CURRICULUM DEVELOPMENT USING GRAPHS OF LEARNING OUTCOMES

T. AUVINEN
Aalto University, Department of Computer Science and Engineering
P.O.Box 15400 FI-00076 Aalto, Finland
e-mail: tapio.auvinen@aalto.fi

EXTENDED ABSTRACT

Students may not always have a clear picture of how courses in university studies are connected to each other and how they contribute to professional competences. For example, basic courses in the beginning of studies may be unmotivating because they seem abstract and unconnected to practice. The whole degree program may seem like a list of mandatory courses without much justification on why each topic must be studied.

To give students a more meaningful picture of how the courses in a university curriculum contribute to future courses and to higher-level goals, we propose a curriculum model that defines the connections between learning outcomes of different courses in detail. In our model, the learning outcomes of each course are identified, and prerequisite dependencies are specified between course outcomes instead of between whole courses. The instructor of each course specifies which topics from earlier courses must be known before the new topics can be learned. This way, the outcomes form a graph, or a network, which emphasizes the hierarchical nature of knowledge.

The outcome graph-based curriculum model offers advantages for both students and staff. Learning paths can be visualized to show students how each course contributes to the professional competences. We hypothesize that studying motivation is increased when learning outcomes are tied to higher-level goals. Students can also be given more freedom to construct a personal competence profile according to their own interests. The list of courses required to build the desired competences can be automatically calculated from the outcome graph. The dependency graph can also help staff to identify problems in the curriculum. It will reveal if a skill in a target competence cannot be developed from the available outcomes of existing courses or if a learning outcome of some course does not contribute to any professional competence or advanced course. Unnecessary overlaps or insufficient coverage of important topics can also be identified.

We are developing the outcome graph-based curriculum model in order to develop the Structural Engineering and Building Technology curriculum at the department of Civil and Structural Engineering at Aalto University. However, the proposed concept is generic and can be applied to any field where knowledge is hierarchical and courses have prerequisite connections. Our model is not in use by students yet. In this paper, we are reporting work in progress and discussing possible advantages and disadvantages of the concept.

KEYWORDS

Curriculum development, Learning outcome graph, Core curriculum analysis

1. INTRODUCTION

If university studies are to be motivating, students should have some freedom to choose courses according to their own interests. Indeed, university curricula often include elective courses, but in order to ensure a minimum level of knowledge, many mandatory courses are required. The first few years of engineering education typically have many basic courses such as mathematics and physics that students may find unmotivating because their connection to practice is not apparent [2]. It is also problematic if the curriculum is merely a list of courses that one needs to pass before graduating. That way, students do not have justification for why each course is necessary and how the skills learned from each course are going to be needed in future [1].

There is an ongoing process at the Department of Civil and Structural Engineering at Aalto University to reconstruct the Structural Engineering and Building Technology curriculum using the STOPS curriculum model developed by Paavola and Hartikainen [6]. We are currently developing a software that facilitates the implementation of the model and allows students to construct personal study plans.

In the STOPS model, the learning outcomes of each course are first identified, i.e. what a student knows after completing each course. Next, the prerequisite dependencies between the outcomes are identified, i.e. which skills from previous courses a student must acquire before entering an advanced course. Third, high-level competence areas are identified. In civil engineering, these include concrete structures, steel structures, construction economics, etc. The competence definitions consist of learning outcomes that students should have upon completion of their studies. Learning outcomes of the competences have prerequisite links to the outcomes of courses to indicate how each competence is built from skills taught on individual courses. The model is illustrated in Figure 1.

