Towards a Blended Learning Approach to Teach a Theoretical Computer Science Module

Leila Silva[1] and Leonor Barroca[2]

[1]*Departamento de Computação, Universidade Federal de Sergipe, Av. Marechal Rondon, s/n, São Cristóvão-SE, Brasil*

[2]*Faculty of Mathematics, Computing and Technology, The Open University, Walton Hall, Milton Keynes, UK*

leila@ufs.br, leonor.barroca@open.ac.uk

Keywords: Blended learning, Flipped classroom, Theoretical computer science.

Abstract: Theoretical computer science is a difficult subject in the computer science curriculum. Innovations in teaching and new pedagogic practices have been developing in the last decade but are still far from being widely applied to computer science We propose that the teaching of more challenging areas of computer science can benefit from opportunities created by a blended approach of face-to-face with online teaching and individual and group activities. We present the design of a Design and Analysis of Algorithms including innovations in pedagogy, as flipped classroom, problem-based lectures and social learning.

1 INTRODUCTION

In Computer Science teaching, the more theoretical modules have been known to be hard for students. In 1988, Robins (1988) reported his experience of teaching theoretical modules in the University of California: "theoretical computer science has an awful reputation among undergraduates.... I have heard many resentful undergraduates describe this course using terms such as *dry*, *boring*, *unmotivated*, *contrived*, *impractical*, and *too abstract*. Interestingly, those very few students (usually those who excel in the material) describe it as *elegant*, *challenging*, *practical*, and *stimulating*.". He also reported that in a 50-student undergraduate class there were 2 or 3 individuals that achieved near-perfect scores. More than a decade later, Hamäiläinen (2004) reported that at most a third of the students that registered for a theoretical module on computability would pass it. More recently, Enström (2014) also mentions the challenge of teaching theoretical computer science and reports some experience of introducing more interactive activities to improve the students' understanding.

From the experience of the first author, in Brazil, student grades in these modules are lower than in other modules, and the dropout and failure rates are usually the highest. In particular, at the Federal University of Sergipe (UFS), in average, 50% of the students dropout or fail the theoretical modules. We do not have statistical analysis to justify this scenario, but many lecturers who teach these modules attribute the cause for bad performance to the poor mathematical background of students.

Despite the importance of theoretical computer science in the curriculum, there are only a few studies that apply innovations in pedagogy to this area, in order to improve motivation, engagement and performance of students. Recent studies include Hamäiläinen (2004) using a problem-based approach to teach computability, Enström and Khan (2010) introducing lab exercices to teach NP-completeness proofs and Chakraborty et al (2011) reviewing the main initiatives in the use of simulators to teach automata theory. These works, however, do not address the entire design of a theoretical module.

The popularization of the use of Information and Communication Technologies (ICTs) in the last decades allows for the exploitation of new pedagogic practices, such as social learning, in which students change their role away from information consumers to start engaging in active cooperation to produce knowledge (Brown and Adler, 2008; Sharples et al., 2013).

The face-to-face (f2f) model of education has incorporated some tools commonly used in distance education, and *blended learning* has emerged as a new trend in education. According to Garrison and

Vaughan (2008) blended learning is defined as, “the thoughtful fusion of f2f and online learning experiences. [..] f2f oral communication and online written communication are optimally integrated such that the strengths of each are blended into a unique learning experience congruent with the context and intended educational purpose.” A module designed to be blended may combine f2f classes, small and large groups, self-directed learning, communication between lecturer and students and between students (Bath and Bourke, 2010). In general, it is possible to blend time, place, resources and activities. For example, a module may include f2f and video lectures, online forums, small group tutorials on-campus, online and f2f quizzes. Several works have reported the positive effect of the use of blended learning and ICT resources (e.g Dziuban et al., 2004; Rovai and Jordan, 2004; Wang et al., 2008). Teaching of computer science and, in particular, of more theoretical areas could benefit from the opportunities created by the use of blended learning.

We present the design of a theoretical computer science module using blended learning. We detail the approach which includes flipped classroom, social learning and problem-based learning, and show some illustrative artefacts. We intend to evaluate our design to answer questions such as:

- How can blended learning help with motivation and learning of theoretical computer science subjects?
- Which methods, practices, tools and resources can be used in a blended module in theoretical computer science?
- Which is the role of social learning in the learning of theoretical computer science?
- Which kinds of open educational resources (OERs) can be used to improve motivation and understanding of hard topics?
- What is the impact of using flipped classroom vs. strictly f2f classes?

