Conclusion

Initially, taking into consideration available resources, we attempt to build a polarity lexicon for Ukrainian language. In this paper we tried to tackle this problem employing graph polarity propagation algorithm. Since this pilot experiment was encouraging, this allows us in future work to generate the lexicon for Ukrainian language.

It should be stressed that the present research is aimed at laying a foundation for further researches in the field of sentiment analysis in Ukraine.

1. Agerri, R. Q-WordNet: Extracting polarity from WordNet senses / R. Agerri, A. Garcia-Serrano // Seventh Conference on International Language Resources and Evaluation, Malta, 2010. 2. Baccianella S. SentiWordNet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining / S. Baccianella et al.// Proceedings of the 7th Conference on Language Resources and Evaluation (LREC'10). Valletta, 2010. P. 2200–2204, 3. Esuli A. SentiWordNet: A publicly available lexical resource for opinion. – mining/ A. Esuli, F. Sebastiani // In Proceedings of the 5th Conference on Language Resources and Evaluation (LREC'06). Citeseer, Genova, 2006. – P. 417. 4. Hatzivassiloglou V. Predicting the semantic orientation of adjectives / Vasileios Hatzivassiloglou and Kathleen McKeown // In Proc. of the 35th ACL/8th EACL. – P. 174–181, 1997. 5. Hu M. Mining and Summarizing Customer Reviews / M. Hu, and B. Liu // In KDD, 2004. 6. Liu B. Sentiment Analysis and Subjectivity / B. Liu // Handbook of Natural Language Processing, Second Edition, 2010. 7. Miller, G. WordNet: A lexical database for English /G. Miller // Communications of the ACM. – 1995. 11. – P. 38. 8. Santos A. P. Determining the Polarity of Words through a Common Online Dictionary / A. P. Santos, C. Ramos and N. C. Marques // Volume Lecture Notes in Computer Science, 7026/2011. – P. 649–663, 2011 9. Stone, P.J. The General Inquirer: A Computer Approach to Content Analysis /P. J. Stone // M.I.T. Press; First Edition, 1966. 10. Strapparava C. WordNet-Affect: an affective extension of WordNet/ C. Strapparava, A. Valitutti // Proceedings of LREC. – P.1083–1086 Citeseer, 2004.

UDK 004.652.4:004.514

O. Markelov
Lviv Polytechnic National University,
Computer-Aided Design Department

DATABASE OF VISUAL AND SOURCE CODE COMPONENTS OF USER INTERFACE: INFORMATION MODEL DESIGN

In this paper proposed structure of database as part of repository UI software elements for functional integration of storage, search, selection of alternatives visual graphic element representations and their multi source code implementation in different programming languages.

Key words: usability, user interface, UI, information model, database, repository, environment

Запропоновано структуру бази даних для репозиторію елементів інтерфейсів користувача програмного забезпечення для інтеграції засобів накопичення, пошуку, підбору з альтернатив візуальних графічних представлень елементів та їх мульти-реалізації програмними кодами на різних мовах.

Ключові слова: зручність використання, інтерфейс користувача, інформаційна модель, база даних, репозиторій, середовище розробки.

