STUDENT RESEARCH PROJECTS IN SYSTEM DESIGN

Mark Sh. Levin
Instute for Information Transmission Problems, Russian Academy of Science
19 Bolshoj Karetny lane, Moscow, Russia

Keywords: Student projects, System design, Combinatorial optimization, Decision making, Heurisctics, Application.

Abstract: The article describes a new educational approach to support the movement from traditional educational process to research and/or engineering (design) activity in computer science/information technology, engineering, and applied mathematics.The approach was realized as faculty course for advanced senior undergraduate students(information technology).The basic educational flow is: applied problem, mathematical model, algorithm, software, computing the results, report.Twelve laboratory works support the flow above.In each work a combinatorial and/or multicriteria problem is (are) examined (e.g., multicriteria ranking, multiple choice problem, multicriteria assignment/allocation, clustering) including applied examples, algorithm(s), Matlab program(s).Thus students can obtain their skills in applied problems, models, and algorithms. In addition, each student can take into account his/her inclination, motivations, and personal goals. As a result, the student can select a part of the educational flow above to prepare a modified or new version of the part(i.e., applied problem, model, algorithm).Concurrently, students obtain a skills in composition of problems/models/algorithms to get a framework for a complex real world application. Motivated students have conducted advanced research projects and their articles were published.

1 INTRODUCTION

The movement from traditional educational process to research and/or design activity is often the critical part of many educational efforts in computer science (CS) and information technology, engineering, applied mathematics, management.

In the article our approach for the above-mentioned movement is described. Our suggestion is based on the course including a set of laboratory works in which an educational flow is realized: applied problem, mathematical model, algorithm, software, computing the results, and report (Figure 1). In each laboratory work a special combinatorial and/or multicriteria problem is examined (e.g., multicriteria ranking, multiple choice problem, assignment/allocation, clustering) including basic applied examples and algorithm(s) and often a basic Matlab program (Mathworks, Inc.). As a result, each student can understand all aspects of the problem. The examined problems are organized as a layered framework.

Thus students can obtain the skills in the applied problems, models, algorithms. In addition, each student can take into account his/her inclination, motivations, and personal goals. After that the student can select a part of the educational flow above to prepare a modified or new version of the part(i.e., applied problem, model, algorithm).Concurrently, student can obtain a skills in composition of problems/models/algorithms to get a framework of complex real world applications.

In is reasonable to point out the suggested approach supports problem formulation and structuring and the skill in this domain(in our opinion)is a crucial one.

Our material is based on the author's course on system design in Moscow Institute of Physics and Technology (State University) that was implemented in 2004...2008 (Levin, 2006b; Levin, 2007c). Generally, the course corresponds to author's books (Levin, 1998; Levin, 2006a) and articles (Levin, 1996; Levin,

Sh. Levin M. (2009).
STUDENT RESEARCH PROJECTS IN SYSTEM DESIGN.
In *Proceedings of the First International Conference on Computer Supported Education*, pages 67-72
DOI: 10.5220/0001974400670072
Copyright © SciTePress

2001; Levin, 2002; Levin, 2005; Levin, 2006b; Levin, 2007a; Levin and Danieli, 2005; Levin and Firer, 2005; Levin and Last, 2006; Levin and Nisnevich, 2001; Levin and Sokolova, 2004). In recent years, results of student research projects have been published as articles (Levin and Khodakovskii, 2007; Levin and Leus, 2007; Levin and Merzlyakov, 2008; Levin and Safonov, 2006; Levin and Vishnitskiy, 2007).

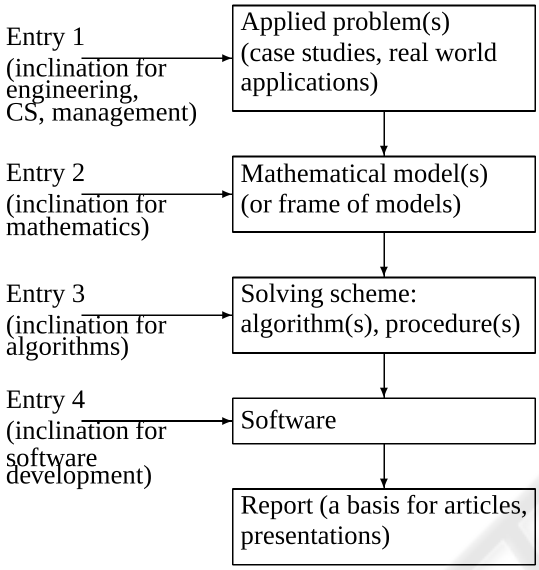

Figure 1: Educational flow-chart.