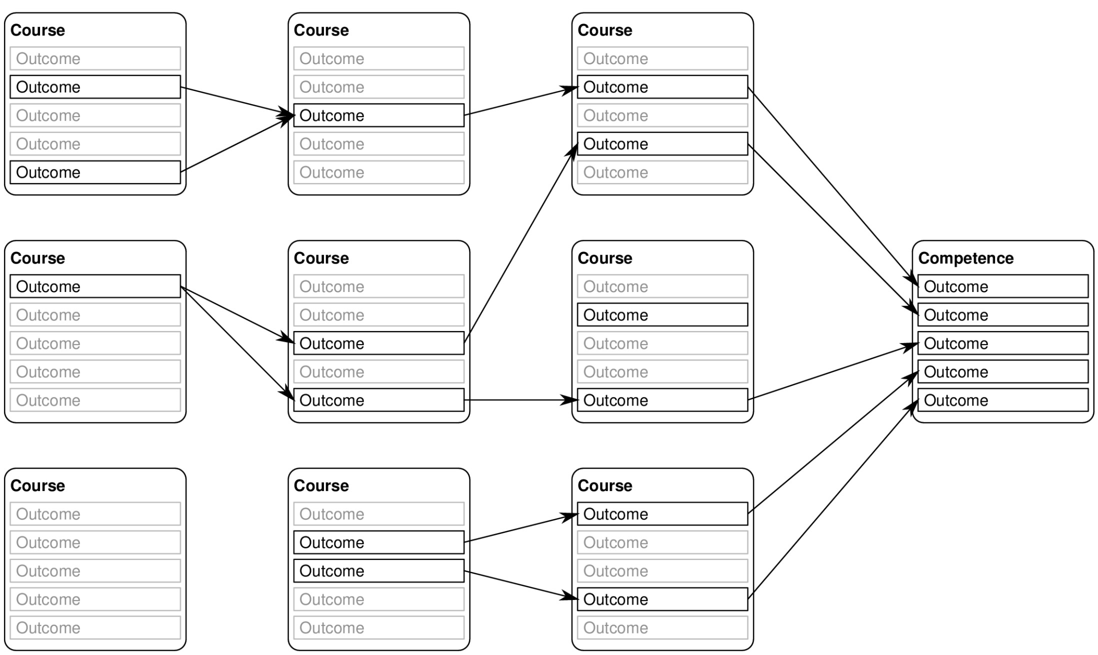

Figure 1: Example of the curriculum structure. A competence consists of outcomes that have prerequisite dependencies to course outcomes. Course outcomes can, in turn, depend on other course outcomes. From the graph, students can easily see how each course contributes to the competence.

Students can construct personal study plans by choosing competences according to their personal interests. The list of required courses is determined by following the prerequisite links from the selected competences all the way to basic courses. This way, the degree is no longer an arbitrary list of courses, but students can see how the skills taught on each course contribute to the desired competences.

The goal of this work is to increase study motivation by connecting the learning outcomes of each course to each professional goal. The model can also help instructors to improve the curriculum and courses by revealing problem areas. For example, prerequisite chains that are too long to make it impossible to graduate in a reasonable time can be identified.

2. RELATED WORK

2.1 Core curriculum analysis

Our curriculum model is inspired by a curriculum development method called *core curriculum analysis*. In core curriculum analysis, learning outcomes are categorized into content that every student *must know* in order to graduate, supplementary content that they *should know*, and specific content that is *nice to know* because it offers a deeper knowledge, but cannot be required from all students [9]. Categorization of content can typically be done by a panel consisting of experts from the industry and education.

Lindblom-Ylänne et al. [9] list numerous ways to use core curriculum information in curriculum design. For example, the "*must know"* content can be located in bachelor level studies and the supplementary content in different master's programs. The categorization can also be used in assessment. The "*must know"* core content can be required for passing a course, while supplementary and specific matter are required for higher grades.

Core curriculum analysis does not consider how courses are connected to each other. It can be helpful for determining what should be taught in courses but does not address the problem of conveying a meaningful picture of the structure of the curriculum to students. Also, core curriculum analysis does not take into account that different skills can be important for different students. Often, the importance of a skill depends on what the student is going to study in the future.

Core curriculum analysis should not be confused with outcome-based education (OBE). Spady describes OBE as an educational paradigm which ensures that every student achieves the same minimum outcomes but not necessarily at the same speed [14]. He specifically insists that OBE is not an existing curriculum with outcomes added on top [15]. A key idea in OBE is that students are not allowed to move forward before sufficient mastery is acquired, whereas in traditional education, it is possible to pass a course with a bad grade. Typically, OBE is associated with more flexibility in demonstrating the acquired mastery, and students can be given more time to improve if necessary. OBE began in pre-university education but has later been applied also in higher education, especially medical education [3]. OBE is suitable for fields where it is essential that students cannot be allowed to graduate before mastering a specific set of minimum skills.