The paper is organised as follows: section 2 introduces the design approach used; section 3 introduces the proposed module in the context of the computing curriculum at the UFS; section 4 reviews related work, and section 5 presents the discussion and directions for further research.

2 METHODS

We combine the approaches of Bath and Bourke (2010) and Conole (2010). The former gives us the process, within which the conceptual views proposed by the later are elaborated in an interactive and incremental way. The process comprises five phases: planning, design, development, implementation, review and improvement.

In the planning phase we define: aims and learning objectives of the module, structure and timetable, materials and resources, teaching and learning activities (including communication and collaboration between students and lecturer and between students), teaching strategies (for example, which part of the course should be online or f2f), assessment, and student feedback.

In the design and development phases we detail how learning objectives, teaching and learning activities, and assessment are integrated to enable the lecturer to judge constructive alignment (Biggs and Tang, 2011). This means that we evaluate whether resources and learning and teaching activities support students achievement of the stated learning objectives. We also judge if assessment is consistent with the activities and objectives. We decide on the resources to be used and on the workload. We detail, for each learning objective, which activities take place f2f and online and decide on the balance of the types of activities undertaken by students. We use Conole’s (2010) pedagogy profile that defines the following types: assimilative (attending and understanding contents), information handling (gathering and classifying resources or manipulating data), adaptive (using of modelling or simulation software), communicative (dialogic activities, such as group based discussions), productive (construct an artefact) and experiential (practising skills in a particular context or undertaking an investigation).

For the implementation phase, when the module is taught, we prepare a welcome orientation for students explaining the blended approach used.

In the review and improvement phase we collect and analyse feedback on different aspects of the module (e.g. content, activities, assessments, etc). This feedback provides an opportunity to review different aspects of the module and to reflect on improvements for future instantiations.

3 PROPOSED MODULE

3.1 Background

Design and Analysis of Algorithm is a second year module of the BSc in Computer Science at UFS. Until now the module has been offered strictly f2f, with 50 hours of lectures and 10 hours of assessment. It is expected that students engage in at least six hours of extra-class reading and exercises per week. Assessment comprises four tests and two exams. The lecturer is assisted by at most two more senior students, here called assistants, who have passed this module with good grades. The size of the classes ranges from 30 to 55 students each semester. The success rate is around 50% or less. In general, in a 0 to 10 scale, only up to 10% of the students who pass the module have grades above 8; the majority of grades range between 5.5 and 6.5. This scenario is not common in other advanced modules, and even in comparison with first years modules the success rate is low.

Feedback questionnaires applied in some classes reveal that most student complains are about the difficulty of the subject, the need for more problem solving classes, and for better integration between the theoretical content and real world problems. Students recognise that this module, unlike others, requires continuous dedication to study and that it is not possible to pass a theoretical module only studying for the tests.

3.2 Module Design

In what follows we give an overview of the module design, according to the approach in section 2.

3.2.1 The Planning Phase

An overview of the module is expressed by the Module Map View (Conole 2010). This artefact enables lecturers to think about, and share, the design of the module considering the following meta aspects: Guidance & Support, Contents & Activities, Reflexion & Demonstration and Communication & Collaboration. Guidance & Support include some elements such as module structure, timetable and human resources. Contents & Activities include the topics and activities of the module and the materials used. Reflexion & Demonstration define how internalization and reflection is carried out. Communication & Collaboration list the techniques and resources that support the interaction between students and lecturer or between students. In addition, a module summary and key words indicating the pedagogical approach are provided at the beginning of artefact. The Module Map View has a worksheet format, but due to space restrictions, here we present its content in a textual form.

Module Summary: 2nd year course; 4 credits over 15 weeks; 5 blocks covered by 12 theoretical f2f lectures, 11 theoretical on-line lectures, 10 problem solving classes, 1 project guideline class, 4 assessment classes, 2 project presentation classes, 1 consolidation and feedback class.

Key Words: Blended Learning; Flipped Classroom.

Guidance & Support: module guide; study calendar; study planner; VLE, Google, Google docs, social learning tools, programming environments and tools; one lecturer and two assistants.

Contents & Activities: Blocks - (1) Introduction to Complexity Theory, (2) Searching and Sorting, (3) String Processing, (4) Geometric Algorithms, (5) Problems solved by Dynamic Programming and Backtracking techniques; Activities - assimilative (f2f and online lectures); communicative (problems to discuss and solve in pairs or in group); productive (problems to implement in group, presentation of solutions); experiential (analyse and solve a real problem applying studied algorithms); Resources - books; lecture notes; lecture videos available in VEDUCA (www.veduca.com.br); educational resources from the Web; problems in programming environments; learner-generated presentations.

