Deeper insight into graph theory using multimedia applications

Eva Milkova, Andrea Sevcikova

Abstract —The aim of subjects dealing with graph theory and combinatorial optimization is above all to develop and deepen students' capacity for logical and algorithmic thinking. The ability to form images in mind is also supported within these subjects. Students learn how to describe various situations with the aid of graphs, solve the given problem expressed by the graph, and translate the solution back into the initial situation. Research concerning learning style preferences of our students indicates that most of them are visual learners. Hence it is efficient to enhance teaching and learning process using various multimedia applications. In this paper is introduced a program dealing with objects appropriate to the graph theory and combinatorial optimization course subject matter, and which is created on a script given by the teacher with regard to students' needs.

Keywords Graph theory, combinatorial optimization, multimedia application; visualization.

INTRODUCTION

MATHEMATICAL visualization positively affect mathematical reasoning. "Students need images and visualization in addition to words. Science learning is about creating images in mind, and teaching should support such image formation" [1]. According [2] mathematical visualization it is not merely 'math appreciation through pictures' - a superficial substitute for understanding. It rather supplies depth and meaning to understanding, serving as a reliable guide to problem solving, and inspiring creative discoveries. In order to achieve this understanding, however, it can't be isolated from the rest of mathematics; symbolical, numerical and visual representations of ideas must be formulated and connected.

Graph theory and combinatorial optimization, the areas located at the intersection of applied mathematics and operations research, belong to fundamental parts of computer science. For the reasons that students could get a deeper insight into a problem and were able to solve it using a suitable algorithm, it is necessary to ensure especially in the instruction of subjects dealing with these areas that the students are first informed about the necessary graph-concepts and they thoroughly understand their properties (corresponding theorems and their proofs). There are various pedagogical ways to achieve this.

Eva Milková is full professor and supervisor of Andrea Sevcikova. She works at the Department of Informatics, Faculty of Science, University of Hradec Kralove (Hradec Kralove 500 38, Rokitanskeho 62, Czech Republic, e-mail: eva.milkova@uhk.cz; http://lide.uhk.cz/prf/ucitel/milkoev1/).

Andrea Sevcikova is Ph.D. student at University of Hradec Kralove, Faculty of Education, Hradec Kralove 500 38, Rokitanskeho 62, Czech Republic, andrea.sevcikova@uhk.cz.

Our approach is based on the following basic teaching principles that we have been applying in our teaching for many years.

When starting an explanation of new subject matter, we introduce a particular problem with a real life example, logical task or puzzle as a prototype of the explained concept or algorithm and properly discuss suitable graph-representation of a problem.

We examine each problem from more than one point of view and discuss various approaches to the given problem solution with respect to the already explained subject matter. Using the constructed knowledge and suitable modification of the problem solution we proceed to new subject matter.

Demonstration and visualization using suitable multimedia applications make the subject much clearer and comprehensible. In addition to words we visualize the particular issue as well as it is possible.

We thoroughly practice the explained topic on various examples, discuss students' own examples describing the topic and encourage them to solve similar examples.

Intelligent learning system has been focusing on acquiring and building the student´s knowledge, experience, to minimize their weaknesses and boost their strengths. It allows flexibility in teaching methods, achieving many of the same benefits as one-on-one instruction. (cf. [3])

With respect to above mentioned principles, in the paper we present the multimedia program GrAlg enabling visual representation of basic graph-concepts, graph-algorithms and proofs of theorems. The program is available on http://lide.uhk.cz/prf/ucitel/milkoev1/en_index.htm, the page GRAFALG / Lectures.

PRELIMINARIES

The main aim of the subjects dealing with the graph theory and combinatorial optimization is to develop and deepen students' capacity for logical and algorithmic thinking.

After learning basic graph concepts students are usually introduced to more complex algorithms with, at first a polynomial time complexity. At this point we must not forget to mention that although there are usually more methods which can be used to solve the same graph-problem, by using effective modifications of one algorithm other methods of solving various other tasks can be devised.