OBE is more an assessment and instruction paradigm than curriculum design paradigm. Our curriculum model, on the other hand, does not define what educational practices should be used in courses. It is more a descriptive model that aims to help students to get a picture of the structure of their studies. In practice, defining the learning outcomes probably affects educational practices as well. The outcomes should, for example, be tied to what is being measured in exams.

2.2 Intelligent tutoring systems

Graph-based curriculum models have previously been used in intelligent tutoring systems (ITS). ITS are computer-based instructional systems that dynamically adapt the content that is delivered to students based on what they have already learned and where they have made mistakes [10]. They contain models of instructional content that allow learning material to be generated “on the fly” and give students more control over learning compared to static material such as books or web pages.

Nkambou et al. [11] have developed a subject-matter model and authoring tools for course and curriculum construction in intelligent tutoring systems. In their concept, a curriculum is represented by three models: capability model, instructional objectives model and pedagogical resources model. The capability model describes the domain knowledge, i.e. what content should be taught to students and what are the learning objectives. The instructional objectives model describes the behavior that the student must demonstrate following the learning process, i.e. the learning outcomes and assessment standards. It also defines the prerequisite relationships between capabilities. The pedagogical resources model connects instructional objectives to the learning resources necessary for acquiring the capabilities. Their ITS is designed to automatically guide students during the learning process. If the system notices that a student has trouble solving one type of exercise, it can offer more learning materials that are connected to that area of knowledge. On the other hand, if a student demonstrates good mastery of a topic, redundant material can be skipped.

Hwang [7] proposes a conceptual map model that describes how concepts and knowledge are accumulated to form higher level concepts. For example, multiplication and subtraction are required for someone to be able to understand division. Students are given tests, and by using an *item test relationship table*, it is possible to estimate which concepts a student has understood correctly. The system is able to give students a detailed list of subjects that require more practice. Hwang argues that traditional tests and exams that give student a numerical score are not equally helpful because multiple learning outcomes are assigned to the same grade.

Models aimed for intelligent tutoring systems must contain very detailed descriptions of the contents of a course so that learning materials can be automatically delivered to students. The objective of our system, however, is to describe a university curriculum on a higher level so that students can choose courses and plan studies. The aim is not to create an online learning environment but to leave implementation of individual courses up to the teacher of each course. Ideas from intelligent tutoring systems can, however, be adapted to curriculum design if the level of detail and granularity of the models are adjusted.

2.3 Curriculum visualization tools

Sommaruga and Catenazzi have made an application for the visualization of curricula as 3D graphics [13]. Departments, degree programs and semesters are rendered as regions in space and courses as boxes of different sizes. The user can navigate in the 3D space and zoom into details. Numerical properties of courses, such as credits and duration, determine the dimensions of the course boxes so that the user can easily discern the characteristics of different courses. The system focuses on the visualization of whole degree programs, without going into details such as learning outcomes of courses or the prerequisite relations between courses.

Gestwicki [5] has made the CurricViz application, which automatically generates visualizations of curricula as directed graphs. The study order of courses can easily be read from the graph. The system does not show details of the learning outcomes of the courses. The user can see that a course is a prerequisite of another course but not which outcomes specifically are important.

Zucker [16] has made the ViCurriAS application, which allows staff to construct a curriculum map, i.e. a graph of courses that are connected by prerequisite dependencies. The program allows instructors to examine how changes in courses affect the whole curriculum. The program can also be used for student counseling as it allows tracking their progress. Passed, current and upcoming courses are rendered in different colors so that the progress of studies can be easily seen. The program does not, however, show separate learning outcomes of the courses but deals with prerequisite dependencies at a course level.

Kabicher and Motschnig-Pitrik [8] have made a wiki-based curriculum planning tool that automatically creates visualizations of module dependencies. The application is meant for participatory curriculum design so that instructors can collaboratively plan how contents are divided between courses. However, the visualizations are not adapted for each student separately. Also, the application visualizes the dependencies of whole courses instead of separate learning outcomes.

In summary, it is common for existing curriculum visualization tools to show dependencies of whole courses instead of showing which learning outcomes specifically are connected. In this way, a student on a basic course does not know where is each skill going to be needed in future courses or how do skills contribute to professional competences. Also, the existing tools do not allow students to construct personal study plans. The visualizations are used as ways to improve current textual curriculum descriptions and course lists rather than providing new ways to design curricula.