Clearly, basic books in several corresponding domains are used as well, for example:(i) combinatorial optimization (Cela, 1998; Garey and Johnson, 1979; Kelleler and Pisinger, 2004; Martello and Toth, 1900);(ii) multicriteria decision making (Fishburn, 1970; Keeny and Raiffa, 1976; Roy, 1996; Saaty, 1998; Steuer, 1986);(iii) design frameworks (Altshuller, 1984; Ayres, 1998; Jones, 1981; Zwicky, 1969);(iv) systems engineering (Sage, 1979).

The course was realized for very good educated and motivated students, but the approach may be used (as a simplified version)for other students groups as well. Note the course materials and results (including a set of published student research articles) are accessible via Internet.

2 COURSE ON SYSTEM DESIGN

The course consists of the following parts:(i) lectures,(ii) laboratory works(here good results can be prepared as papers for conferences and/or journals), and (iii) examination. Lecture materials correspond to the following:

1. Issues of systems engineering, system life cycle.

2. System analysis and multicriteria decision making including principles of system analysis, decision making framework, methods for multicriteria choice/ranking (utility function approach, methods based on revelation of Pareto-effective solutions, outranking techniques).

3. Optimization and combinatorial optimization (including basic models as follows: knapsack problem, multiple choice problem, scheduling, satisfiability problem, graph coloring, clique problem, covering problem, clustering, assignment/allocation problem, travelling salesman problem (TSP), design of spanning structures).

4. Basic design frameworks and approaches:(a) optimization approaches, multidisciplinary optimization, parameter space investigation method, morphological analysis and its modifications, evolution approaches;(b) frameworks for system improvement; and (c) modeling of system evolution/development.

5. Additional system issues: requirements engineering, system evaluation and diagnosis, maintenance,testing.

Course environment (computer class and Internet site)involves the following parts:(i) lecture notes(at course site and in computer class);(ii) program support:(a) basic software (editors, etc.),(b) network environment (Internet, etc.), and (c) Matlab environment(site of Mathworks, Inc.); and(iii) additional materials:(a) research articles;(b) support materials as examples:to design student homepages, to prepare reports on laboratory works, to prepare student articles (a set of student published articles).

Twelve laboratory works are included into the course:

Laboratory work 1.

Part 1A: Introductory work (computer environment, design of student homepages, preparation of presentations).

Part 1B: Hierarchical morphological design of a modular system (Levin, 1996; Levin, 1998; Levin, 2001; Levin, 2002; Levin, 2005; Levin, 2006a; Levin, 2008; Levin and Firer, 2005; Levin and Khodakovskii, 2007; Levin and Last, 2006; Levin and Nisnevich, 2001; Levin and Sokolova, 2004; Levin and Vishnitskiy, 2007; Ritchey, 2006).

Laboratory work 2. Multicriteria ranking (utility function approach, Pareto-based approach, outranking technique as ELECTRE).

Laboratory work 3. Multicriteria knapsack problem.

Laboratory work 4. Method of proximity to ideal point(s).

Laboratory work 5. Hierarchical clustering (Jain, 1999; Levin, 2007b).

Laboratory work 6. Multicriteria multiple choice problem (Levin and Safonov, 2006; Poladian, 2006).

Laboratory work 7. Hierarchical ordinal evaluation of composite system (hierarchical method based on integration tables) (Levin, 2006a; Levin and Danieli, 2005).

Laboratory work 8. Composite applied example: clustering and multiple choice problem.

Laboratory work 9. Assignment/allocation problem.

Laboratory work 10. Composite applied example: clustering & clustering & allocation & multiple choice problem (Levin and Merzlyakov, 2008).

