

Techniques to check properties of complex design models

Edward Z. Kryksunov*, Anatoly V. Perelmuter

SCAD Group, 13, Chokolovsky blrd., room 508, 03186, Kiev, Ukraine

e-mail: aperel@i.com.ua

Vladimir I. Slivker

Giprostroimost Co., Saint-Petersburg, Russia

e-mail: slivker@gpsm.ru

Abstract

The report discusses issues of verification of data specified for finite-element analysis of complex systems where large arrays of information have to be entered and results of the analysis have to be checked and reviewed. The emphasis of the report is on the development of special-purpose software tools to facilitate this complicated process. Examples are given of such tools included in the SCAD finite element analysis/design software.

Keywords: software; design model verification; graphical control

1. Introduction

The analysis of the behavior of a complex multi-element system requires more than just a problem of computational mechanics to be solved. Also, there are ergonomic issues that a good software system must find a way to resolve in order to work properly. This point includes overcoming a contradiction between gigantic computational capabilities of software and a limited ability of humans to perceive and review great amounts of input and output information.

The problem is especially serious with the computational analysis of construction objects where both the dimensionality of problems and number of parameters to be specified are large. An example shown in Fig. 1 is pretty typical and far from ultimate. That design model includes over 23 thousand nodes, 35.5 thousand elements of various kinds (bars, plates, shells, elastic links) which have over one hundred different sets of rigidity properties (also referred to as rigidity types). This analysis must be performed under 15 to 20 different load patterns, each one containing hundreds of components of nodal and distributed loads.

The total amount of output data in a problem of this kind can be up to thousand pages of tabular text.

It is commonly known that large-scale problems make the probability for an error in source data dramatically high. Researches in engineering psychology state an approximately power (exponential) dependence of a human error on the amount of information to be processed by the human. The problem yet more complicated is how to review the large amount of output information. In this output data the analyst can hardly find data of his interest manually, such as ultimate values, irregularities, abnormalities etc.

Any contemporary computational software operates fairly inhomogeneous data such as those intended to describe properties of elements, nodes, loads and actions etc. When solving large dimensionality problems, one can hardly consider all this information at the same time. Involving a standard approach of fragment extraction is not always convenient for this purpose. It is particularly inconvenient when one tries to track a relationship between a certain factor and a system as a

whole, for example, how elements of certain rigidity are arranged within a structural system.



Figure 1: An example

To facilitate the analysis of input information, special tools are used. These are usually built in preprocessing and post-processing units of CAD software. Below there is a description of a set of such tools implemented by the SCAD software [1, 2] that is pretty typical for the contemporary approach to the problem.

2. Using color indication

In order to facilitate the usage of the system that includes hundreds of functions and control actions, SCAD suggests highly unified working conventions and uses colors extensively to mark objects with similar properties. Nearly all properties of elements and nodes can be represented as color scales and markers. Those include rigidities and types of elements, constraint directions, overall loads and numbers of loads. Color is used to highlight groups of nodes or elements, merged displacements, structural parts etc.

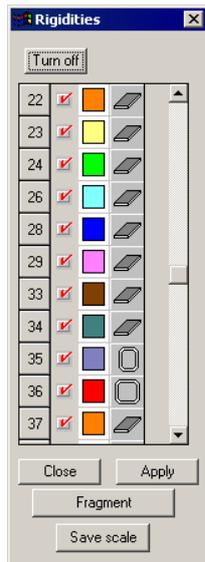


Figure 2: A color scale

The principal tool for color handling is a color scale (Fig. 2) which does more than purely informative job and therefore includes a variety of controls. This tool can be used to decompose a model by object properties specified for the scale, such as bed response factors. Also, it can be used to color a specific group of objects in the full design model.

Controlling color scales enables one to vary the color of the view. When computational results are being reviewed, one can change ranges of iso-fields of quantities in question: build those with a non-uniform scale, display only min-max values, or display values from a specific range only.

Many years of experience with SCAD has shown that the color separation of properties is a convenient and self-obvious way of data representation that enables the user to detect any abnormalities easily. The capability of controlling color scales available in the system provides a great flexibility of this tool.

3. Filters and markers

Displaying of design model parameters can be controlled via a system of filters and markers (Fig. 3) based on visualization of given data sets and removal of information of

no interest at the moment from the screen. For example, information about elements of certain types or certain spatial location/attitude can be obtained either separately or in any thinkable combinations.