As seen from the above mentioned teaching principles, in our instruction we properly apply both methods to the student´s knowledge development; *rule-based reasoning* when we introduce students to the theoretical background (definitions, theorems and their proofs) and *case-based reasoning* when working with graph algorithms.

“In case-based reasoning, new problems are approached by remembering old similar ones and moving forward from there. Situations are interpreted by comparing and contrasting them with previous similar situations. And learning happens as part of the process of integrating a new case into memory” [4].

The aim of our efforts is to achieve that students were able, based on a thorough understanding of theoretical knowledge, to see relationships between algorithms, to perceive what have the discussed algorithms in common and what is different, and to be able to get algorithms solving other problems by means of certain algorithm modification. Would be able to describe various given situations with the aid of graphs, solve the problem expressed by the graph, and translate the solution back into the initial situation.

Let us discuss our approach in more detail.

A. Introduction to a topic using a prototype

Student engagement is crucial for successful education. Practical tasks attract students to know more about the explained subject matter and to apply gained knowledge (cf. [5]). If an interesting enjoyable task is assessed to each topic, students recall the explained subject matter much better and their engagement progresses when looking for similar examples.

When starting an explanation of new subject matter, a particular problem with a real life example, logical task or puzzle is introduced as a prototype of the explained concept or algorithm and suitable graph-representation of a problem is discussed.

Example

As an example let us illustrate the following puzzle *Pins* which can be used as a prototype of an important basic graph theory concept isomorphism.

Pins

In the garden there are 7 pins, bordering parts of beds, connected by strings in the way given on Fig. 1. Someone deliberately ignored them, changed their positions as it is shown on Fig. 2. The task is to find the initial position of pins (the initial order of pins).

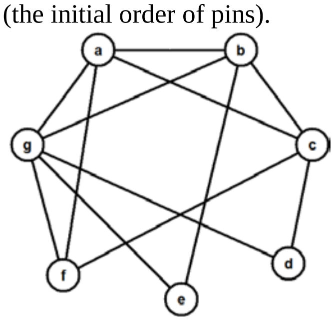


Fig. 1. Initial position of pins

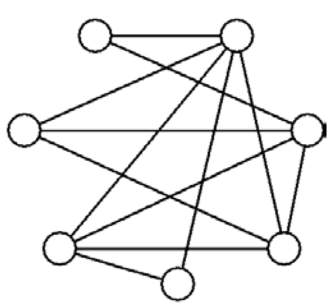


Fig. 2. Changed position of pins

This puzzle captures well the essence of the concept of isomorphism, which can be simply described as follows:

“Two graphs G and G’ are considered identical (or equal) if they have the same set of vertices and the same set of edges. But if two graphs differ „only“ by names of their vertices and edges and have the same „structure“ this is captured by the notion of isomorphism” [6].

Solution of the puzzle is introduced in the section Multimedia application.

B. Teaching in contexts

On the one hand, usually, there are more methods which can be used for solving the same graph-problem, while on the other hand, using effective modifications of one algorithm, we can devise methods of solving various other tasks.

To get deeper insight into each problem and entirely understand it we consider as very important to examine each concept and problem from more than one point of view and discuss various approaches to the given problem solution with respect to the already explained subject matter.

In the introductory course dealing with graph theory and graph algorithms we explain and discuss only the algorithms on simple undirected graphs (cf. [7] - [10]). In Fig. 3 the graph whose edges illustrate the interactions between these algorithms is presented. Numbers of vertices indicate the order in which the algorithms are gradually discussed in the lectures.