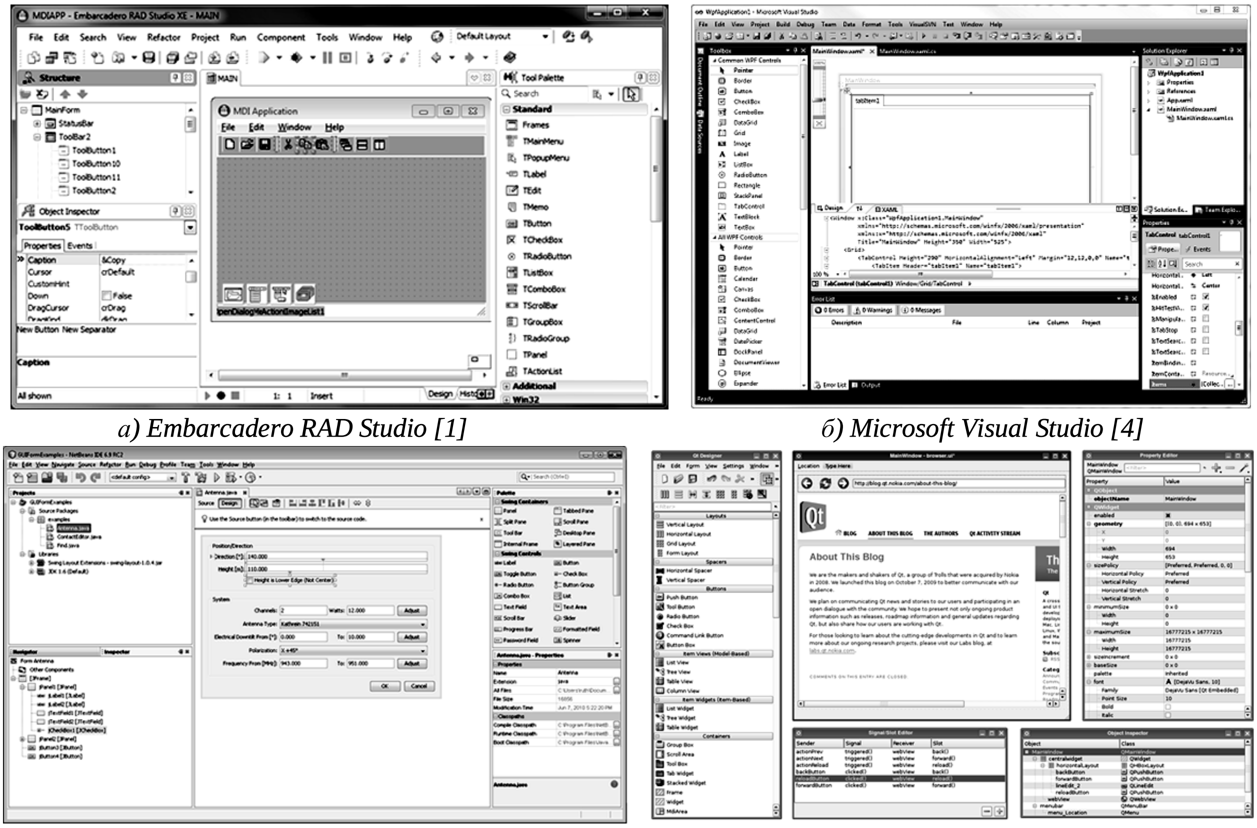
Problem and actuality

Most of today's popular software development environments [1-5], sketching and prototyping interfaces [6-18] not allow for an alternative choice among: or visual elements and program source of user interface controls; or search; or selection of alternatives software source codes of these elements. Set of the available interface elements for UI design is the result of the selection of a programming environment or interfaces prototyping software. So the problem of selection owns UI toolkit for a specific project is not an unrelated task of selection the programming environment.

Overview of user interfaces designing tools

When creating a new version of software programmers and managers must have the opportunity to compare a components of screen interface that implemented by different technologies [20, 21], to select them for use. At present, integrated development environments (IDE) used only sets of the preinstalled components with reference to the source programming language, for which those IDE were designed.