Figure 3: Filters and markers

Unlike markers that just initialize displaying some needed data, filters have individual setup tools capable of narrowing the range of displayed data down to a needed level. For example, only constraints in a given direction or only loads of given types and values can be visualized. In many cases the user can choose the method of data displaying such as alphanumerical or color scale. Also, results of several filters can be combined together. Fig. 4a shows an example of a simultaneous operation of filters intended for removal of invisible lines and color indication of elements' rigidity properties. Note that a realistic display of bar element cross-sections allows both to verify the section correctness and to check their attitude in the design model (the orientation of principal axes).

Similar ideas of searching for deviations from a general law form the basis for a check of rigidity properties by a rigidity spectrum (Fig. 4b) where all element rigidities present in the analysis are shown graphically.

Loads make up a most popular source of errors. Fig. 5 demonstrates how it is possible to extract one load of a specific type, direction and value (Fig. 5b) from all loads specified for a current loading (Fig. 5a). This functionality helps correct erroneously specified values of a particular load on-the-fly. Also, a color map of overall values of distributed loads helps one easily detect elements or nodes with erroneous data or missed loads (Fig. 6) which are usually quite clear against correct values.

Color maps can be used also to check directions of local finite element axes, particularly normal directions of plate and shell elements (Fig. 7). Another application of those is the selection of fine details such as coincident nodes or elements in the model which are hardly perceivable by other means available in the software.

In similar ways a lot of other parameters of design models can be monitored and verified. Together with logical analysis, these tools can be used to easily detect and amend inaccuracies in design models at the stage of their creation.

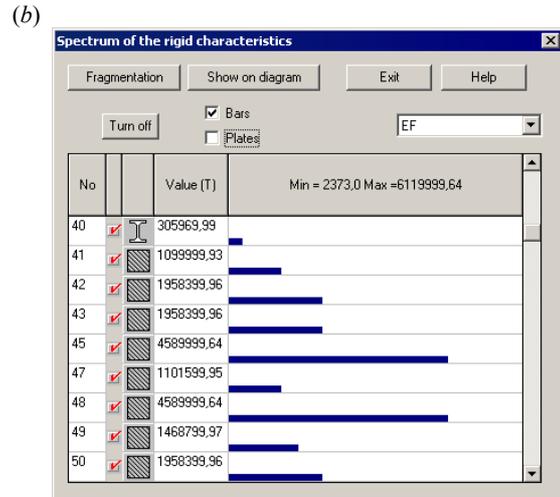
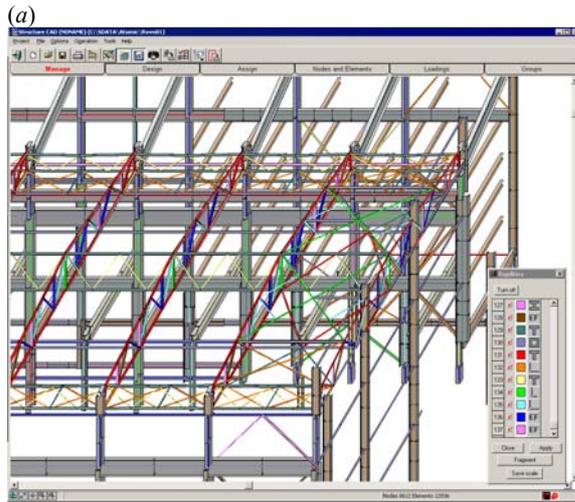


Figure 4: An example of rigidity visualization: (a) simultaneous display of section shapes and a color scale; (b) rigidity spectrum.