Legend to the figure Fig. 3

1. MST = Minimum spanning tree problem,
2. Tarry = Tarry’s approach to maze problem,
3. Trémaux = Trémaux’s approach to maze problem,
4. E-J = Edmonds-Johnson’s approach to maze problem,
5. Eulerian Trail = algorithm determining an Eulerian trail,
6. Trails = algorithm determining a minimum trail cover,
7. Train = algorithm solving the Chinese Postman Problem,
8. BFS = Breadth-First-Search,
9. DFS = Depth-First-Search,
10. Components = algorithm determining components,
    Cut Edge = algorithm determining if an edge is/isn’t cut edge,
11. Shortest Path = algorithm determining the length (according number of edges) of the shortest path between two vertices,
    Circles = algorithms determining circles with given properties (in more detail - see [11]),
12. Cut Vertex = algorithm determining if a vertex is/isn’t cut vertex,
    Blocks = algorithm determining blocks,
13. Dijkstra = Dijkstra’s algorithm.

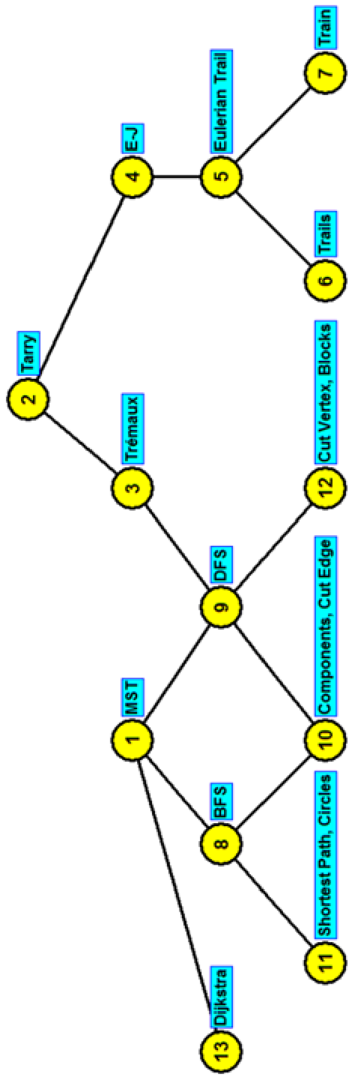


Fig. 3. Mutual relation among given graph algorithms

Example

As an example let us illustrate a possible formulation (cf. [9]) of Jarník's approach [12] to the MST problem and its modification to BFS as well as to DFS algorithm.

Jarník's algorithm solving the Minimum Spanning Tree Problem

1. Initially all vertices and edges of a given edge weighted undirected graph *G* are uncoloured. Let us choose any single vertex and suppose it to be a trivial blue tree.

2. At each of $(n - 1)$ steps, colour the minimum-weight uncoloured edge, having one vertex in the blue tree and the other not, blue. (In case, there are more such edges, choose any of them.)

3. The blue coloured edges form a minimum spanning tree.

Now let us imagine a connected undirected graph with all edges having the same weight and let us trace the Jarník's method for gaining the minimum spanning tree on this graph. One can see that at each step (step 2) an arbitrary edge, having one vertex in the blue tree and the other not, is coloured blue. A consecutive adding vertices into the blue tree can be understood as a consecutive search of them. Hence, to get either the Breadth-First Search or Depth-First Search algorithm for consecutive search of all vertices of the given connected undirected graph G, we simply modify step 2 of Jarník's method in the following way.

Breadth-First Search: At each of (n - 1) steps we choose from the uncoloured edges, having one vertex in the blue tree and the other not, such an edge having the end-vertex being added to the blue tree as the first of all in blue tree vertices belonging to the mentioned uncoloured edges and colour it blue. (Remark: To identify such an end-vertex we store vertices adding into the blue tree in the data structure queue.)

Depth-First Search: At each of (n - 1) steps we choose from the uncoloured edges, having one vertex in the blue tree and the other not, such an edge having the end-vertex being added to the blue tree as the last of all in blue tree vertices belonging to the mentioned uncoloured edges and colour it blue. (Remark: To identify such an end-vertex we store vertices adding into the blue tree in the data structure stack.)