List, basis of the review [22], of typical elements of visual user interface, that are commonly available in different IDE, currently, has 98 items: Button, Icon Button, Toggle Button (switch button), Split Button, Cycle Button, Button bar, Checkbox, Check box list, Radio button, Radio button list, Spinner (StepperInput), Switcher (Toggles), Dynamic Layout, Static Layout, Frame, Fieldset, Group Box (Titled panel, Group panel), Panel, Collapsible panel, Split panel, Splitter (H-divider, V-divider), Tabs (Pages, Tab panel, Tab container, Wedge bar), Vertical Tabs, Icon, Image, Separator (line), Shapes, ChartArea (Bar chart, Pie chart, Line graph), Label, Bulleted list, Numbered list, Link, Hyperlink, Linkbar, Text area (Note), Code block, Greeking (underlined string), Slider, Rating (Stars), Tooltip (Hint), Balloon help, Infobar, Toast (Notifier), Status bar, Progress bar, Progress Indicator, Gauge, Scrollbar, Zoomer, Text Edit (Edit box, Field), Password box, Text Box (Memo, Editable Text area), RichText, List box (Value list), Drop-down list, Combo box, Tree view, Grid view (Table), Menu, Hierarchical Menus, Context menu (Popup menu), Pie menu (circle menu), Menu bar, Menu extra (Dock), Toolbar, Ribbon, Thumbnail grid, Accordion, Carousels, Coverflow Carousel, Place holder, Window, Modal window, Pop-up window, Dialog box, Alert box, Search box, Address bar, PathViewer (Breadcrumb), List Navigation Bar, Pagination, Tag Cloud, AlphaNumeric (Quick jump alphabet), Pointer (Cursor), HotSpot, Disclosure widget, Palette window (Utility window), Inspector window, ColorPicker, Calendar, Drop-down calendar (Date picker),Calculator, Clock, Timer, Audio player, Video player, Webcam, Map, Captcha, Banner ad. This list is regularly updated with new data from the new reviews of IDE tools.



а) Embarcadero RAD Studio [1]

б) Microsoft Visual Studio [4]

в) NetBeans IDE GUI Builder [3]

г) Qt Creator IDE [5]

Fig. 1. Examples of user interfaces design environments

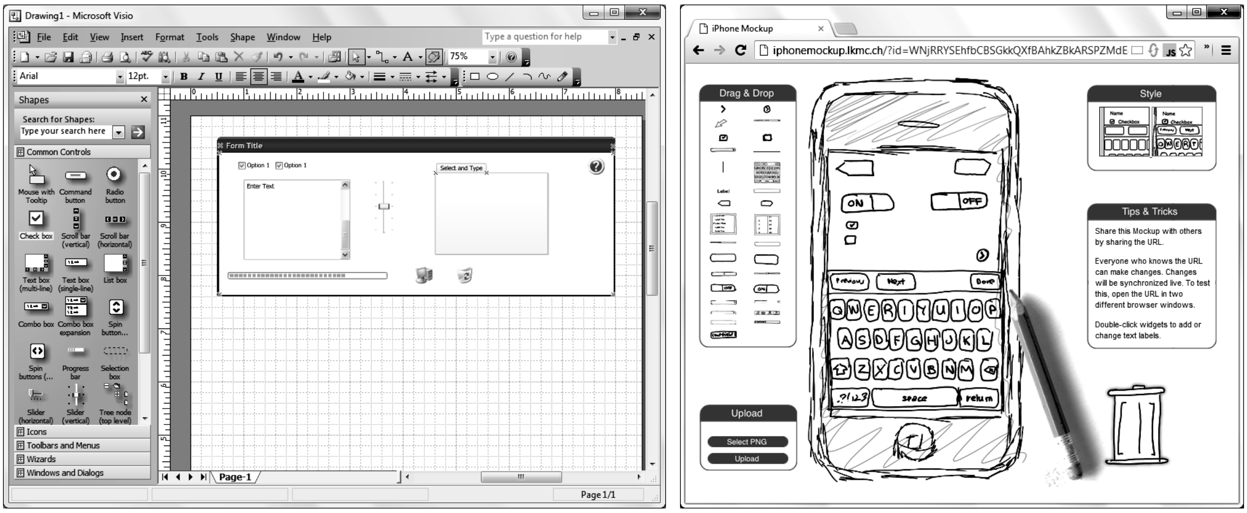
On figure 1 the examples of window interfaces of integrated design environment for window forms design based on a set of standard UI controls or widgets presented.