Laboratory work 11. Travelling salesman problem.

Laboratory work 12. Additional work on the topic selected by him/herself (e.g., heuristics, genetic algorithms, method based on space filling curves, cross-entropy method, quadratic assignment problem, covering problems, real world applications).

In each laboratory work, students have to do the following:

(1) to understand the material: applied problem, model, solving scheme (algorithm, framework),

(2) to develop software and to test it,

(3) to prepare a numerical example (or a real world application),

(4) to compute the results, and

(5) to prepare the report.

For some basic laboratory works there are some basic Matlab programs, for example: multicriteria ranking (utility function method, revelation of Pareto-effective solutions, outranking techniques, e.g., method Electre), heuristic for knapsack problem, heuristic for multiple choice problem, hierarchical clustering (agglomerative algorithm). As a result, each student has to improve the basic Matlab program to get his/her resultant program. For more complicated (i.e., composite) laboratory works (for example: 1, 6, 8, 10) each student can combine his/her Matlab programs to get the resultant composite software. Note students hav their opportunity to choose a software environment (e.g., C, Java).

Our framework consists of the following layers:

Layer 1. Basic combinatorial problems and multicriteria decision making problems: multicriteria ranking, multicriteria knapsack problem, 'ideal point' method, and system evaluation/diagnosis.

Layer 2. Advanced models (e.g., clustering, multicriteria multiple choice problem, multicriteria assignment/allocation, TSP)

Layer 3. Composite models/procedures (as basic solving composite scheme consisting of problems/models): (i) Hierarchical Morphological Multicriteria Design (HMMD)(ranking, combinatorial synthesis) to design modular systems, (ii) design of a modular solving strategy (e.g., a partitioning/synthesis heuristic for problem), (iii) system upgrade, (iv) multistage design (two-level HMMD), (v) system evolution/forecasting, and (vi) special multistage composite framework (clustering, assignment/location, multiple choice problem), etc.

Figure 2 illustrates the corresponding framework over laboratory works.

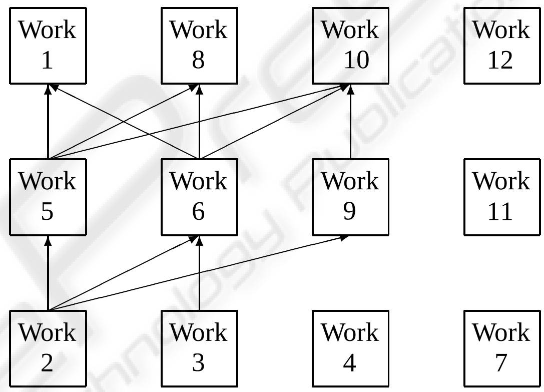

Figure 2: Framework over laboratory works.

Site of the course involves the following (http://www.mslevin.iitp.ru/SYSD.HTM):(i) description of the course, (ii) lecture notes, (iii) students lists with corresponding information on the students's results/achievements, (iv) references to student homepages (in the case of their existence),(v) references to significant bibliography and Web-sites, and (vi) references to student published articles (as examples for future research works).

3 RESEARCH PROJECTS

Composite laboratory works are used as a fundamental for student research projects. Generally, there are four possible directions for the student research: (a) real world application (an engineering/CS/management research), (b) new or modified model (research in mathematics or computer science), (c) new or modified algorithm/procedure (solving scheme) (research in mathematics or computer science), and (d) new or modified program (research in applied computer science).

As a result, each student can choose the certain kind of his/her research. This choice process is based on student inclination and/or experience. In the simplest situation, students can use a simple numerical version of an applied problem (e.g., multicritertia ranking, knapsack, multiple choice problem, assignment, TSP).

On the other hand, each student can examine a special real world application. This step is crucial for composite problems in the following laboratory works: 1, 5, 6, 7, 9, 10, 11, 12. The example list of student research projects involves the following: (1) sport applications (e.g., organization of sport events, planning of sport activity), (2) educational application (computer class, interactive educational software),(3) software development (e.g., software for modeling of signals), (4) hardware applications (e.g., computer memory), (5) communication systems (analysis of communication protocols, routing, devices for computer networks, connection of clients to network, improvement/upgrade or extension of a network),(6) sensors and wireless systems (telemetry system, wireless sensor), (7) VLSI design (planning of test processes, planning of manufacturing), (8) private electronic devices (digital camera, notebook, car, mobile phone), and (9) Web-based application (information system, Internet-based station), etc.