C. Visualization

"Visual attributes of a problem or situation are the primary, and most influential, connection to meaning and understanding. Visualization used before symbolic development increases the likelihood that students will remember the mathematical concept being taught. Mathematics reasoning both takes from and gives the other parts of the mind. Thanks to graphs, we primates grasp mathematics with our eyes and our mind's eye" [13].

"The use of visual models such as diagrams, sketches and animations in mathematics education has become an area of renewed interest. This interest may stem from the fact that visual aids have always played an important role in the understanding of various mathematical concepts. In addition, suitable diagram or image can be used as visual proof of some mathematical characteristic or theorem" [14].

From research done in last years at our university (see e.g. [15], [16]) most of our students are visual learners. There are several multimedia applications used as a useful support of various subjects at our university. For the subjects dealing with the graph theory and combinatorial optimization was created GrAlg program that is closer presented in the following section.

D. Practice and revise

The explained topic is thoroughly practiced and students' own examples describing the topic are discussed. Moreover, practical and enjoyable examples serve very well as good tool for finding out if students are able to describe a given task with the aid of graphs, i.e. find a graph-representation of the task, solve it and translate the solution back into the initial situation. When solving tasks using graph theory knowledge, it isn't always easy to find immediately the needed graph-representation.

Students are provided with teaching text [17], that is, just as the teaching of the subject itself, designed to thoroughly explain the various discussed concepts and theorems and indicate extensive practical use of lectured matter.

Courses in Moodle virtual learning environment supporting subjects dealing with the graph theory and combinatorial optimization contain electronic version of the text book, GrAlg program, a detailed plan of lectures, samples of credit and exam tests. Presentations and graphs used in the lecture together with problem statements of exercises solved in the lesson are placed in the appropriate course-module.

Multimedia application GrAlg

“Technology and social media have not only become a part of school life, but have also begun to have a significant effect on the forms and methods of teaching and learning” [18].

There have been already developed a lot of multimedia tools for subjects dealing with graph theory. Let us mention for example MatrixPro [19], Graph Magics [20], Applet – Graph theory applet [21], Graph Drawing in Education [22], Jive [23].

Many small tools originated also as student work. In this way the students receive study material closely corresponding with their reasoning. Student-author forms the content of the application, according to teacher’s documents and guidelines, but on the other hand he/she formally captures the visual sensibilities of their fellow-students (cf. [24]).

At our university one of the most important applications dealing with objects appropriate to the graph theory and combinatorial optimization course subject matter created on a script given by the teacher with regard to students’ needs is the GrAlg program [25].

A. GrAlg program – brief description

The program is created in the Delphi environment and its main purpose is the easy creation and modification of graphs and the possibility to emphasize with colours basic graph-concepts and graph algorithms on graphs created within the program.

There are the following main possibilities of the program:

The program enables the creation of a new graph represented by figure, editing it, saving graph in the program, in its matrix representation and also saving graph in bmp format.

It also makes it possible to display some graph properties of the given graph.

The program enables to add colour to vertices and edges, to add text next vertices, and to change positions of vertices and edges by “drop and draw a vertex (an edge respectively)”.

The program allows the user to open more than one window (see Fig. 4) so that two (or more) objects or algorithms can be compared at once. Window size can be adjusted as needed.

In the GrAlg program there is the option to run programs visualizing all of the subjects explained algorithms in a way from which the whole process and used data structures can clearly be seen.

Example

Let us present the use of GrAlg program in solving *Pins* puzzle (see section Preliminaries).

Fig. 2 represents a graph G_1 with the set of vertices {a, b, c, d, e, f}. We will create from the object in Fig. 2, which shows the situation after the pins were dislocated, graph G_2 with the set of vertices {1, 2, 3, 4, 5, 6}. To solve the *Pins* task involves finding an isomorphism of graphs G_1 and G_2.

Fig. 4 illustrates the procedure in which to search for isomorphic vertices, we use the option to add text next to vertices.