3. OUTCOME GRAPH-BASED CURRICULUM MODEL

Paavola and Hartikainen [6] propose a curriculum model where the learning outcomes of each course are specified, and prerequisite dependencies are defined between the outcomes instead of whole courses. The outcomes form a directed acyclic graph where the vertices represent learning outcomes and the edges represent prerequisite dependencies between them. For a chosen outcome, it is possible to follow the prerequisite connections and collect the list of courses that are required to reach the outcome.

Figure 2 illustrates the learning outcomes and some of their dependencies in the *Bridges and Foundation Structures* course. It can be seen that the highlighted outcome "*Understands the dimensioning principles of pile foundations and can determine the forces on piles"* requires several tools from *Mathematics 1* and *Structural engineering* courses, and in turn, acts as a prerequisite for several outcomes of the *Bridges, General course*. Now, if a student is aiming to reach the outcome *"Can determine and sketch the principal dimensions for bridges and select suitable foundation types"*, it is possible to visualize how that outcome is built from the skills taught on earlier courses beginning from basic mathematics.

One of the goals in the new curriculum model is to offer students more freedom to construct personal study plans. The model contains professional competences for students to choose from and build their own competence profiles. Competences are divided into three levels (I, II, III), each consisting of outcomes that define what a student should know upon graduation. The competence outcomes, in turn, have prerequisite dependencies to course outcomes. Table 1 shows a working draft of the competences in the Structural Engineering and Building Technology curriculum.

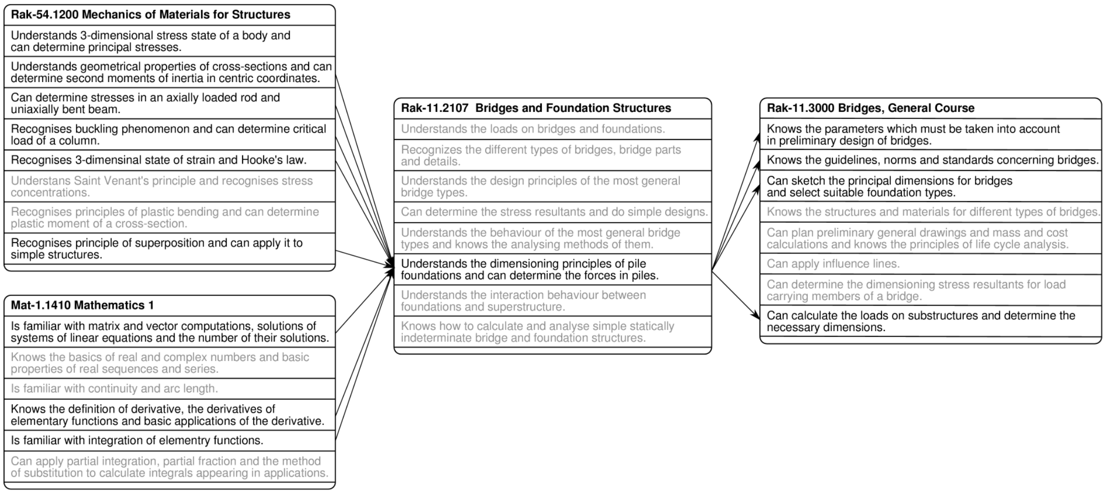

Figure 2: Connections from one outcome of the Bridges and Foundation Structures course to prerequisite and advanced courses.

Table 1: Example competences

Structural Analysis
Structural Engineering – Bridges and other Infrastructural Constructions
Structural Engineering – Concrete Structures Structural Engineering – Steel Structures
Structural Engineering – Timber Structures
Structural Engineering – Repair of Buildings
Building Materials Technology
Building Physics – Heat and Moisture Engineering
Construction Economics and Management
Building Services Engineering

The exact rules for choosing the competences are not finalized as of this writing, but the current idea is that level I is mandatory for all students and gives a basic understanding of the whole field. In addition, each student has to choose one or two level III competences and enough level II competences to reach 300 credit points.