Reflexion & Demonstration: Assessment - assimilative (quizzes and tests), communicative and productive (problems and project), experiential (project); self-evaluation questions; feedback questionnaire.

Communication & Collaboration: problem solving in pairs; group discussion and implementation of problems; group project; individual or tutor groups led by lecturer and assistants; email and news trough the VLE; chats between students and assistants through social learning tools; online materials and notes; f2f classes.

3.2.2 The Design and Development Phases

In this phase we use artefacts that relate the learning objectives to the activities in the module. Here we show some of these artefacts: the module flow gives an overview of the sequence of the activities within the module; the blended learning design worksheet, adapted from (Bath and Bourke, 2010), shows the learning objectives, how they are assessed, and the teaching and learning activities and resources to achieve those objectives; and the pedagogy profile (Conole, 2010) shows the distribution and weight of the different types of activities.

Module flow

The module includes f2f and online lectures. F2f lectures have two hours of duration and can either consist of the explanation of a topic or be a problem solving class. Online lectures explain a topic through slides, commented with audio recordings. Each problem solving class relies on the contents of online lectures, implementing the flipped classroom model, in such a way that the theory learned extra-class is applied and discussed in f2f-classes.

The slides for each lecture (f2f and online) are delivered to students in the VLE. These slides are elaborated by the lecturer and may integrate OERs. They may include also a list of recommended exercises, chapter readings and selected articles. Some videos available in VEDUCA may be recommended. VEDUCA is a repository of video lectures from well recognised universities translated to Portuguese.

A variety of assessments is considered. To stimulate continuous learning, quizzes are applied. A quiz consists of short questions to test understanding of the main concepts taught in the previous lecture. To improve the link between theory and practice, a supervised group project has been introduced. It consists of finding and solving a real problem using some of the algorithms learned in the module. Four tests are applied; they test problem solving skills using the techniques learned in the module.

The module is divided in five blocks. The first provides the basis for the other blocks and requires more mathematical background. Experience shows that students need more interaction and support to understand the topics of this block and to be motivated for engaging in the module's activities. Therefore, the block is taught in f2f classes. Some classes include quizzes, and a test is applied at the end of the block. Guidelines about the project are given to students at the end of the first block.

Blocks 2 to 5 are taught with lectures and problem solving classes. As problem solving classes require the knowledge of online lectures, students are required to study the topics covered in online lectures before attending problem solving classes. These are designed in pairs, in such a way that the solutions achieved in one class of a pair will be discussed in the next class. In a problem solving class the lecturer starts by clarifying any doubts from the corresponding online lecture. After that, the students answer a quiz, individually. Then, the students are divided in groups of 4 and a different problem is assigned to each group. Each group is subdivided in pairs to work on a solution. Finally, the whole group discusses the best solution in the group. In this process, students are assisted by the lecturer and assistants. The group is asked to implement the solution outside the class and to prepare a short presentation, using slides, to discuss it in the next class, with all students and lecturer. Students are also advised to solve the other groups' problems before the next class, to take an active part in the discussion of the different solutions. All solutions will be available through the VLE after the second class, for further discussion using social learning tools. A test is applied at the end of blocks 2, 3 and 5. Students are required to develop their project before the end of block 5. They have extra-class support given by the lecturer and assistants for the project. The project is presented after block 5.

Blended learning design worksheet

Here we illustrate the content of the worksheet by taking as example block 2, which is taught in a flipped classroom approach and is a good representative of the module design.

Example

Learning objective: understanding and applying the main sorting algorithms.

Ways of assessing the objective: f2f quizzes**,** pair and group exercises in f2f classes and extra-class activities, test. The purpose of the quizzes is to check the understanding of the main concepts of the sorting algorithms, whereas the ability of applying these algorithms in diverse situations is verified in exercises and block test.

Teaching activities: a f2f class about a robust sorting algorithm; online lectures about other sorting algorithms; opportunities to discuss the main difficulties from the online lectures in f2f classes; problem solving f2f class about sorting algorithms; recommended readings; a list of exercises for home work; feedback on assessment and students' work; suggestions for online lectures on the subject from platforms such as VEDUCA; preparation of the assistants for the evaluation of students' work. Assistants play also a teaching role helping with: tutoring students in chat rooms and f2f; preparing a practical f2f class on the use of programming environments; selecting problems to explore in problem solving classes; helping the lecturer in problem solving classes, and with marking.