In the case of student inclination for algorithmic studies, it is possible to design new or modified algorithms (e.g., student works: for clustering, for TSP, for covering problems, for multicriteria multiple choice problem, for quadratic assignment problem, for graph coloring).

In the field of algorithm studies, important and useful research projects correspond to comparison of various algorithms for the same problem (e.g., for multicriteria multiple choice problem, for multicriteria assignment problem, for multicriteria Steiner tree problem).

It is reasonable to point out in laboratory work 12 some topics of students research projects are targeted to algorithms, for example: algorithms for routing in communication networks; experimental comparison of heuristics for TSP (or quadratic assignment problem): genetic algorithm, ant colony optimization, cross entropy method; algorithms based on space-filling curve; and algorithms for scheduling in wireless sensor networks. In this case, the student has to study corresponding materials (articles, books, Internet) and to conduct the algorithmic research without assistance (on his//her own).

Thus the course contains several entries for student research projects (Figure 1):

Entry 1 (Application): new real world application (on the basis of existing problem(s), model(s), algorithm(s), software).

Entry 2 (Mathematical Models):model(s) (new or modified mathematical model).

Entry 3 (Algorithms): solving scheme and/or their analysis (theoretical, experimental).

Entry 4 (Software): program.

Figure 3 illustrates basic and advanced student research project flows.

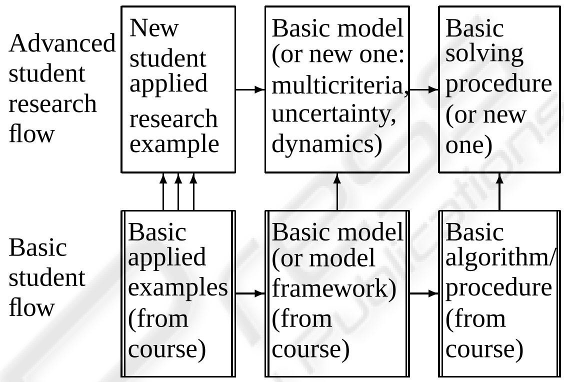

Figure 3: Basic and advanced research flows.

Figure 4 depicts typical domains for new student research applied problems where students have their experience.

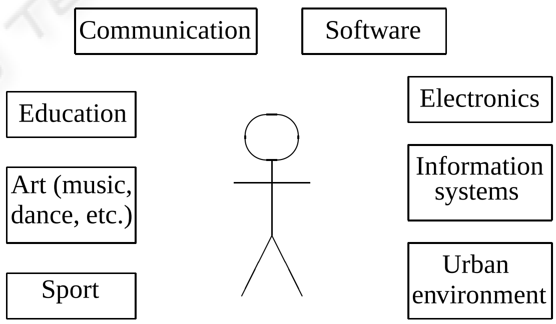

Figure 4: Domains for new applied problems.

Note some advanced student projects have been published as articles (Levin and Fimin, 2007; Levin and Khodakovskii, 2007; Levin and Leus, 2007; Levin and Safonov, 2006; Levin and Vishnitskiy, 2007; Levin and Ryabov, 2007).

4 DISCUSSION

Generally, there exist four basic levels of educational process (Nilsson, 1971): (1) by instructions, (2) by explanation, (3) by examples, and (4) by creation. Table 1 depicts correspondence of support educational materials to levels above in our course.

Table 1: Educational levels and course support.

Educational level	Support processes and materials
1.by instructions	(a) lectures, (b) books, articles, (c) Internet
2.by explanations	(a) lectures, (b) laboratory works, (c) collaboration (professor, other students)
3.by examples	(a) lectures, (b) laboratory works, (c) examples-analogues (from lectures, articles, student works, Internet, etc.)
4.by creation	materials for educational levels 1, 2, 3

Note a library of examples-analogues is a crucial support material (including previous student projects and corresponding published student articles).