The developers can not export / import ready-made GUI forms from one programming environment from one programming language to another environment or another language. Software editors of window GUI contain the basic set of elements for the user interface. You could increase the number of components by connecting additional third party libraries. The programmer manually selects only those GUI components that are available from the list of IDE.

For developers of new software interface concepts (or upgrading older) will be useful to have central reference and search tool, which includes different implementation versions of a user interface.

In addition to interface design environments there are interfaces prototyping environments [1-18]. Prototype of interface project is a version of the application or website in simplified presentation form. Prototypes are created in specialized editors.

Prototypes are called wireframes when they are abstracted from the details of graphic design and focus on planning and structuring graphical user interface, using block-placeholders. Prototype allows showing placement of widgets on the device screen.



Share this Mockup with others by sharing the URL.

Everyone who knows the URL can make changes. Changes will be synchronized live. To test this, open the URL in two different browser windows.

Double-click widgets to add or change text labels.

a) Microsoft Visio [6]

б) iPhone MockUp Online tool [18]

Fig. 2. Screens of environments for creating prototype model of user interfaces

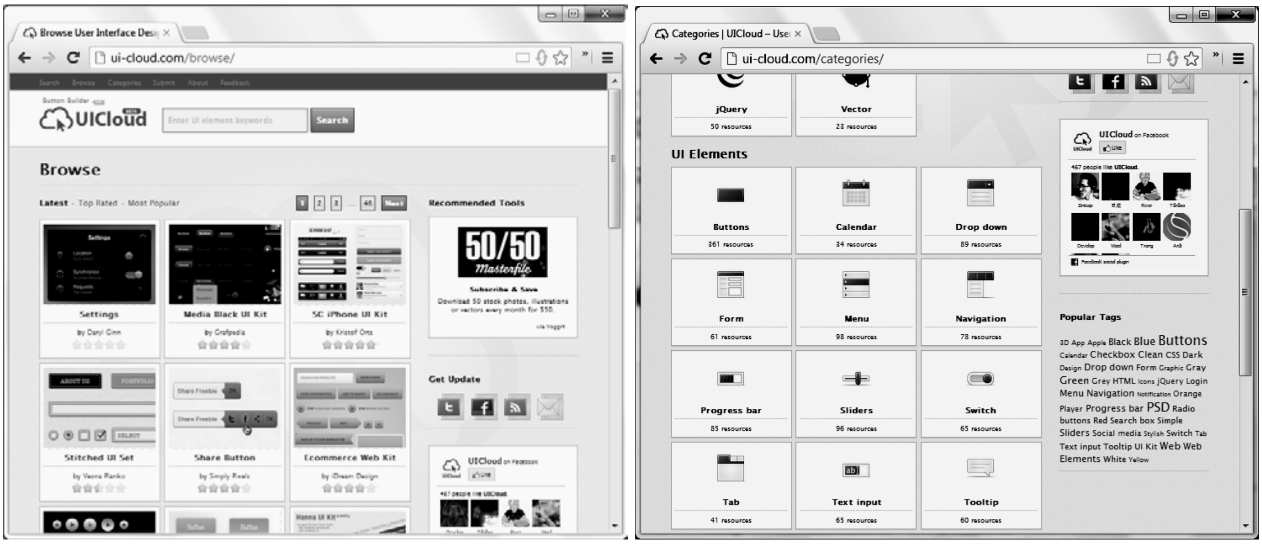


Fig. 3. Screen examples of GUI online-store “UICloud” [19]

Figure 2 presents examples of window screen of two environments for creating prototype model of user interfaces. All examined environments are not allowed to generate program source code based on implementation of the developed structure of the interface prototype.

Analogue repository could be online UICloud store of user interface examples [19] (screen form shown in Fig. 3).

Inconvenience: not represent all types of GUI items; there is no variety of program code implementation.

