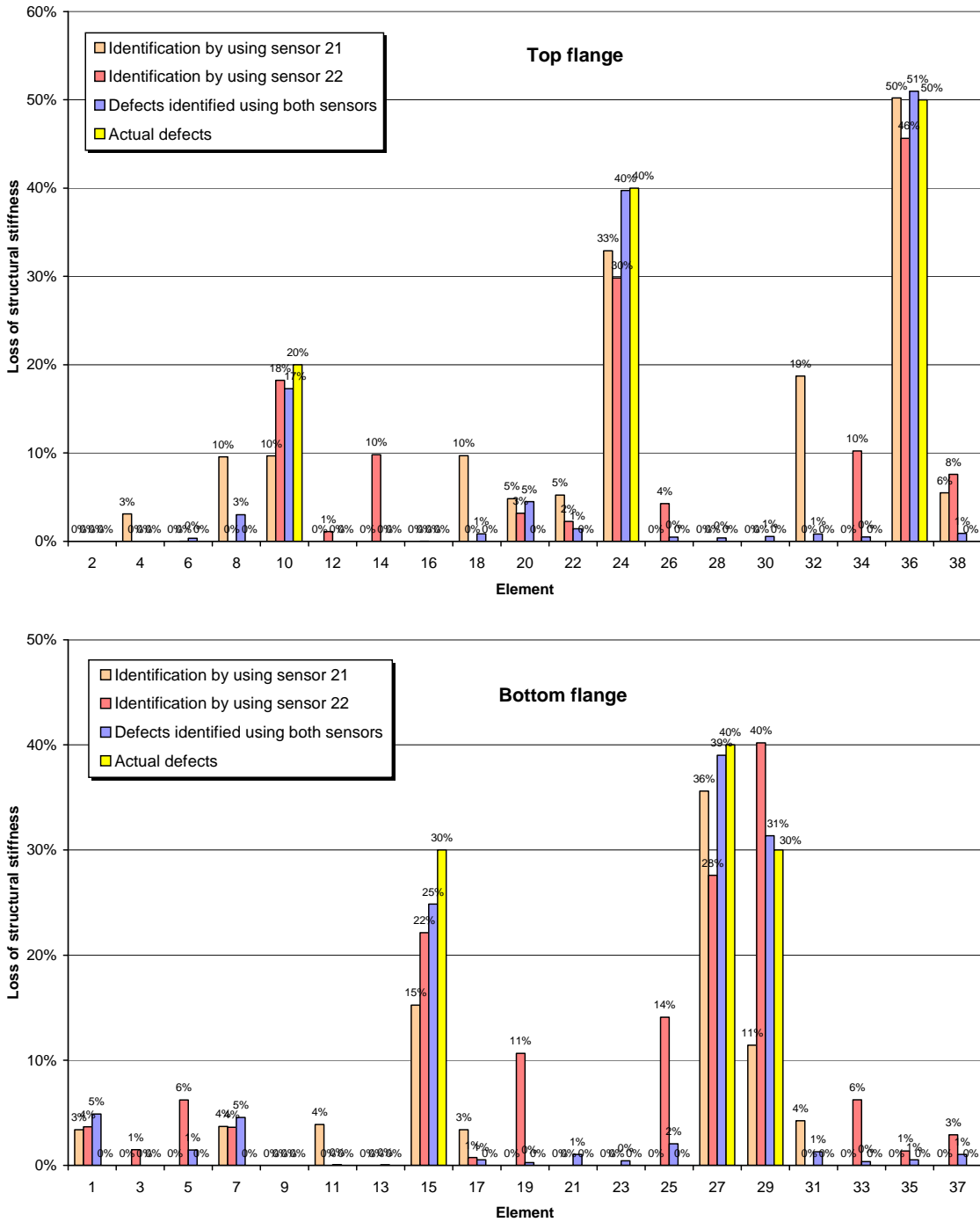


Figure 6: Defects identified in the elements of both flanges of the truss cantilever. In every position corresponding to an element from the top or bottom flange of the truss cantilever defect intensities identified independently in the three processes of identification are shown. The “actual” values of numerically-modelled defects are presented as well.

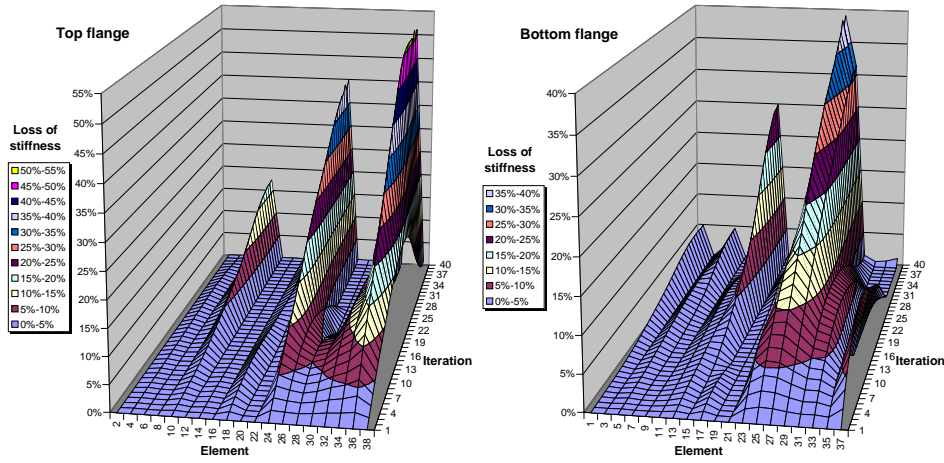


Figure 7: Histogram of the defect identification process (the identification using both sensors).