Learning activities: study the main sorting algorithms, propose solutions for recommended exercises, complete assessments, work in pairs and in a group of 4 to solve and implement problems, prepare a short presentation of solutions achieved, discuss solutions of problems, watch videos about the subject from other universities, access assistants, lecturer and other students to clarify doubts.

Helpful resources: books and OERs, programming environments, VLE and social learning tools.

The Pedagogy Profile

The pedagogy profile (Figure 1) gives an overview of the distribution of learning activities: assimilative, information handling, adaptive, communicative, productive and experiential. The balance between productive and assimilative activities, shows the integration between practical and theoretical activities, one goal of the module conception.

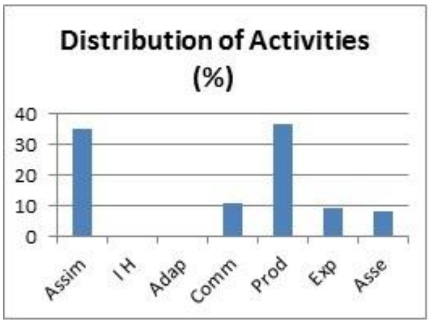

Figure 1. The Pedagogy Profile.

3.2.3 The Implementation and Review and Improvement Phases

The blended module proposed is under development and will be implemented in the first semester of 2016. The authors are currently preparing the module materials.

An experiment will be conducted to validate the approach. The same content and similar assessment will be applied in two classes, one following a blended approach and the other a traditional approach. We want to identify whether students benefit from the innovations introduced. The dropout rate and students' grades, in each block and in the whole module, will be compared.

A feedback questionnaire will be applied at the end of each block and of the module, to measure students' satisfaction concerning various aspects of the module. For example, we intend to analyse the students perception of workload for each activity in the pedagogy profile and the consistency between assessment and learning objectives. We intend to use feedback from the questionnaires to support future iterations of the module.

4 RELATED WORK

Blended learning needs to be strategically planned to be adopted in the whole curriculum or in a specific module (Mortera-Gutiérrez, 2006; Oblinger, 2006). However, there is no well-established procedure to design a blended learning program (Oliver and Trigwell, 2005) and different researchers have suggested different approaches based on five blending dimensions: online and offline learning, self-paced and collaborative learning, structured and unstructured learning, custom and off-the-shelf content and prior and on demand support (Singh, 2003; Garisson and Vaughan, 2008; Dziuban et al., 2004; Larson and Murray, 2008).

Blended learning is being used successfully in several areas of higher education (see for example, Dziuban et al., 2004; Rovai and Jordan, 2004;, Holley and Dobson, 2008), as well as in training programs (Moe and Rye, 2011). In Computer Science, blended modules have been recently experimented with. For example, Alonso et al (2011) report an experience of a blended module of Program Development Models, in which the approval rate is significantly higher than in the strictly f2f module. Similar results have been achieved by Deperlioglu and Kose (2013), in the context of Data Structures. Marin and Pascual Nieto (2012) introduced a free-text score system to encourage students to study core concepts of an Operating System module after classes. Gannod et al. (2008) adopted the flipped classroom approach in the design of a Software Engineering module. Day and Foley (2006) combined successfully video lectures with f2f exercise classes in a module of Human Computer Interaction. Nevertheless, we are not aware of any blended learning approach to teach Design and Analysis of Algorithms as proposed here.

5 CONCLUSIONS

In this paper we propose the introduction of a blended approach to the teaching of the module Design and Analysis of Algorithms. In the design of the module, we introduce some innovations in pedagogy such as flipped classroom, social learning and problem-based learning. OERs, such as video lectures, are used to widen the learning opportunities.

New materials are being identified/adapted/ prepared to help with more difficult topics. The engagement of students in activities is stimulated by the introduction of quizzes and problem solving classes. The introduction of a supervised project is intended to diminish the gap between theory and practice. The problem solving classes and the supervised project are intended to improve students' collaboration and communication skills.

The proposed module is still under construction and will be implemented in the first semester of 2016 at UFS. We expect that the change in the module's dynamics, mixing online and f2f lectures, introducing new forms of assessment and opportunities for social learning will improve the dropout rate and grades.

The immediate future work is to conduct an experiment in order to compare the traditional and the blended approaches in the learning of the same module. As theoretical computer science is a challenging subject, this case study should be regarded as a proof of concept for the applicability of blended learning. We also hope to contribute to the introduction of pedagogic innovations in the Computer Science curriculum.

ACKNOWLEDGEMENTS

This work was partially supported by the National Institute of Science and Technology for Software Engineering (INES), funded by CNPq, grant 573964/2008-4.