When a student adds a competence goal to the study plan, the list of courses that are required to build the competence is calculated by following the prerequisite links. Figure 3 illustrates how the Bridges II competence consists of six outcomes that depend on the learning outcomes of three courses. These, in turn, depend on outcomes of other courses, which depend on other courses, etc. Now, each course has a justification for being part of the studies because it is possible to follow the learning path from the outcomes of each course all the way to the professional competence.

We have constructed a prototype of a web application that allows students to explore the curriculum model and build a personal study plan. The software has three main views: *competence profile view*, *course view* and *scheduling view*.

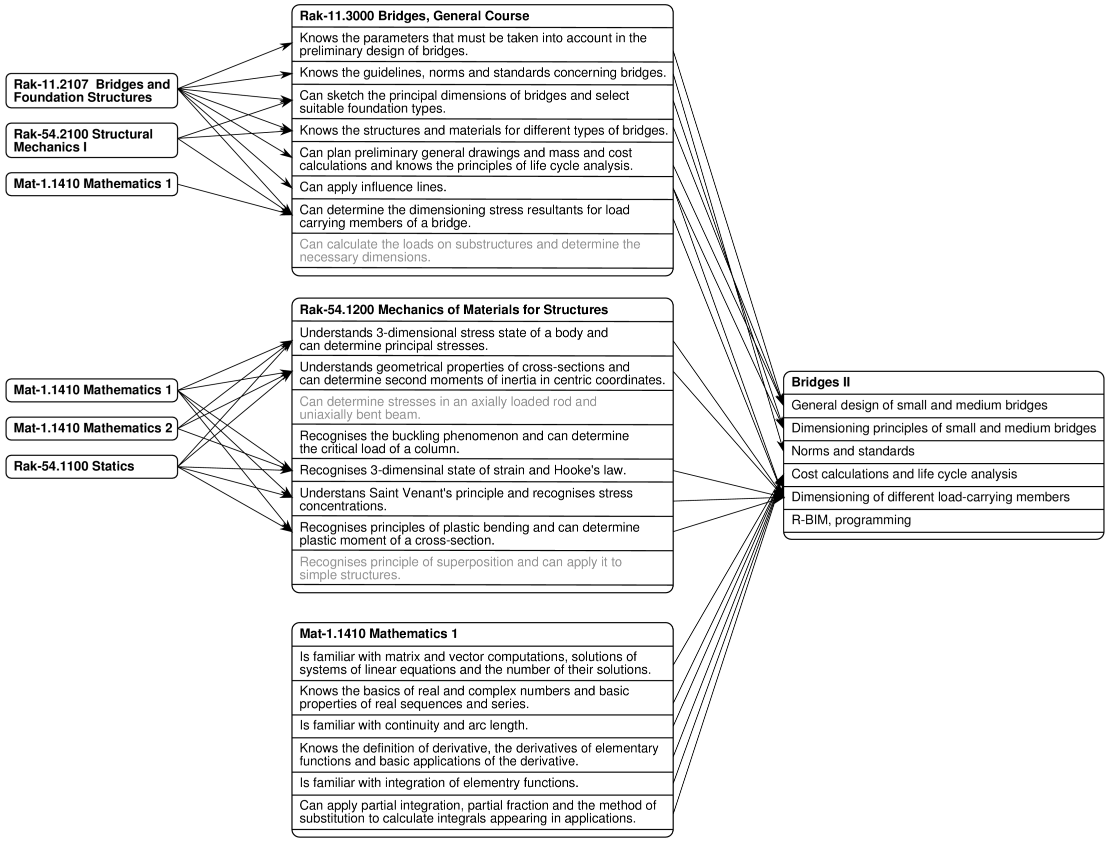

Figure 3: Some prerequisites of the Bridges II competence.

In the *competence profile view*, students are shown the available competences that can be selected as goals. A list of courses and the amount of credit points that are required to build the competence are also shown. Because some courses contribute to multiple competencies, the list of required courses depends on which other competences the student has previously selected. The dynamic course lists also make it easy to see how would changing the goals affect the list of remaining courses and graduation time in case a student wishes to alter the plan during studies.

In the *course view*, students can see how each course contributes to the selected competences. An example study path is shown in Figure 4. The course view also shows the learning outcomes and their connections to immediate prerequisite courses, as well as advanced courses for which each outcome acts as a prerequisite.