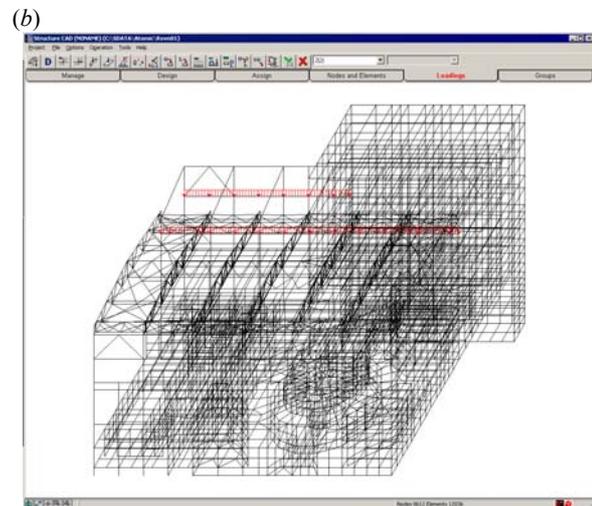
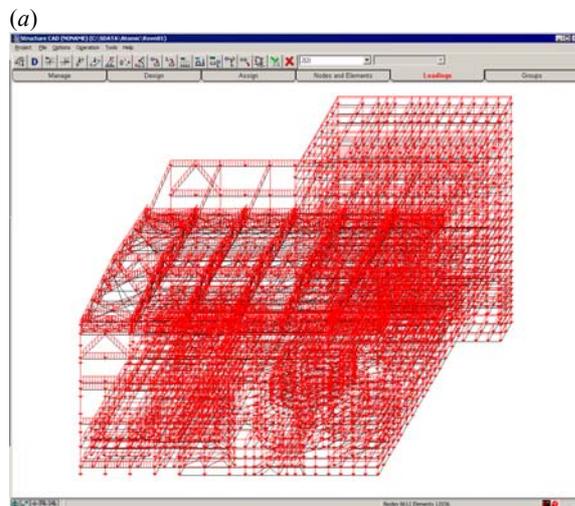


Figure 5: An example of search for a particular load: (a) all loads displayed; (b) the result of the search.

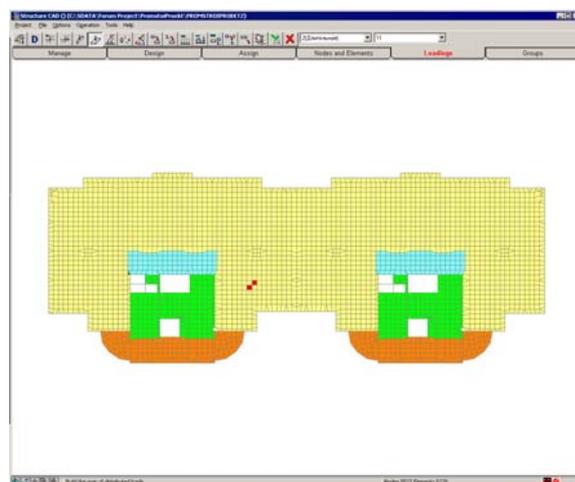


Figure 6: A color map of load values (the arrow indicates the error)

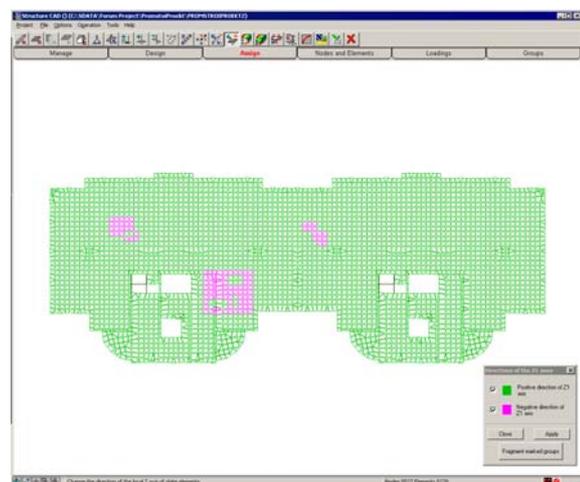


Figure 7: A color map of normal directions to surfaces of two-dimensional elements

4. Preliminary verification and verification in the course of solution

Almost any software system can organize checks of correctness of its source data. Most often, the tool that does this job is a special preprocessor. What is really important is the extent of smartness of such preprocessor and settings it uses for its operation.

For example, if one needs to check whether coordinates of nodes coincide, there is a very important setting for this check — accuracy of the coincidence — especially in cases when the analyst commands to remove coincident nodes because nodes in the same point of space appear most often due to an error in source data. Hence the wish to make this check as severe as possible by decreasing the allowable offset of coordinates. On the other hand, if this accuracy setting is made too little, there can be a problem when “assembling” the system out of different subsystems in which node coordinates cannot have exactly the same values.