The purpose of developing

The purpose of this study is: systematic analysis of features of programming tools, tools of creating prototype model of software user interfaces, collection approaches and implementation source code in different programming languages of common and specific elements of visual user interfaces; implementation of systematization and controls classification of visual user interface and on those data to develop information models of UI controls repository and their software implementation.

Database of the parameters, characteristics and implementation the visual components of a software user interface makes it possible to catalog the items to organize the search for alternatives functional elements and source code. Repository provides choice and versions comparison of source code in different programming languages.

Information structure

Information structure (Fig. 4) of repository contains the following tables:

1. “Controls” − list of all available versions of one or more controls;
2. “Types” − grouping controls in to type lists;
3. “Related_controls” − identifies controls that may be similar as alternative to functional activity, but it are another type of;

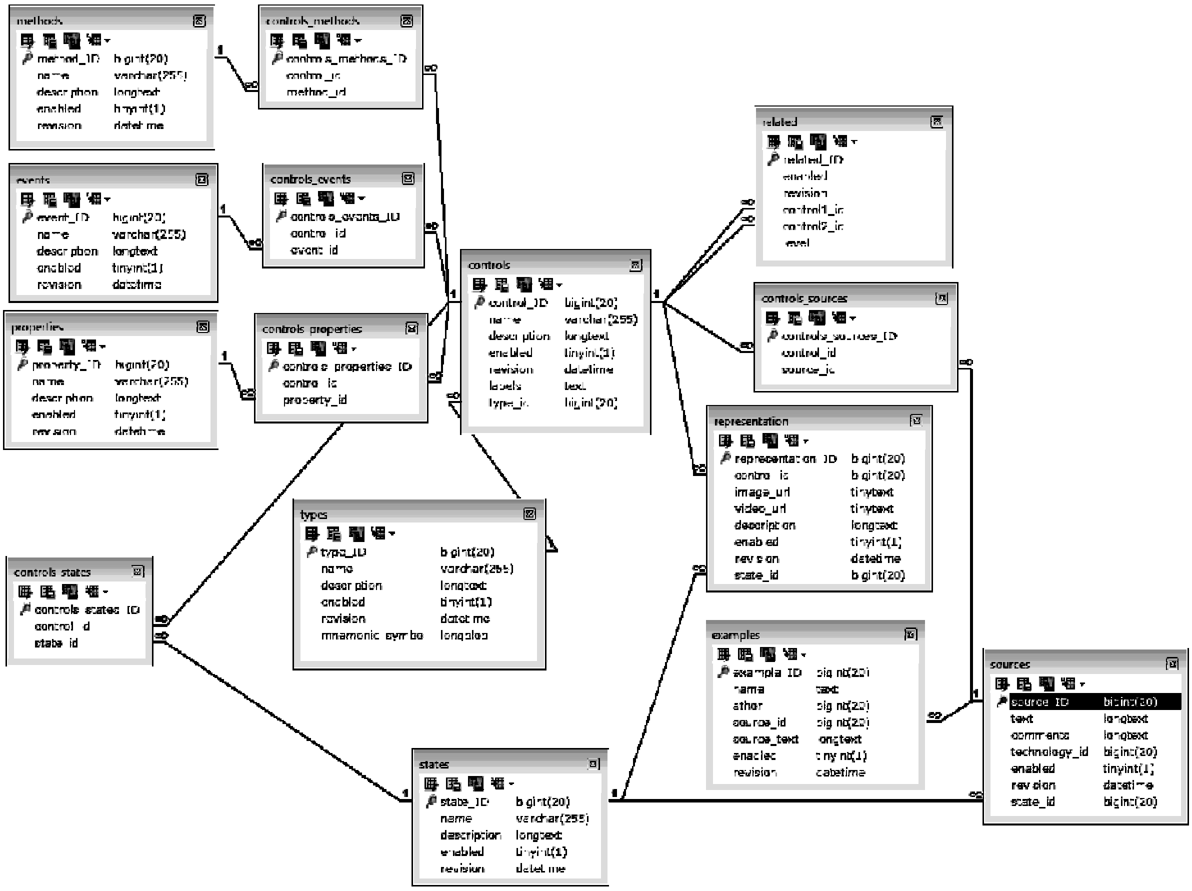


Fig. 4. Relational database model of user interface components repository

4. “Properties” − describes a list of all the properties that describe look of the element, its functional characteristics;