In *scheduling view*, students can arrange courses in semesters. An initial schedule is automatically calculated so that prerequisite courses come before advanced courses. When a student selects a course, its prerequisite courses and the courses for which the selected course is a prerequisite, are highlighted. This allows students to see if moving one courses to another semester requires other courses to be moved as well.

The prerequisite graph also makes it possible to automatically construct a personal core curriculum analysis for each student, based on which courses have been selected. For example in Figure 3, when the student has selected the competence *Bridges II*, the outcome "*Knows the structures for different types of bridges and materials"* of the *Bridges, General course* belongs to core curriculum of the student because there is a path from the outcome to the goal competence. The outcomes shown in gray are not core content for the *Bridges II* competence but can contribute to some other competences.

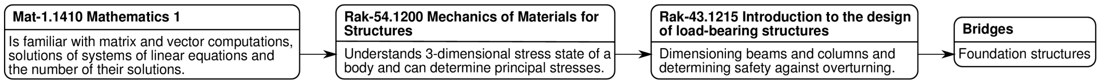

Figure 4: One of many study paths between the Mathematics 1 course and the Bridges competence. Each outcome requires the outcome on its left as a prerequisite.

To create the outcome graph in the first place, an adjacency matrix that defines the prerequisite relations between learning outcomes must be constructed. Each responsible teacher at the Department of Civil and Structural Engineering was first asked to list the learning outcomes of their own courses. A two-dimensional matrix was constructed with all outcomes of all courses on the first row and the first column. Each teacher then went through the outcomes of all other courses and marked the cells that represent a prerequisite to their own course. The task is labourious as every pair of outcomes must potentially be considered. However, the task is divided between multiple teachers, and they have prior knowledge of which courses should contain the relevant prerequisites. The competences have been defined by the professors of the department because at this point, the focus is on describing the current course offering. In the future, it could be fruitful to assemble a panel consisting of representatives from the industry and other interest groups.

4. DISCUSSION AND CONCLUSION

The goal of this work is to motivate students by giving them more freedom, and responsibility, in designing their own study plans. Instead of simply giving students a list of mandatory courses to study, the aim is to let students set personal competence goals and then show which skills must be learned in order to build the competence. A list of courses that produce the necessary skills is generated using the outcome graph. Our hypothesis is that studying motivation is increased when the learning outcomes of each course are linked to higher-level goals.

This approach introduces some challenges in the beginning of studies when students may not yet know where they want to focus. However, the model makes it possible to show to students where can the basic skills be used in in the future and that the basic skills do have practical applications. Constructing a personal core curriculum for each student could also have negative impacts on the learning style some students. Knowing which outcomes are not going to be important in the future, some students might be tempted to optimize the time used for studying by ignoring the less important topics. This could lead to a shallow learning style where they only study the minimum required skills without deepening their knowledge. On the other hand, the motivated students who aim for the best grades would still be required to master all topics on a course. Furthermore, current curricula also give students the freedom to choose whether they aim for the best grades or not.

Constructing the outcome graph can help teaching staff to identify problems in the curriculum. It may, for example, turn out that a competence is built from many separate outcomes of a large number courses, while most of the outcomes on those courses are irrelevant. This could indicate that there is a need for a new course that collects the relevant outcomes into a single course. It may also turn out that some necessary skills of the professional competences cannot be constructed from the available outcomes of existing courses, which indicates a need to add more content to existing courses or to create new courses.

The outcome graph also gives teachers a clear picture of what is taught on other courses, and what students can be expected to know when they enter a course. On one hand, unnecessary overlaps can be removed, and on the other hand, teachers cannot have false assumptions that topics are covered by other courses when they, in fact, are not. Also, when planning changes to courses, teachers can make sure that removing an outcome does not break important study paths. Having to specify the learning outcomes of each course can also help to improve teaching. Ecclestone notes that it gives teachers an opportunity to consider if the current teaching methods actually support the outcomes that the course is supposed to generate [4].

4.1 Future work

The new curriculum model and the software have not been used by students yet. More work is needed to ensure that the model does not produce degrees with too narrow set of competences and that students have a diverse enough basic knowledge of their field. In the future, we are going to let some students test the prototype to evaluate its usefulness and collect feedback. When the curriculum model and the application are mature enough, we plan to let a pilot group use the application in real life to create personal study plans, and evaluate whether it has an effect on their study motivation.