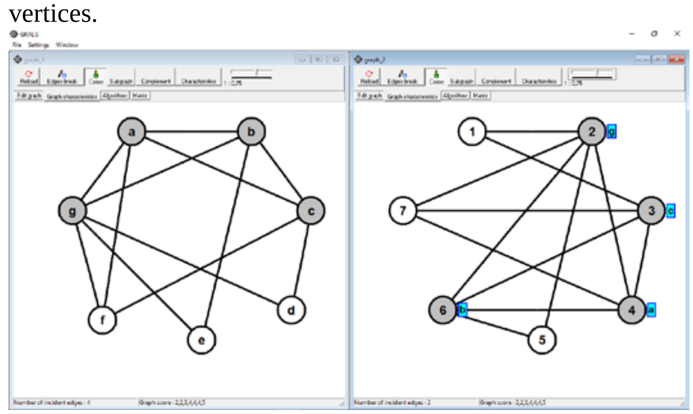


Fig. 4. Two open windows of the GrAlg program with a partial solution of the task *Pins*

Another method of finding the isomorphism can be drawing graph G_2 in a shape of graph G_1 by using the option to change positions of the vertices, see Fig. 5.

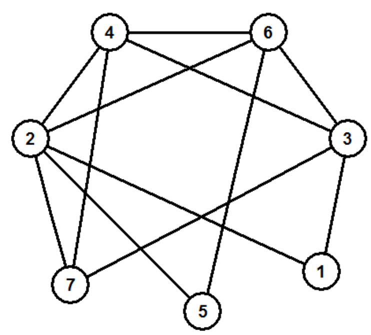


Fig. 5. Shift of pins

Both these techniques lead to the same solution of the problem, see Fig. 6.

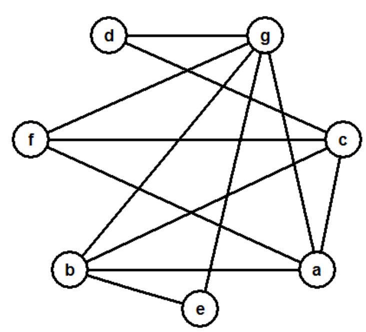


Fig. 6. Result of the task *Pins*

B. GrAlg program – advantages from the teacher point of view

The GrAlg program enables the teacher to complete his/her explanation within lectures in such a way that the topic is more comprehensible; the possibility to use colours allows the teacher to emphasize needed objects and relations, the option to open more than one window enables him/her to explain the problem from more points of view and show mutual relations among used concepts and algorithms. The possibility to save each created graph in bmt format allows him/her easy insertion of needed graphs into the study material (all graphs in this paper were prepared with the program) and thus saves his/her time when preparing text material and presentations.

C. GrAlg program – advantages from the student point of view

Using the GrAlg program students can revise subject-matter within the area of graph theory and more deeply understand it.

They can use not only graphs prepared by the teacher but also graphs created by themselves, explore the properties of these graphs and monitor the behaviour of algorithms.

The possibility to open more than one window enables them to follow mutual relations among used concepts and algorithms.

The option "Save Graph in bmp format" enables them easy creation needed graphs for their tasks (texts and/or presentations) where they describe various practical situations with the aid of graphs and solve the given problem.

D. GrAlg program – visualization of mathematical proofs

According to student's opinion subjects dealing with the graph theory and combinatorial optimization belong to most interesting however also most difficult ones thanks to the fact that thorough understanding to mathematical proofs explained within these areas is demanded of students. One of the reasons is certainly lower mathematical knowledge and skills of Czech secondary school students in recent years, which is also evident in poorer understanding of other mathematical subjects.

