MEASURING EFFICIENCY OF FORMALLY REPRESENTED INSTRUCTIONAL STRATEGIES

Goran Savić, Milan Segedinac, Aleksandar Kovačević, Zora Konjović
{savicg, milansegedinac, kocha78, ftn_zora}@uns.ac.rs
University of Novi Sad, Faculty of Technical Sciences, Novi Sad, Serbia

Abstract – *This paper deals with comparing the efficiency of guided and unguided learning using formal representation of an instructional strategy. Strategies were formally represented with ELIDL – an XML-based language specially designed for this purpose. Three instructional strategies were defined. Two of them, named worked-out examples and process-worksheets, represent guided learning, while the third one represents unguided learning. The efficiency of the strategies was evaluated on the course "Numerical algorithms and numerical software in engineering" held at the Faculty of Technical Sciences at University of Novi Sad. A sample of 81 students was divided into three groups, where each group was given the course created according to different instructional strategy. The results show that the most efficient instructional strategy is process-worksheets strategy.*

1. INTRODUCTION

Engineering education not only covers theoretical background, but also enables students to solve real-world problems. This implies that a teacher should define an appropriate instructional design which helps students to solve the problem. There are two basic approaches regarding the role of a teacher in a problem-solving process. The first one is a traditional approach, commonly named *guided learning*. In this approach, a teacher helps student by providing various guidelines. The second approach is more recent and it implies more limited role of a teacher. The basic idea of this teaching technique is that a student should discover a problem and solve it individually. This technique is commonly named *minimally guided* or *unguided learning*, but the terms *discovery learning* [1][2], *inquiry learning* [3][4], *experiential learning* [5][6] and *constructivist learning* [7][8] are also used[9].

So far, a number of studies have been conducted to compare the efficency of guided vs unguided learning [10][11]. A part of the scientific community is sceptical about the effectivness of the unguided learning and within their researches, they show the inefficiency of this teaching method[9].

This paper presents yet another case study about the efficiency of the guided vs unguided learning in problem-solving situations. We have tested the efficency of the two guided learning methods – *worked-out examples* and *process-worksheets* compared to a pure unguided learning. These three instructional strategies have been applied to the course "Numerical algorithms and numerical software in engineering" held at the Faculty of Technical Sciences at University of Novi Sad.

This paper is organized as follows. Formal representation of the evaluated instructional strategies is presented in Section 2. Sections 3 and 4 describe measurements on the efficiency of the strategies. Finally, conclusions and future work are given in Section 5.

2. FORMAL REPRESENTATION OF AN INSTRUCTIONAL STRATEGY

In general, the efficiency measurement of an instructional strategy requires a clear and formal definition of the observed strategy.

Instructional strategy (often referred to as *instructional design*) represents specific pedagogical principles upon which a learning process has been designed. Reigeluth and Moore in [12] explain that instructional design is a theory that offers an explicit guide about how to teach to learn. In [13] authors define instructional design as a process for the design and development of instructional materials and learning activities based on learning theory research. This is precised in [14] in a form that instructional design is the systematic specification of instruction to include: objectives, presentation, activities, materials, guidance, feedback and evaluation. It applies learning principles to decisions about information content, instructional method, use of media and delivery system. Instrucional design is applied in a course in order to ensure instructional quality, effectiveness, efficiency and enjoyment [14].

Our research examines formal representation and machine-based management of instructional design. We are focused on instructional design in a sense of organization of learning resources in the course. Take for example a course, which contains illustrative examples, followed by theoretical background. By this, course's authors have embedded specific instructional technique, named *learning by example*. So, among various interpretations of the term *instructional design*, in our research we understand instructional design in accordance with the Willey's definition[15] as sequencing and selection of learning activities.

Defining instructional design as a sequence of learning activities is possible using so-called instructional design languages [16]. These languages describe concrete learning process emerged as a result of applying specific instructional strategy on a concrete course content. With these languages, instructional design is implicitly defined within learning proces, but is not defined in an explicit and domain-independent manner. Such an implicit definition prevents an independent management of the instructional design in a course. In our research we are focused on representing a generic instructional strategy, which is independent from concrete learning activities. Generic instructional strategy represents a template which can be applied on any concrete course.