REFERENCES

Alonso, F., Manrique, D., Martinez, L., Vines, J M., 2011. How Blended Learning Reduces Underachievement in Higher Education: An Experience in Teaching Computer Science. In *IEEE Transactions on Education,* 54(3): 471-478.

Bath,D., Bourke, J., 2010. *Getting Started with Blended Learning.* Griffith Institute for Higher Education. Technical Report.

Biggs, J., Tang, C., 2011. *Teaching for quality learning at university (Society for Research in Higher Education)*, The Open University Press, Milton Keynes, 4th Edition.

Brown, J. S., Adler, R. P., 2008. Minds on Fire: Open Education, the Long Tail, and Learning 2.0. In *Educause Review*, 43(1):16-32. Educause.

Chakraborty, P., Saxena, P.C., Katti, C.P., 2011, Fifty Years of Automata Simulation. In *ACM Inroads,* 2(4):59-70.

Conole, G., 2010. Learning design- making practice explicit. In *ConnectEd Design Conference,* 1-31.

Day, J. A., Foley, J. D., 2006. Evaluating a Web Lecture Intervention in a Human Computer Interaction Course. In *IEEE Transactions on Education,* 49(4): 420-431.

Deperliogli, O., Kose, U., 2013. The Efectiveness and Experiences of Blended Learning Approaches to Computer Programming Education, In *Computer Applications in Eng. Education*, 21(2): 328-342.

Dziuban, C. D., Hartman, J. L., Moskal, P. D., 2004. *Blended Learning*. EDUCAUSE Center for Applied Research Bulletin.

Enström, E., Kann, V., 2010. Computer Lab Work on Theory. In *Proc. ITiCSE*, 26-30.

Enström, E., 2014. *On Difficult Topics in Theoretical Computer Science Education*, Doctoral Thesis, KTH School of Computer Science and Communication, Stockholm.

Gannod, G. C., Burge J. E., Helmick, M. T., 2008. Using Inverted Classroom to Teach Software Engineering. In *Proc. of ICSE Conf.*, 777-786.

Garrison, D. R., Vaughan, N. D., 2008. *Blended Learning in Higher Education: Framework, Principles, and Guidelines.* John Wiley & Sons, Hoboken, 1st edition.

Hämäläinen, W., 2004. Problem-based Learning of Theoretical Computer Science. In *34th ASEE/IEEE Frontiers in Education Conference,* 20-23.

Holley, D., Dobson, C., 2008. Encouraging students engagement in a blended learning environment: The use of contemporary learning spaces. In *Learning Media and Technology*, 33(2): 139-150.

Larson, R., Murray, M., 2008. Open educational resources for blended learning in high schools: overcoming impediments in developing countries, *J Asynchronous Learn Netw*, 12(1): 1-19.

Marin, D. P., Pascual Nieto, I., 2012. A Case Study on the Use of Blended Learning to Encourage Computer Science Students to Study. In Journal Sci. Educ. Technol., 21: 74-82.

Moe, C. E., Rye, S. A., 2011. Blended Learning: communications, location and work-life practices. *Educational Media International*, 48(3):165-178.

Mortera-Gutiérrez, F., 2006. Faculty best practices using blended learning in e-learning and face-to-face instruction. *International Journal on Elearning, ProQuest Education Journals*, 5(3): 313-337.

Oblinger, D. G., 2006. Space as a change agent. In D. G. Oblinger (Ed.), *Learning spaces* (pp. 1.3). Educause e-book.

Oliver, M., Trigwell, K., 2005. Can 'blended learning' be redeemed? *E-learning*, *2* (1): 17-26.

Robins, G., 1988. *Teaching Theoretical Computer Science at the Undergraduate Level: Experiences, Observations and Proposals to Improve the Status Quo*, Technical Report, UCLA CSD, vol. 880063.

Rovai, A. P., Jordan, H. M., 2004. Blended Learning and Sense of Community: a Comparative Analysis with Traditional and Fully Online Graduate Courses. In *Int Rev of Research in Open and Distance Learning*, 5(2).

Sharples, M., McAndrew, P., Weller, M., Ferguson, R., Fitzgerald, E., Hirst, T., Gaved, M., Smith, J., 2013. In *Innovating Pedagogy*. The Open University Innovation Report 2.

Singh, H., 2003. Building effective blended learning programs. *Educ Technol Mag,* 43(6):51-54.

Wang, F. L., Fong, J., Choy, M., Wong, T., 2008. Blended Teaching and Learning of Computer Programming. In *Int. Conf. on Web-based Learning,* LNCS 4823, 606-617, Springer.