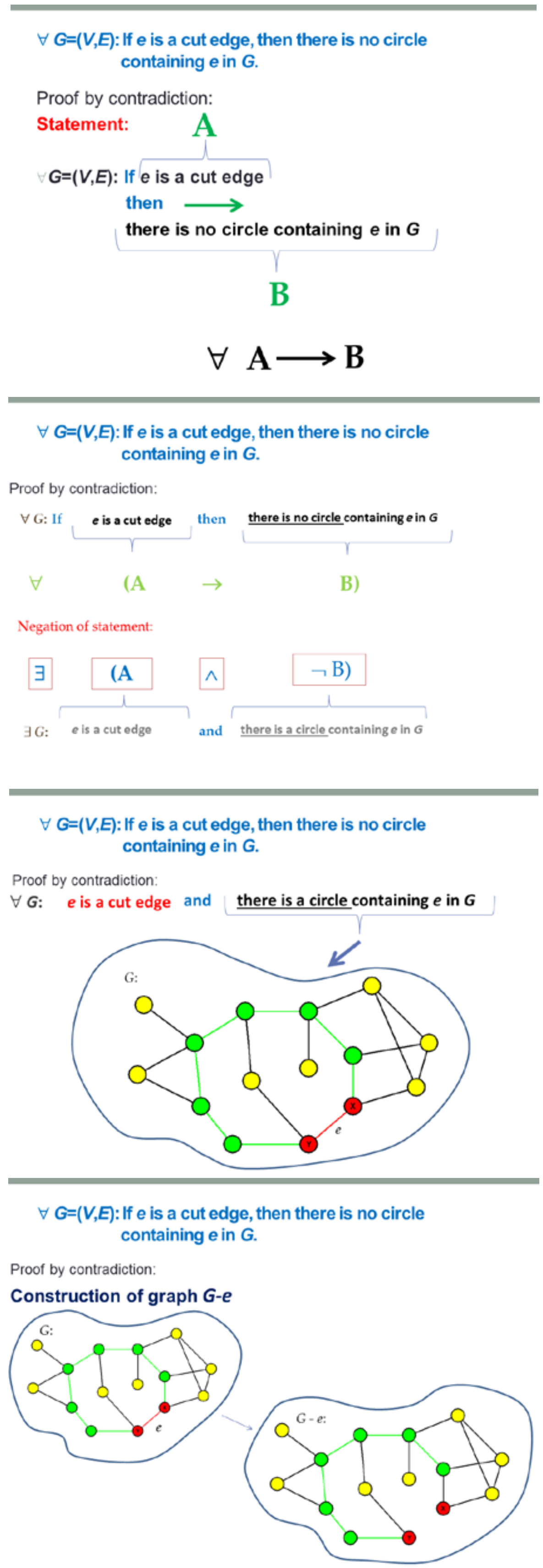
To help students in this direction we prepare various presentations visualizing proofs of explained theorems. Appropriate graphs used in presentations are prepared within GrAlg program.

Example

On Fig. 7 there are Power Point screens visualizing a proof to the following statement.

Statement:

Let $G = (V, E)$ be a graph and e an edge of G. If e is a cut edge, then there is no circle containing e in G.