Some ideas from intelligent tutoring systems could also be integrated into the curriculum model. Nkambou et al. [12] maintain in their intelligent tutoring system a student model that represents the current knowledge of each student so that learning materials can be adapted accordingly. In the same way, if the study records would contain information of how well each student has learned each learning outcome, students could automatically be offered extra learning materials on later courses for filling missing prerequisites.

We are currently constructing a model that describes the existing curriculum in order to give students more tools for planning studies. However, the model could also be used for redesigning the whole course offering. After identifying the learning outcomes of courses and competences, the outcomes could be algorithmically arranged into a new set of courses in an optimum way.

5. ACKNOWLEDGEMENTS

The author is preparing his PhD thesis on the software implementation of the STOPS curriculum model developed by Juha Paavola and Juha Hartikainen. The author acknowledges the Department of Civil and Structural Engineering of Aalto University for providing the curriculum data presented in this paper.

REFERENCES

[1] Bordogna, J., Fromm, E. and Ernst, E.W. (1993), Engineering education: Innovation through integration, *Journal of Engineering Education*, **82:1**, 3-8.

[2] Carlson, B., Schoch, P., Kalsher, M., Racicot, B. (1997), A motivational first-year electronics lab course, *Journal of Engineering Education*, **86:4**, 357-362.

[3] Davis, M.H. (2003), Outcome-based education, *Journal of Veterinary Medical Education*, **30:3**, 227-232.

[4] Ecclestone, K. (1999), Empowering or Ensnaring?: The Implications of Outcome-based Assessment in Higher Education, *Higher Education Quarterly*, **53:1**, 29-48.

[5] Gestwicki, P. (2008), Work in progress - Curriculum visualization, *Proceedings of the 38th Annual Frontiers in Education Conference*, Saratoga Springs, NY, 2008.

[6] Paavola, J. and Hartikainen, J. (2011), STOPS curriculum model. Retrieved October 15, 2011, from http://buildtech.tkk.fi/en/studies/stops/presentation.pdf

[7] Hwang, G.J. (2003), A conceptual map model for developing intelligent tutoring systems, *Computers & Education*, **40:3**, 217-235.

[8] Kabicher, S. and Motschnig-Pitrik, R. (2009) Coordinating Curriculum Implementation Using Wiki-supported Graph Visualization, Proceedings of the 2009 Ninth IEEE International

Conference on Advanced Learning Technologies, Riga, Latvia, 742-743.

[9] Lindblom-Ylanne, S. and Hamalainen, K. (2004), The Bologna Declaration as a tool to enhance learning and instruction at the University of Helsinki, *The International Journal for Academic Development*, **9:2**, 153-165.

[10] Murray, T. (1999), Authoring intelligent tutoring systems: An analysis of the state of the art, *International journal of artificial intelligence in education*, **10:1**, 98-129.

[11] Nkambou, R., Gauthier, G. and Frasson, C. (1996), CREAM-Tools: An authoring environment for curriculum and course building in an intelligent tutoring system, *Proceedings of the Third International Conference on Computer Aided Learning and Instruction in Science and Engineering,* Montreal, 1996, 420-429.

[12] Nkambou, R., Lefebvre, B. and Gauthier, G. (1996), A curriculum-based student model for intelligent tutoring systems, *Proceedings of the Fifth International Conference on User Modelling,* Kailua-Kona, HW, 1996, 91-98.

[13] Sommaruga, L. and Catenazzi, N. (2007) Curriculum visualization in 3D, *Proceedings of the Twelfth international conference on 3D web technology*, Perugia, Italy, 2007, 177-180.

[14] Spady, W.G. and Marshall, K.J. (1991), Beyond Traditional Outcome-Based Education, *Educational Leadership*, **49:2**, 67-72.

[15] Spady, W.G. (1993) Outcome-Based Education, *ACSA report no 5,* Belconnen: pii Australian Curriculum Studies Association.

[16] Zucker, R. (2009) ViCurriAS: a curriculum visualization tool for faculty, advisors, and students, *Journal of Computing Sciences in Colleges*, **25:2**, 138-145.