Creating templates which describe generic instructional strategies has been analyzed in the Pedagogical Patterns Project [17]. The project specifies a collection of pedagogical patterns commonly used in educational institutions. A pattern identifies best pedagogical practice in a specific domain. The project's goal is to enable sharing and reusability of the practical pedagogical knowledge. Although very useful for teachers and pedagogues, these patterns are not intended for computer interpretation. The patterns are represented using textual descriptions, so they are not machine-readable.

In [18] authors describes how to convert aforementioned pedagogical patterns from a natural text to the machine-readable IMS Learning Design [19] (IMS LD) language. The result of the conversion is not a concrete IMS LD course, but a template which represents generic learning activities. Still, IMS LD is designed to describe concrete learning activities and it is not appropriate for representing generic learning process.

The second research which also formalizes the representation of pedagogical patterns is described in [20], where authors present a structured description of a pedagogical pattern. Although the structure of a pattern is clearly defined, it still contains textual fields which makes it inadequate for the machine processing.

Derntl in [21] presents PceL project with a repository of patterns which describe learning process. The repository contains a predefined set of patterns defined in the UML notation. Within concrete course, a specific pattern is identified and matched to an appropriate pattern from the repository.

AUTC [22] project presents 5 representative patterns which describe instructional design. Patterns are defined in a textual and visual form and provide guidelines for creating concrete teaching courses.

In [23] Hayashi et al. presents a method for the formal definition of instructional strategies using ontologies. The strategy defines a sequence of learning activities in a course. However, more complicated learning process flows, like conditional branching, are not supported yet.

LAMS project has developed a tool named Activity Planner [24] which provides defining a generic LAMS course. Using this tool, a user doesn't specify concrete learning activities, but a template which presents a sequence of apstract learning activities. Based on this template, a user can create a course with concrete learning content later. Although it has been designed generically, the LAMS template is still LAMS-specific and it can be only used to generate a course in the LAMS format.

By analyzing instructional design description in all these papers, two conclusions about instructional design treatment can be derived: (1) instructional design is not represented as a machine readable format and/or (2) expressiveness and flexibility are not sufficient for describing a variety of instructional techniques (in most papers these techniques are predefined or the underlying planning mechanisms are of general type which cumbers defining a new instructional techniques).

In order to enable machine processing of instructional design, it is necessary to define instructional design in an explicit, formal and machine-readable manner. For this purpose, we have developed E-learning Instructional Design Language (ELIDL) [25][26] specially designed to provide machine-readable description of generic instructional strategies. An instructional strategy defined in ELIDL may be considered as a set of rules or a template which defines a general method for sequencing and selection of learning activities. This template applied to the concrete learning content gives a concrete teaching course as a result. We have decided that ELIDL should be XML-based language. Such an approach ensures machine readability, portability, reusability and multi-platform support. At the same time, XML is a textual format, which is human-readable. The XML schema for ELIDL, altogether with the explanation of all schema elements, is publicly available at http://www.informatika.ftn.uns.ac.rs/GoranSavic/IDTemplates. Also, representative examples of instructional strategies written in ELIDL are available at http://www.informatika.ftn.uns.ac.rs/GoranSavic/IDTemplates/Examples.

To measure the efficency of different instructional strategies, we defined three strategies using ELIDL - *worked-out examples*, *process-worksheets* and *unguided learning*. First two strategies are related to the guided learning. These strategies specify that a teacher helps student to solve a problem by providing various hints which bring the student closer to the problem solution. The third instructional strategy is related to the unguided learning where a student individually solves the problem with a minimal participation of a teacher. The strategies are explained further below.

2.1. Worked-out examples

This instructional strategy is a kind of guided learning where a student gets representative examples as assistance in problem solving. The examples focus learners' attention on the concrete problem, enabling them to induce more generalized solutions [27]. In our research we have formally represented this strategy using ELIDL. Created ELIDL document is shown in listing 1.

```
<instructional-design root = "course">
 <uol-structure>
  <sequence element = "learning-object">
   <selection-rule>
     <include att-name="type"
              att-value="exercise"
              priority="1"/>
     <include att-name="type"
              att-value="example"
              priority="2"/>
   </selection-rule>
  </sequence>
  <sequence element = "lesson">
   <sequence element = "topic">
    <sequence element = "learning-object">
     <selection-rule>
      <include att-name="type"
               att-value="*" priority="1"/>
     </selection-rule>
    </sequence>
   </sequence>
  </sequence>
 </uol-structure>
 <element-relationships />
</instructional-design>
```