Evidently, the 4th level above is the most important for student research projects, but the usage of this level is not easy for majority of students.

Our course involves three basic parts: (a) applied problem, (b) mathematical model or frame of the models, and (c) algorithm(s) and/or solving procedure. The levels of above-mentioned educational process can be targeted to a certain basic part of the course. Clearly, the examination of an applied problem is often more easy for students and this approach is the basic one in the course. Thus, students can analyze their experience and select a certain applied problem to use basic model(s) and basic algorithm(s) (or solving procedure) from the course. This way is very useful for engineering, management, and CS students because it can be based on obtained courses and skills.

On the other hand, a small part of students (about 4...10 percent) can have their inclination and motivation to mathematical models and/or algorithms. In this case they can be oriented to creation at the level(s) of models and/or algorithms.

In addition, it is reasonable to point out a prospective technique for organization of educational environment (a system *professor-student*): an educational process is based on a level i ($i = 1,2,3$), but a student is thinking that the educational process corresponds to more higher level, i.e., $> i$.
This technique is often crucial to increase the level of student creativity (i.e., creativity level of student research projects). As a result, the student will not be afraid to create.

Finally, it is necessary to point out student motivation is the main and crucial basis for successful student research projects.

5 CONCLUSIONS

In the paper, our approach to student research projects has been described. Laboratory works are used as a basis for several kinds of research projects (e.g., real-world applications, mathematical models, algorithms). As a result, each student can choose the corresponding kind of research or a combination of the research kinds. Evidently, this student choice is based on his/her inclination and/or experience and/or motivation. The educational approach above can be used for organization of team research projects as well.

It is reasonable to point out the suggested educational approach is very useful for good educated and motivated students for movement from learning stage to research/design stage.

The future research directions are the following:

(1) extension of the set of basic models by other combinatorial optimization problems (e.g., covering problem, the longest path problem) which correspond to design problems at all stages of system life cycle (including requirements engineering, system testing, maintenance, recycling);

(2) usage of problems under uncertainty (e.g., probabilistic and/or fuzzy estimates);

(3) new real world applications;

(4) usage of AI techniques;

(5) further extension of the library of student research articles; and

(6) preparation of a new course for graduate students with extended set of basic problems/models 'Combinatorial problems in system configuration design' (Levin, 2009).

REFERENCES

Altshuller, G. S. (1984). *Creativity as an Exact Science*. Gordon&Breach, London.

Ayres, R. Y. (1998). *Technological Forecasting and Long-Time Planning*. McGraw-Hill, New York.

Cela, E. (1998). *The Quadratic Assignment Problem*. Kluwer, Dordrecht.

Fishburn, P. C. (1970). *Utility Theory for Decision Making*. Wiley, New York.

Garey, M. R. and Johnson, D. S. (1979). *Computers and Intractability. The Guide to the Theory of NP-Completeness*. W.H.Freeman and Company, San Francisco.

Jain, A. K. (1999). Data clustering: a review. *ACM Computing Surveys*, 31(3):264–323.

Jones, J. C. (1981). *Design Methods*. Wiley, New York.

Keeny, R. L. and Raiffa, H. (1976). *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*. Wiley, New York.

Kelleler, H. and Pisinger, D. (2004). *Knapsack Problem*. Springer, Berlin.

Levin, M. S. (1996). Hierarchical morphological multicriteria design of decomposable systems. *Concurrent Eng.: Res. and Appl.*, 4(2):111–117.

Levin, M. S. (1998). *Combinatorial Engineering of Decomposable Systems*. Kluwer, Dordrecht.

Levin, M. S. (2001). System synthesis with morphological clique problem: fusion of subsystem evaluation decisions. *Information Fusion*, 2(3):225–237.

Levin, M. S. (2002). Towards combinatorial planning of human-computer systems. *Appl. Intell.*, 16(3):235–247.

Levin, M. S. (2005). Modular system synthesis: example for composite packaged software. *IEEE Trans. on SMC - Part C*, 35(4):544–553.