5. “Events” − describes the reaction list of the control to user actions or system events; “Methods” − describes a list of all functions that can be activated by interface elements, as a reaction to events;

6. “States” − describes information about visual display states before, during, after the interaction with the user or system events;

7. “Representation” − visual images, animation properties of the control in different states or alternative graphical modifications;

8. “Sources” − program source codes in different available programming languages;

9. “Examples” − ready to use different versions of software implementation of selected UI controls.

All tables have a field of “Revision” (“Edition/Version”), which contains the date and time values. It allows you to store multiple versions of the same data. If necessary, you can revert to earlier versions or to determine when and who made a particular change.

Field “Enabled” (“Available”) in all tables specifies whether available to be used any of the parts of the database information. Field “Description” in tables containing technical comments or comments to the parts of the database information model.

Conclusions

Overview of functions of more than 30 modern software [1-19] for sketching, for layout mockup, for prototyping and for user interface programming; review [22] of more than 98 types of GUI controls − allowed to create the data model that used to select from a set of alternatives in the visual and source code for the user interface design. The results are useful for automation of the design work within the selection and development of user interfaces of software, program source code generation and automated designer's workplaces.

1. Embarcadero Delphi and C++Builder Quick Start Tutorial, Product Documentation, XE Release, PDF Last Published October 18, 2010, Mode of access: http://www.embarcadero.com/products/rad-studio – Last access: 2012-Sep-10. – Title from the screen. 2. Lazarus Documentation, Lazarus is a free cross-platform IDE, Mode of access: http://www.lazarus.freepascal.org/ – Last access: 2012-Sep-12. – Title from the screen. 3. NetBeans IDE – free, open-source Integrated Development Environment for software developers, Swing GUI Builder Mode of access: http://netbeans.org. – Last access: 2012-sep-20. – Title from the screen. 4. Microsoft Visual Studio 2010. Mode of access: http://www.microsoft.com/visualstudio. – Last access: 2012-Sep-24. – Title from the screen. 5. Qt – cross-platform application and UI framework. Online Reference Documentation, Nokia Corporation. Mode of access: http://qt.nokia.com/. – Last access: 2012-sep-27. – Title from the screen. 6. Microsoft Visio Professional. Mode of access: http://www.microsoft.com – Last access: 2012-Sep-10. – Title from the screen. 6. Evolus Pencil. Mode of access: http://pencil.evolus.vn – Last access: 2012-Sep-10. – Title from the screen. 7. Balsamiq Mockups. Mode of access: http://www.balsamiq.com/products/mockups – Last access: 2012-Sep-10. – Title from the screen. 8.Pidoco. Mode of access: https://pidoco.com – Last access: 2012-Sep-10. – Title from the screen. 9. Produle Systems Mockflow. Mode of access: http://www.mockflow.com/ – Last access: 2012-Sep-10. – Title from the screen. 10. Rudrasoft MockupTiger. Mode of access: http://www.mockuptiger.com/ – Last access: 2012-Sep-10. – Title from the screen. 11. Cyberopsis Webdata Proto.IO. Mode of access: https://proto.io – Last access: 2012-Sep-10. – Title from the screen. 12. Site9 Inc. ProtoShare. Mode of access: http://www.protoshare.com – Last access: 2012-Sep-10. – Title from the screen. 13. Axure Software Solutions. Mode of access: http://www.axure.com – Last access: 2012-Sep-10. – Title from the screen. 14. Tool Software UI Mockup. Mode of access: http://www.toolsoft.net/ui-mockup.html – Last access: 2012-Sep-10. – Title from the screen 15. ERGOSIGN Technologies Antetype. Mode of access: http://www.antetype.com – Last access: 2012-Sep-10. – Title from the screen. 16. Endloop Systems iMockups. Mode of access: http://www.endloop.ca/imockups – Last access: 2012-Sep-10. – Title from the screen. 17. CATCH Screen Architect. Mode of access: http://www.catchlimited.com/index.php?page=screen-architect – Last access:

2012-Sep-10. – Title from the screen 18. iPhone MockUp / Lukas Mathis Mode of access: http://iphonemockup.lkmc.ch – Last access: 2012-Sep-10. – Title from the screen 19. UICloud. A project by Double-J Design. Mode of access: http://ui-cloud.com – Last access: 2012-Sep-10. – Title from the screen 20. Лобур М. Аналіз мов опису користувацьких інтерфейсів для застосування у програмному забезпеченні САПР / Лобур М., Маркелов О., Бобало С. // Вісн. Нац. ун-ту "Львів. політехніка"ю – 2010. – № 663: Комп'ютерні науки та інформаційні технології. – С. 30–38. – Бібліогр.: 25 назв 21. Lobur M. "UI Technologies for CAD Software Development: Evaluative Approach of Componentry Applications" / Mykhaylo Lobur, Oleksandr Markelov // Modern problems of Radio Engineering, Telecommunications and Computer Science / Proceedings of the X International Conference TCSET'2010, Feb 23-27, 2010, Lviv-Slavske, Ukraine; LPNU. – Lviv.: Publishing house of LPNU, 2010. – 380 p.: – PP. 41. 22. Oleksandr Markelov, "The Review of CAD Software in Machinery Design: Spectral Characteristics of GUI components", / Oleksandr Markelov, Mykhaylo Lobur // CAD in Machinery Design. Implementation and Education Problems: Proceeding of the XX Ukrainian-Polish Conference CADMD'2012 (Oct 11-13, 2012). – Lviv, Ukraine: Publishing House Vezha&Co, 2012. –136 c. –P. 96–98, ISBN: 978-617-607-332-1.

UDK 004.9

M. Lobur, I. Farmaga, P. Shmigelskyi
Lviv Polytechnic National University,
CAD Department

EVALUATION THE SOLUTION TIME OF FINITE ELEMENT ANALYSIS USING GAUSSIAN ELIMINATION

The paper describes method allowing predicting solution time of finite element analysis. The method is based on statistical data and computational complexity of software algorithms.

Key words: FEM, CAE, computational complexity, solution time.

Описано метод, що дає змогу прогнозувати час виконання скінченно-елементного аналізу. Метод оснований на статистичних даних і складності обчислення програмно-реалізованих алгоритмів.

Ключові слова: скінченно-елементний аналіз, CAE, складність обчислення, час рішення.

1. Demand for solution time prediction

Nowadays FEM is the most widely used method for the analysis of physical problems described by differential equations. Its main advantages are the high flexibility and versatility, which makes its use for solving a variety of physical problems. On the other hand, the method takes a lot computing resources. And the more accurate results are needed, the more resources are required. Engineers always need to balance between accuracy and reasonable solution time.

The article describes methods that allow one to predict how much time will be spent on the solution by FEM. With these data engineer can decide whether to continue the solution of the problem, or to return to the configuration of analysis so the solution be obtained within a reasonable time.

2. The principle of solution time prediction

The presented method for FEM solution time prediction is based on two components: known computational complexity of the solution algorithm; and collection of statistical data of solved of tasks.

Computational complexity is presented in the form of O-notation. Substituting specific values in its expression we obtain a number that we call coefficient of complexity. Solution time of task is almost