Listing 1. *Worked-out examples* instructional strategy specified with ELIDL

The element `uol-structure` specifies learning elements in the course which should be shown to a student. We can notice that a sequence of learning objects is shown firstly. Within `selection-rule` element is defined that, among all learning objects in the sequence, assignments (learning resources annotated with the meta-data `exercise`) are shown firstly. Then, a student gets examples (learning resources annotated with the meta-data `example`). The order of these two types of learning objects is specified using `priority` attribute in `include` element. After that, other learning material is shown by iterating through the sequence of lessons. Within each lesson, the sequence of its topic is chosen. For each topic, all its learning objects will be presented to a student, which is defined with `include` element and `att-value` attribute with the value "*". This instructional strategy specifies that all learning material is always available to a student. There are no any conditions for displaying learning content. Therefore, `element-relationships` element is empty, because we don't have to define any relationships among learning elements.

2.2. Process-worksheet

In this type of guided learning, a student gets process-worksheets as assistance in problem solving. Process-worksheets contain a description of the activities that should be performed in order to solve the problem [27]. Process-worksheets divide the problem in a sequence of simpler tasks whose solving leads to the final solution.

A part of the ELIDL document, which specifies the *process-worksheets* instructional strategy is shown in listing 2.

```
<instructional-design root = "course">
 <uol-structure>
  <sequence element = "learning-object">
   <selection-rule>
     <include att-name="type"
              att-value="exercise"
              priority="1"/>
     <include att-name="type"
              att-value="process-worksheet"
              priority="2"/>
   </selection-rule>
  </sequence>
...
```

Listing 2. *Process-worksheets* instructional strategy specified with ELIDL

Similar to the previous strategy, this strategy presents all learning material from lessons to a student. So, this part of the ELIDL document is the same for both strategies and, due to limited space, it is not shown in the listing 2. Listing 2 shows only a part of the ELIDL document that is different from the document presented in listing 1. We can notice that the strategy presented in listing 2 includes in the course learning objects, which contain process-worksheet (learning resources annotated with the meta-data `process-worksheet`).

2.3. Unguided learning

In unguided learning, a student solves the problem by discovering learning material individually. This strategy is formaly specified in ELIDL document shown in listing 3.

```
<instructional-design root = "course">
 <uol-structure>
  <sequence element = "learning-object">
   <selection-rule>
     <include att-name="type"
              att-value="exercise"
              priority="1"/>
   </selection-rule>
  </sequence>
...
```

Listing 3. *Unguided learning* instructional strategy specified with ELIDL

Once again, listing shows only a part of the ELIDL document which is different from the document shown in listing 1. Listing shows that a student gets assignments firstly. But, in contrast to the other strategies, there are no any additional learning resources which may help student in problem solving.

3. METHOD

Three previously described instructional strategies were applied to the learning material of the course "Numerical algorithms and numerical software in engineering" held at the Faculty of technical sciences at University of Novi Sad. Using an e-learning course generator, described in [25], three teaching courses were generated. Learning processes in the generated courses are in accordance with the *worked-out examples*, *process-worksheets* and *unguided learning* instructional strategy, respectively. Previously, the existing learning material of the course was converted to the HTML format. The course generator generated the courses in the IMS LD format. This format can be presented to a student using an appropriate software tools. Figure 1 shows the course "Numerical algorithms and numerical software in engineering" in the IMS LD format, generated in accordance with the *worked-out examples* instructional strategy. The screenshot is taken from the Reload IMS LD Player [28]. The figure presents a representative example, which helps student to solve the problem. Course content is in Serbian.