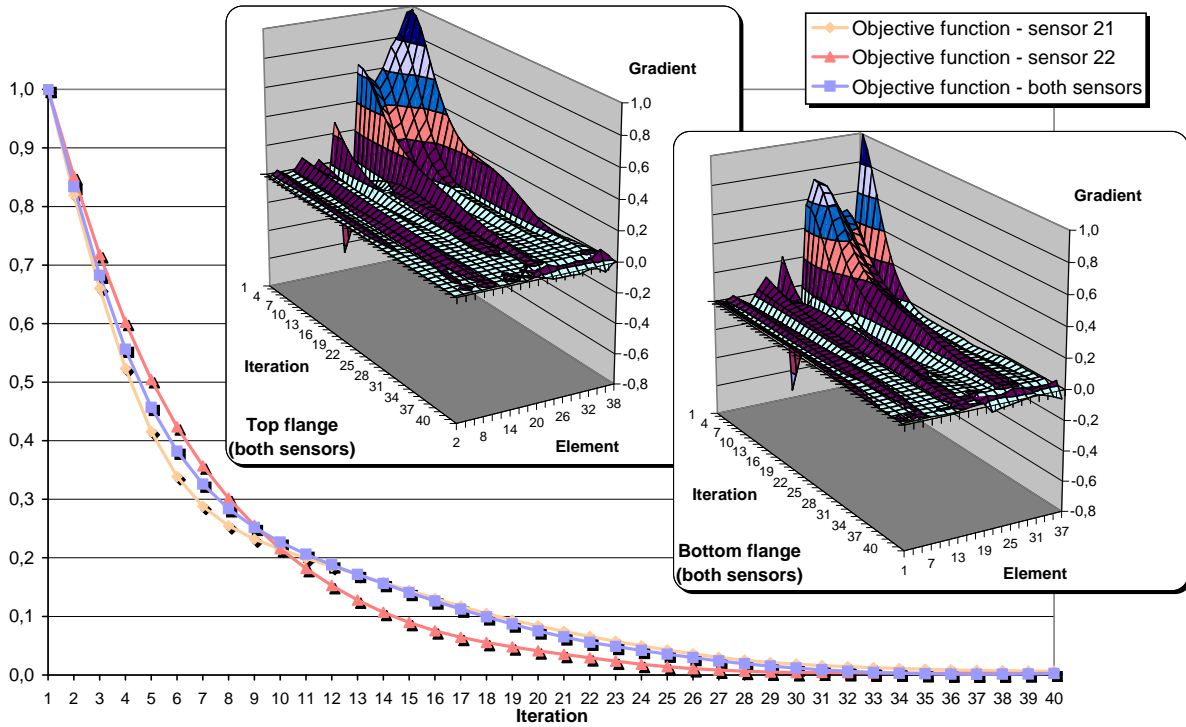


Figure 8: Objective function and its gradient computed in successive iterations of the damage identification process. Here, three objective functions for the three independent identification processes are presented, namely: the process using the sensor from the element No.21, the process using the sensor from the element No.22, and the process using the both sensors simultaneously. The presented graphs of the gradient of objective function was obtained from the process where both sensors were used. Values in the graphs are scaled by the corresponding maximal value obtained in the starting iteration.

Figure 7 presents a histogram of the defect identification process where readings from both sensors were used, while Fig. 8 shows normalized plots of the corresponding objective functions obtained for all three identification processes. Figure 8 depicts also the gradient components of objective function obtained in successive iterations for the identification process with two sensors.

## ACKNOWLEDGEMENT

The authors would like to acknowledge the financial support through the 5FP EU Project Research Training Networks *SMART SYSTEMS* HPRN-CT-2002-00284, through the grant No. KBN 5T07A05222 funded by the State Committee for Scientific Research in Poland, and through 5FP Research Project: G1RD-2001-00659 (2002-2005): *Smart Structural Diagnostics using Piezo-Generated Elastic Waves – PIEZODIAGNOSTICS*, supported by the European Commission. Partners: CEGELEC, TWI, WSATKINS, CEDRAT, CIMNE, CDRiA, IFTR, ECL, ALSTOM.

## REFERENCES

- [1] J. Holnicki-Szulc. *Virtual Distortion Method*. Lecture Notes in Engineering 65, edited by C.A. Brebbia and S.A. Orszag. Springer-Verlag, 1991.
- [2] J. Holnicki-Szulc and T.G. Zieliński. Damage identification method based on analysis of perturbation of elastic waves propagation. In: J. Holnicki-Szulc (Editor). *Lecture Notes 1: Structural Control and Health Monitoring. Proc. of Advanced Course SMART'01, Warsaw, 22-25 May 2001*, pp.449-468.
- [3] T.G. Zieliński. *Impulse Virtual Distortion Method with Application to Modelling and Identification of Structural Defects*. Ph.D. thesis (in Polish). Warsaw, 2003.