The function of detection of coincident elements is just as useful. Most often, coincident elements are erroneous, though both coincident nodes and coincident elements can also be used consciously. Even in this case the capability of detecting those is of great help.

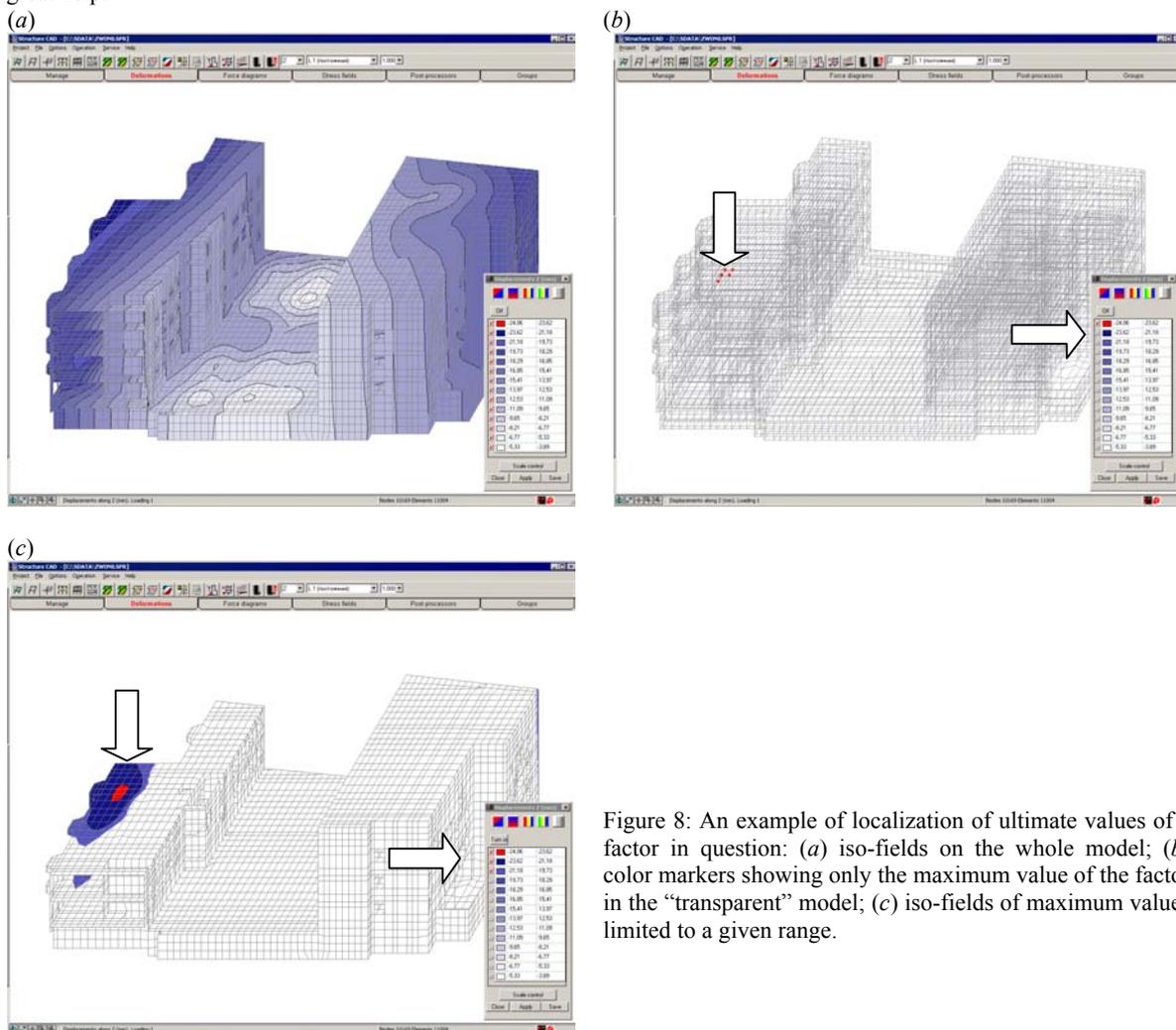
A pretty useful information to know is the total load upon a structure. This data helps analysts detect various blunders easily.