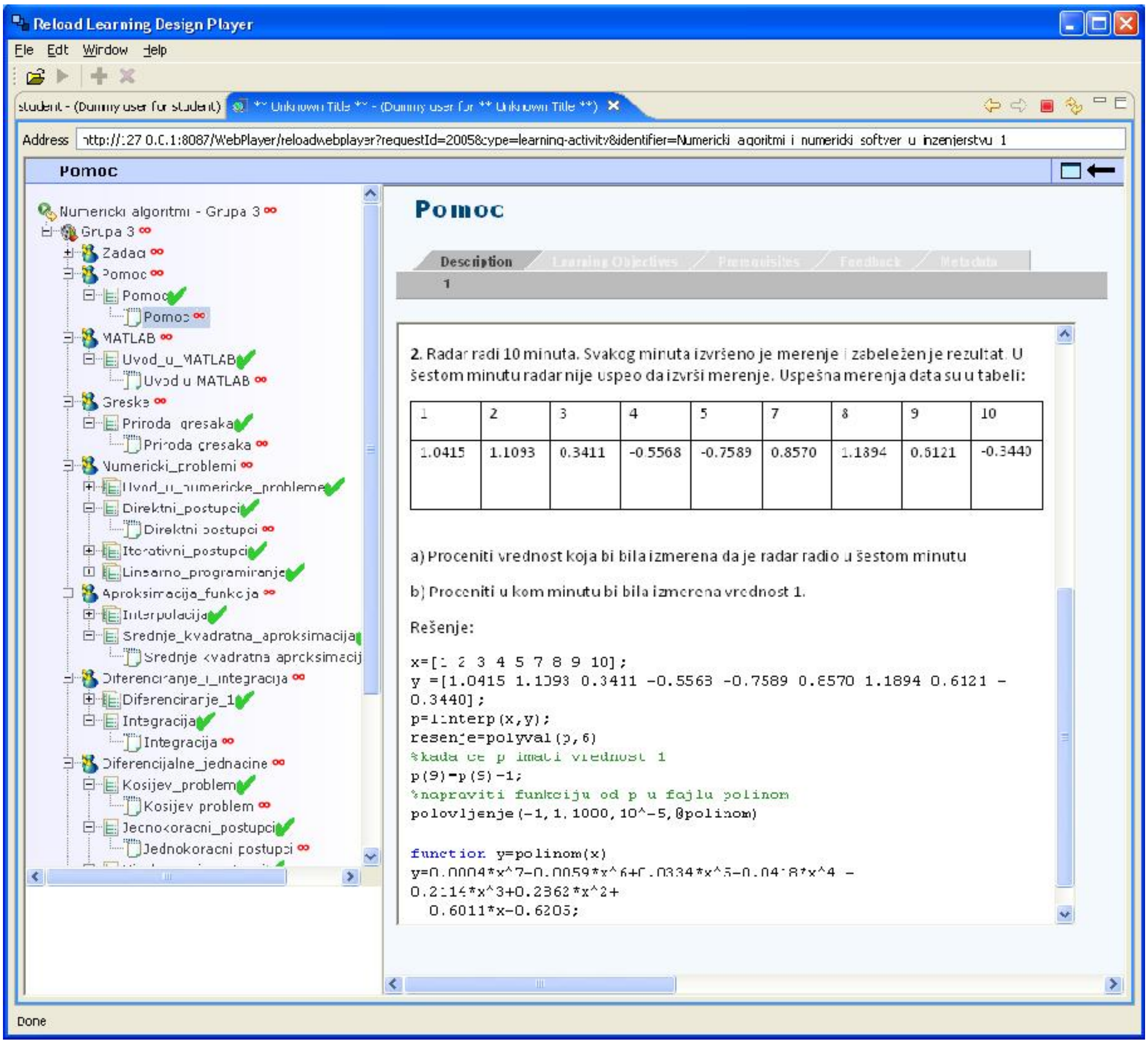

Figure 1. IMS LD course generated in accordance with the *worked-out examples* instructional strategy

The efficiency of the described instructional strategies was measured on a sample of 81 students. Each student was given 2 assignments related to the course domain. In both assignments, a student was expected to solve a particular problem by implementing an appropriate numerical algorithm written in MATLAB[28]. Students were randomly divided into three groups of 27 students. The first group was given the IMS LD course generated according to the *worked-out examples* instructional strategy. Besides the explanatory learning material, this course contained representative examples which solve problems similar to the given assignments. In these examples, students were able to see the syntax of the MATLAB language. The second group was given the IMS LD course generated according to the *process-worksheets* instructional strategy. In this course, for each assignment a student got guidelines in a form of natural language. The guidelines consisted of a set of tasks that should be performed to create the final algorithm. Finally, the third group was given the IMS LD course generated

according to the *unguided learning* instructional strategy. This course contained only explanatory learning material without any kind of assistance in problem solving.