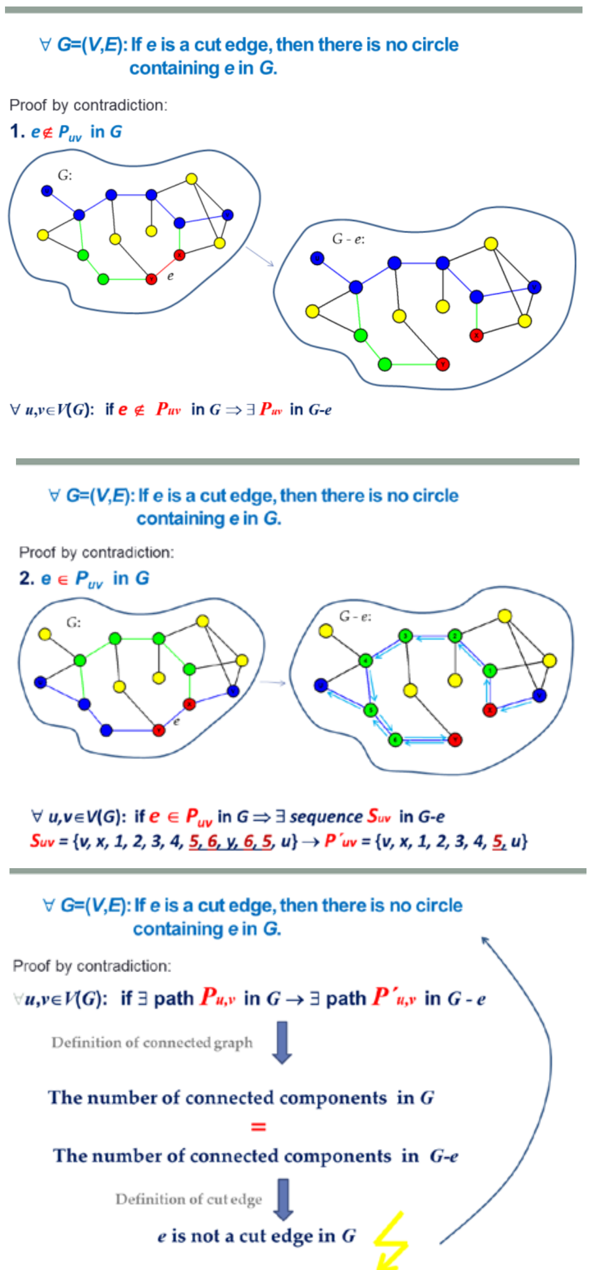


Fig. 7. Visual Proof

CONCLUSION AND FUTURE WORK

Mathematics is one of the oldest sciences however the area known as Combinatorial Optimization close connected with Graph Theory and Computer Science is quite young. Modern technology provides access to education and learning in the areas (and not only in them) in a modern way attractive for students. (cf. [26], [27])

In the paper that is enhanced version of the paper published by APSAC conference 2016 we have presented our teaching principles together with intelligent multimedia application GrAlg, a useful complement of the subjects dealing with the graph theory and combinatorial optimization. This program which visualises the discussed concepts and graph algorithms, and which enables their comparison, gives students the opportunity to get deeper insight into the subject matter and to enhance their facility to solve everyday life practical situations. Students gain many useful ideas and inspiration for their own solutions to tasks within various research areas.

At present we are dealing with a research concerning visualization of theorems and their proofs explained within the above mentioned courses. Instead of Power Point presentations supporting this field (see the previous section) a multimedia interactive application under our scenario has been developed.

ACKNOWLEDGMENT

This research has been supported by specific research project of the University of Hradec Kralove, both Faculty of Science and Faculty of Education in 2016.