A problem can be verified on-the-fly when being solved. The process of solution can detect such model errors as geometrical changeability evidenced by zero eigenvalues and respective natural oscillation modes (see [3] for details). These modes show possible movements of the changeable system, actually a mechanism.

5. Reviewing calculation results

Results of static or dynamic computational analysis of a complex system, when represented by numbers, include enormous arrays of numerical data so their conscious reviewing is hardly possible. The capability of printing only selected results suggested by most software systems is of little help here because the analyst does not always know which of the values are critical.

Much better visual obviousness can be achieved by graphical representations of results such as plots, color maps, iso-fields etc. which compress the information to a great extent to make it perceivable.



Even this is not always enough for the results to be conveniently reviewed because the structure of the system as a whole may hinder the visual perception of output data (Fig. 8a) while its fragmentation will restore the obviousness but create a new problem: how to find the fragment which calculated values of interest belong to. It is quite nontrivial to resolve this issue in a complex model that contains tens or hundreds of thousands of nodes and elements. For example, Fig. 8a presents a model with iso-fields of vertical displacements applied to it, and the picture does not show the area of maximum values.

The way out of the situation can be a technique that the SCAD software suggests: advanced control of color indication. The first step is to use the color map of a factor of interest to find the location of needed values in the “transparent” model (Fig. 8,b). Then, display only those parts of the iso-field in color that belong to the needed range of values, and turn off all other levels (Fig. 8c). This will localize “critical” values in the computational results.

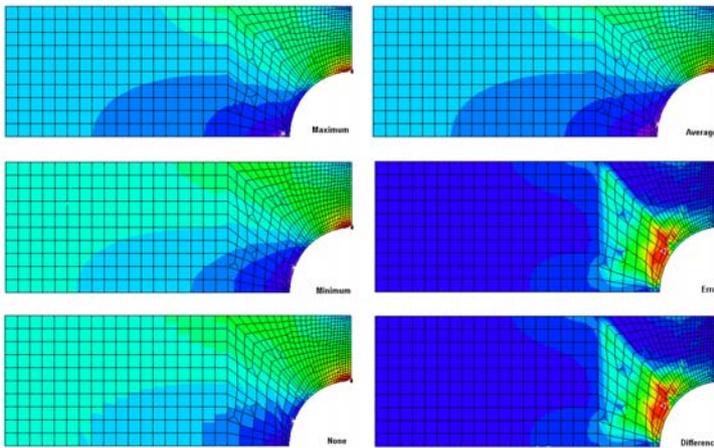


Figure 9: Various displaying methods

The averaging of stress fields should be used carefully when it involves constrained nodes. For example, take a rectangular plate clamped along the ABCDEF contour and along an interior line BE, and loaded inside the BCDE rectangle. There must be no moments on ABEF in this plate. Though, after the moments have been averaged, there appears a picture presented in Fig. 10.

A general graphical depiction of computational results is what conforms best to a well-known Hamming’s statement that the goal of calculation is understanding rather than numbers. Having analyzed the general picture, one can (and must) always turn to numerical results which now can be selected from the total amount consciously.

6. Analyzing energy distributions

It is useful to be aware of a universal tool for understanding peculiarities of stress/strain distribution in a system. This is a deformation energy distribution. If the energy distribution is built taking into account the effect of the geometrical stiffness matrix, then one can classify particular

Generally, iso-fields or iso-lines show a smoothed picture, where stress field discontinuities are hardly seen while they always accompany any finite element solution expressed in displacements. A developed system for analyzing such fields, first suggested in ADINA, is capable of displaying (and printing) the following data:

- raw discontinuous values (**None**);
- values averaged in nodes (**Average**);
- a maximum calculated over all elements in the vicinity of the node (**Maximum**);
- a minimum calculated over all elements in the vicinity of the node (**Minimum**);
- absolute difference between the maximum and minimum values (**Difference**);
- relative difference between the maximum and minimum values (**Error**).

Iso-fields of stresses that conform to the listed modes are shown in Fig. 9.