The points scored on the assignments have not participated in the course final grade. Since the students didn't take the assignments under the same conditions, the results are used only for research purposes and it has not been possible to score the results officially.

4. RESULTS

Each assignment was scored with 5 points. Table 1 presents the average number of points on the assignments for each test group. Also, the average number of total points scored in both assignments is listed.

	Group 1-WOE		Group 2-PW		Group 3 – unguided learning	
	M	SD	M	SD	M	SD
Asgn. 1	1,7	1,53	2,85	1,95	1,55	1,69
Asgn. 2	1,67	2,99	2,48	3,53	0,85	2,14
Total	3,37	3,78	5,33	4,2	2,41	2,24

M – points average
SD – standard deviation
Asgn. – assignment
WOE – Worked-out examples
PW – Process worksheets

Table 1. Efficiency measures for *worked-out examples, process-worksheets* and *unguided learning* instructional strategies

The results show that the process-worksheets group had the best score. The lowest average number of points was scored by students in the *unguided learning* group. Among two guided learning instructional strategies, the *process-worksheets* strategy achieved significantly greater efficiency. We can notice that adding worked-out examples in the course improved student achievement just slightly. These results indicate that the students had the biggest problem with the algorithmic knowledge. Students, who got the algorithm description in a form of natural language, succeeded to solve the problem by writting an appropriate MATLAB code.

We performed a one-way ANOVA test (with alpha of 0.05) to determine if there is a statistically significant difference in scores achieved by the three groups of students. The obtained p-value was 0.01 indicating that the choice of instructional strategy has a statistically significant impact on the students' performances.

Relatively high values of standard deviations indicate different levels of knowledge and/or motivation between students. The large number of students scored the minimum or maximum points on the assignments. Future research should examine the reasons for such result.

As mentioned earlier, the scores were not officially graded. Generally low average number of points can be explained by the lack of motivation among students in the sample.

5. CONCLUSION

The paper presents a case study on measuring the efficiency of different instructional strategies. Instructional strategies are formally described using the machine-readable ELIDL language. The paper compares the efficency of the three instructional strategies, named *worked-out examples*, *process-worksheets* and *unguided learning*.

The research has been conducted on a sample of 81 students who took the assignments from the course "Numerical algorithms and numerical software in engineering". The students were divided into three groups. Each group got the course created according to a one of the three aforementioned instructional strategies. The most successful were students from the *process-worksheets* group. Between two other strategies, the *worked-out* examples strategy has produced slightly better performances than the *unguided learning* strategy. From the results, we can conclude that students have a problem with devising a correct algorithm which solves the assignment. Students in the *process-worksheets* group, who were given the algorithm, got the best scores on the assignments.

Given the relativelly small sample and the lack of motivation among students, the presented results should be taken carefully. Further research should include retesting with a larger sample with the student's motivation taken into account. Also, future work should determine reasons why worked-out examples have not noticeably improved students achievement. Another feature that will be measured in a further testing is how student's prior knowledge and motivation affects the results.

6. REFERENCES

[1] J. S. Bruner: *The art of discovery,* Harvard Educational Review 31, 21-32, 1961.

[2] W. S. Anthony: *Learning to discover rules by discovery,* Journal of Educational Psychology, 64(3), 325-328, 1973.

[3] S. Papert, Mindstorms: *Children, computers, and powerful ideas*, New York: Basic Books Inc, 1980.

[4] F. J. Rutherford, *The role of inquiry in science teaching*, Journal of Research in Science Teaching, 2, 80-84, 1964.

[5] D. Boud, R. Keogh, D. Walker: *Reflection: Turning experience into learning,* London: Kogan Page, 1985.

[6] D. A. Kolb, R. Fry: *Toward an applied theory of experiential learning*. In C.Cooper (Ed.), Studies of group process, (pp. 33-57). New York, NY: John Wiley & Sons, 1975.

[7] D. Jonassen: *Objectivism vs. Constructivism,* Educational Technology Research and Development, 39(3), 5-14, 1991.

[8] L. Steffe, J. Gale: *Constructivism in education,* Hillsdale, NJ: Lawrence Erlbaum Associates, 1995.