REFERENCES

[1] Williams, R. (2005) e-Learning Strategy: What's in the Blend? Proceedings of the 4th Europ. Conference on e-Learning, Amsterdam, Nederland, ACL, UK, pp. 245-251.
[2] Zimmermann, W., & Cunningham, S. (1991). What is Mathematical Visualization? In Zimmermann, W. & Cunningham, S. (Eds), Visualization in Teaching and Learning Mathematics, pp. 1-9.
[3] Salem, A-B. M. (2012) Intelligent Methodologies and Technologies for e-Learning. IEEE 10th International conference on Emerging eLearning Technologies and Applications, pp. 331-337.
[4] Kolodner, J. (1993) Case-Based Reasoning, Morgan Kaufnann Publisher, Inc.
[5] Prazak, P. (2015) Teaching Simulation for Management Students in LMS. Advances in Web-Based Learning -- ICWL 2015, Volume 9412 of the series Lecture Notes in Computer Science, 121-130.
[6] Matoušek, J., & Nešetřil, J. (1998) Invitation to Discrete Mathematics. Oxford: Clarendon Press.
[7] Biggs, N. L., Lloyd, K. E., & Wilson, R. J. (1976) Graph theory 1736-1936. Clarendon Press, Oxford.
[8] Graham, R. L., & Hell, P. (1985) On the History of the Minimum Spanning Tree Problem. Annals of the History of Computing 7, 1, 43-57.
[9] Tarjan, R. E. (1983) Data structures and network algorithms. Ch. 6, CBMS Regional Conf., SIAM, Philadelphia.
[10] Dijkstra, E. W. (1959) A note on two problems in connection with graphs. Numer. Math.,1, 269-271.
[11] Milková, E. (2015) Breadth-First-Search Tree - Levels and Subtrees. International journal of mathematics and computers in simulation, Volume 9, 206 – 212.
[12] Jarník, V. (1930) O jistém problému minimálním (About a certain minimal problem). Práce Mor. Přírodověd. Spol.v Brně VI, (4) 57-63.
[13] Pinker, S. (1997) How the Mind Works, Norton.
[14] Spilka, R., & Popper, F. (2014) Pedagogical experiment with online visualization of mathematic models in math teaching on elementary school. International conference of education, research and innovation (ICERI 2014). Valencie: IATED, s. 901-910.
[15] El-Hmoudová, D. (2015) Assessment of individual learning style preferences with respect to the key language competences, Procedia - Social and Behavioral Sciences, Vol. 171 (2015), pp. 40–48. doi:10.1016/j.sbspro.2015.01.086
[16] Milková, E., Petránek, K. (2016) Programming courses reflecting students' aptitude testing and implementing learning style preferences research results, International journal of mathematics and computer in simulation, vol. 10, 2016, 218–225.
[17] Milková, E. (2013) Teorie grafů a grafové algoritmy (Graph Theory and Graph Algortihms). Gaudeamus, Hradec Králové, s. 123.
[18] Valjataga, T., Fiedler,S., & Laanpere,M. (2015) Re-thinking Digital Textbooks: Students as Co-authors. Advances in Web-Based Learning -- ICWL 2015, Volume 9412 of the series Lecture Notes in Computer Science, 143-151.

[19] Karavirta, V., Korhonen, A., Malmi, L. & Stalnacke, K. (2004) MatrixPro - A Tool for Demonstrating Data Structures and Algorithms Ex Tempore. Proceedings of the IEEE International Conference on Advanced Learning Technologies, ICALT '04, pages 892-893, Washington, DC, USA, IEEE Computer Society.
[20] Ciubatii, D. (2004) Graph Magics. http://www.graph-magics.com/
[21] Laffra, C. (1998) Dijkstra's Shortest Path Algorithm. http://www.dgp.toronto.edu/people/JamesStewart/270/9798s/Laffra/DijkstraApplet.html
[22] Bridgeman, S. (2013) Graph Drawing in Education. Handbook of Graph Drawing and Visualization, R. Tamassia, ed., CCRC Press.
[23] Gestwicki, P. & Jayaraman, B. (2005) Methodology and architecture of JIVE. Proceedings of the 2005 ACM Symposium on Software Visualization, SoftVis '05.
[24] Kurvits, M., Laanpere,M., & Väljataga,T. (2015) Analysis of Tools and Methods for Describing and Sharing Reusable Pedagogical Scenarios. Advances in Web-Based Learning -- ICWL 2015, Volume 9412 of the series Lecture Notes in Computer Science, 251-257.
[25] Šitina J.: Grafové algoritmy a jejich vizualizace (Graph Algorithms and their Visualization). Thesis, University of Hradec Králové, Czech Republic, (June 2010)
[26] Fülep, D. László, I. (2015) Parallel simulation methodological issues for creating 3D nanostructures. International journal of mathematics and computer in simulation, vol. 9, 2015, 222–227.
[27] Klimis Ntalianis, Nikolaos Mastorakis. (2016) Social Media Video Content Diversity Visualization. International Journal of Signal Processing, 1, pp.169-176