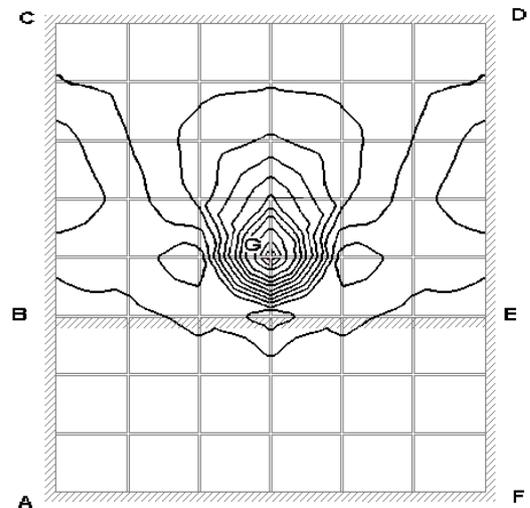


Figure 10: Error of averaging

parts of a system (down to its particular elements) into one of these two classes: a class of restraining and a class of pushing elements (or parts) of the system. [4] The restraining elements facilitate the stable equilibrium of the system while the pushing elements play the negative role because it is they which force (push) the mechanical system to lose its equilibrium stability.

The role played by particular subsystems is checked when counting the energy accumulated in different parts of the system in its deformation in a buckling mode. This energy is zero for the system as a whole, the parts of it where the energy is non-positive are pushing ones while the parts where the energy is positive should be classified as its restraining elements.

Pushing elements can be ranked by their “blame” for the critical state of the system using numerical values of the energy, because the contribution of each element of the system to its total energy balance can be a convenient quantitative measure of its responsibility for the stability of the equilibrium.

A pretty characteristic example is presented in Fig. 11 where the buckling mode is accompanied by a picture of the energy distribution (the pushing part of the structure is red).

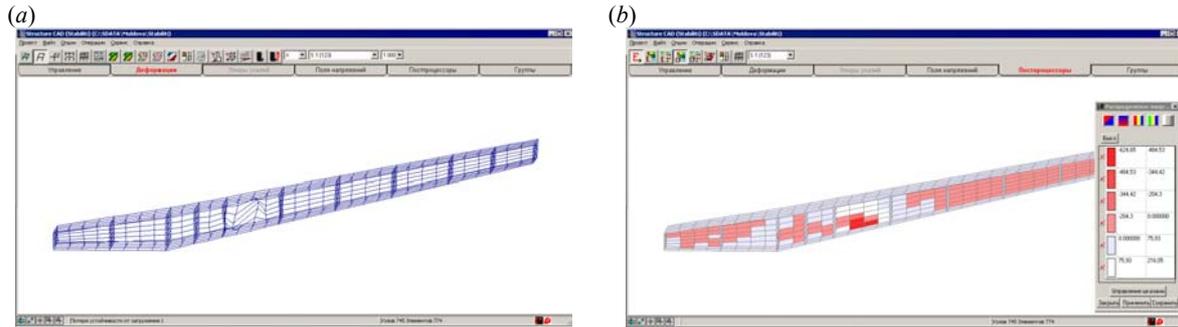


Figure 11: Buckling – loss of stability: (a) a buckling mode; (b) distribution of energy

7. Conclusion

Checking the correctness of large amounts of input and output information is a serious issue in practice of engineering applications of computational mechanics methods. This issue does not belong solely to the field of software implementation of computational techniques. It deals with the very essence of an engineering problem to be solved. Special tools used for this purpose need a careful research and improvement at least to the same extent as the computational techniques themselves. This is a worthy subject to draw the attention of computational mechanics experts.

References

- [1] Karpilovsky, V.S., Kryksunov, E.Z., Perelmuter, A.V. et al. *SCAD for users*. Kiev, Compass Publishing House, 2000 (in Russian)
- [2] Perelmuter, A.V. and Slivker, V.I. *The analysis of constructions — models and interpretations* 2nd ed., Kiev, "Steel" Publ. Co., 2002, (in Russian)
- [3] Fialko, S.Yu., Kriksunov, E.Z., Karpilovsky, V.S. *Block Lanczos method with spectral transformations for natural vibrations and seismic analysis of large structures in SCAD software* (in this Proceeding)
- [4] Perelmuter, A.V. and Slivker, V.I. *The Problem of Interpretations of the Stability Analysis Results // ECCM-2001. 2nd European Conference on Computational Mechanics. Solid, Structures and Coupler Problems in Engineering*. Cracow, Poland, June 26-29, 2001.— Abstracts, Vol. 2.— Kraków: Vesalius, 2001.— pp. 998–999 (Full paper on enclosed CD-ROM).