[9] P. Kirschner, J. Sweller, R. E. Clark: *Why Minimal Guidance During Instruction Does Not Work: An Analysis of the Failure of Constructivist, Discovery, Problem-Based, Experiential and Inquiry-Based Teaching*, Educational Psychologist 41(2), 75 – 86, 2006.

[10] R. Craig: *Directed versus independent discovery of established relations*. Journal of Educational Psychology, 47, 223-235, 1956.

[11] A. Speck, L. Ruzhitskaya: *Guided Versus Unguided Learning: Which One To Choose?* American

Astronomical Society, AAS Meeting #217, #405.06; Bulletin of the American Astronomical Society, Vol. 43, 2011.

[12] C. Reigeluth, J. Moore: *Cognitive Education and the Cognitive Domain*, in: C. Reigeluth (Ed.), Instructional-Design Theories and Models: A New Paradigm of Instructional Theory. (2nd Ed). Lawrence Erlbaum Associates, Hillsdale, NJ, 51–68, 1999.

[13] S. Frizell, R. Hubscher: *Using Design Patterns to Support E-Learning Design*, in: Instructional Design: Concepts, Methodologies, Tools and Applications. IGI Global, 114–134, 2011.

[14] Cognitive Design Solutions, Inc. *Instructional Design*, Available online at http://www.cognitive designsolutions.com/Instruction/InstructionalDesign .htm (accessed January 2013), 2003.

[15] D. Wiley: *Learning Object Design and Sequencing Theory*, Phd Dissertation, Department of Instructional Psychology and Technology, Brigham Young University, Available online at http://opencontent.org/docs/dissertation.pdf (accessed January 2013), 2000.

[16] M. C. Rogriguez, M. Derntl, L. Botturi: *Visual instructional design languages*. Journal of Visual Languages & Computing 21, 311–312, 2010.

[17] *The Pedagogical Patterns Project*. Available online at http://www.pedagogicalpatterns.org (accessed January 2013), 2001.

[18] C. O. De Moura Filho, A. Derycke: *Pedagogical Patterns and Learning Design: When Two Worlds Cooperate*, in Proceedings of the UNFOLD-PROLEARN Joint Workshop. Valkenburg, The Netherlands, 2008.

[19] IMS Global Learning Consortium: *IMS Learning Design Specification*, Available online at http://www.imsglobal.org/learningdesign (accessed January 2013).

[20] D. Ljubojevic, D. Laurillard: *A theoretical approach to distillation of pedagogical patterns from practice to enable transfer and reuse of good teaching*, in Proceedings of the 2010 European LAMS & Learning Design Conference, 2010.

[21] M. Derntl: *The Person-Centered e-Learning Pattern Repository: Design for Reuse and Extensibility*, in Proceedings of the 2004 ACM Symposium on Applied Computing. Nicosia, Cyprus, 916–923, 2004.

[22] AUTC: *Learning Designs*, Available online at http://www.learningdesigns.uow.edu.au (accessed January 2013), 2003.

[23] Y. Hayashi, J. Bourdeau, R. Mizoguchi: *Using Ontological Engineering to Organize Learning/Instructional Theories and Build a Theory-Aware Authoring System*, I. J. Artificial Intelligence in Education 19, 211–252, 2009.

[24] LAMS Foundation: *Activity Planner*, Available online at http://wiki.lamsfoundation.org/display/ planner/Activity+Planner (accessed January 2013), 2012.

[25] G. Savić, M. Segedinac, Z. Konjović: *Automatic Generation of E-Courses Based on Explicit Representation of Instructional Design*. Computer Science and Information Systems 9, 839 – 869, 2012.

[26] M. Segedinac, G. Savic, Z. Konjovic: *Knowledge Representation Framework for Curriculum Development*, in KEOD 2010 - Proceedings of the International Conference on Knowledge Engineering and Ontology Development, Valencia, Spain, 327–330, 2010.

[27] H. G. K. Hummel, F. Paas, E. J. R. Koper: *Cueing for transfer in multimedia programmes: process worksheets vs. worked-out examples*, Journal of Computer Assisted Learning 20, 387–397, 2004.

[28] The MathWorks: *MATLAB*, product home page: http://www.mathworks.com/products/matlab (accessed January 2013), 2013.

Acknowledgments

Results presented in this paper are part of the research conducted within the Grant No. III-47003, Ministry of Education, Science and Technological Development of the Republic of Serbia.