Levin, M. S. (2006a). *Composite Systems Decisions*. Springer, New York.

Levin, M. S. (2006b). Course 'system design: structural approach. In *18th Int. Conf. Design Methodology and Theory DTM2006, DETC2006-99547*.

Levin, M. S. (2007a). Combinatorial technological systems models (examples for communication system). In *2nd Int. Conf. ICSEM'07*.

Levin, M. S. (2007b). Towards hierarchical clustering. In *CSR 2007, LNCS 4649*. Springer.

Levin, M. S. (2007c). Towards teaching of system design. *Inform. Technol.&Comput. Syst.*, 2:89–94 (in Russian).

Levin, M. S. (2008). Morphological approach to electronic shopping. In *IEEE Reg. 8 Int. Conf. Sibircon-2008*. IEEE.

Levin, M. S. (2009). Combinatorial problems in system configuration design. *Autom. and Remote Control*, 70(3).

Levin, M. S. and Danieli, M. A. (2005). Hierarchical decision making framework for evaluation and improvement of composite systems. *Informatica*, 16(2):213–240.

Levin, M. S. and Fimin, A. V. (2007). Hierarchical design of fire alarm wireless sensor element. In *7th Int. Conf. CAD/CAM/PDM-2007*. Inst. of Control Problems.

Levin, M. S. and Firer, M. (2005). Hierarchical morphological design of immunoassay technology. *Computers in Biology & Medicine*, 35(3):229–245.

Levin, M. S. and Khodakovskii, I. A. (2007). Structural design of the telemetry system. *Autom. and Remote Control*, 68(9):1654–1661.

Levin, M. S. and Last, M. (2006). Design of test inputs and their sequences in multi-function system testing. *Appl. Intell.*, 25(1):105–124.

Levin, M. S. and Leus, A. V. (2007). Modular design of integrated security system. In *Int. Workshop DCCN 2007*, volume 2. Inst. for Inform. Transmission Problems.

Levin, M. S. and Merzlyakov, A. O. (2008). Composite combinatorial scheme of test planning (example for microprocessor systems). In *IEEE Reg. 8 Int. Conf. Sibircon-2008*. IEEE.

Levin, M. S. and Nisnevich, M. L. (2001). Combinatorial scheme for management of life cycle: example for concrete macrotechnology. *Intell. Manuf.*, 12(4):393–401.

Levin, M. S. and Ryabov, V. A. (2007). Morphological tree model for communication protocol (example). In *7th Int. Conf. CAD/CAM/PDM-2007*. Inst. of Control Problems.

Levin, M. S. and Safonov, A. (2006). Design and redesign of configuration for facility in communication network. *Inform. Technol.&Comput. Syst.*, 4:63–73 (in Russian).

Levin, M. S. and Sokolova, L. (2004). Modular system synthesis: example for composite packaged software. *Computer Meth. & Progr. in Biomedicine*, 73(1):3–11.

Levin, M. S. and Vishnitskiy, R. O. (2007). Towards morphological design of gsm network. *Inform. Processes*, 7(2):182–189.

Martello, S. and Toth, P. (1900). *Knapsack Problem: Algorithms and Computer Implementation*. Wiley, San New York.

Nilsson, N. J. (1971). *Problem Solving Methods in Artificial Intelligence*. McGraw-Hill, New York.

Poladian, V. (2006). Task-based adaptation for ubiquitous computing. *IEEE Trans. on SMC - Part C*, 36(3):328–340.

Ritchey, T. (2006). Problem structuring using computer-aided morphological analysis. *J. of the ORS*, 57(7):792–801.

Roy, B. (1996). *Multicriteria Methodology for Decision Aiding*. KLuwer, Dordrecht.

Saaty, T. L. (1998). *The Analytic Hierarchy Process*. McGraw-Hill, New York.

Sage, A. P., editor (1979). *Systems Engineering: Methodology and Applications*. IEEE Press, New York.

Steuer, R. L. (1986). *Multiple Criteria Optimization: Theory, Computation, and Application*. WIley, New York.

Zwicky, F. (1969). *Discovery Invention, Research Through the Morphological Approach*. McMillan, New York.