Integrating Computational Modelling in Science, Technology, Engineering and Mathematics Education

Rui Gomes Neves, Jorge Carvalho Silva and Vítor Duarte Teodoro

1 Introduction

Science, technology, engineering and mathematics (STEM) are evolving structures of knowledge which are symbiotically interconnected. On one hand, science is based on hypotheses and models, leading to theories, which have a strong mathematical character as scientific reasoning, concepts, and laws are represented by mathematical reasoning, entities, and relations. On the other hand, scientific explanations and predictions must be consistent with the results of systematic and reliable experiments, which depend on technological developments as much as these depend on the progress of science and mathematics (see, e.g., Chalmers 1999; Crump 2002; Feynman 1967). The creation of STEM knowledge is a dynamical cognition process, which involves a blend of individual and collective reflexions where modelling occurs with a balance between theoretical, experimental and computational elements (Blum et al. 2007; Neunzert and Siddiqi 2000; Schwartz 2007; Slooten et al. 2006). In this research paradigm, computational modelling plays a key role in the expansion of the STEM cognitive horizon through enhanced calculation, exploration and visualization capabilities.

Although clearly related to real world phenomena, STEM knowledge is built upon abstract and subtle conceptual and methodological frameworks, which have a complex historical evolution. These epistemological and cognitive features make

---

R. G. Neves (✉) · V. D. Teodoro
Unidade de Investigação Educação e Desenvolvimento (UIED) e, Departamento de Ciências Sociais Aplicadas (DCSA), Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa, Lisbon, Portugal
e-mail: rgn@fct.unl.pt

V. D. Teodoro
e-mail: vdt@fct.unl.pt

J. C. Silva
Centro de Física e Investigação Tecnológica (CEFITEC) e, Departamento de Física, Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa, Lisbon, Portugal
e-mail: jcs@fct.unl.pt

A. Damlamian et al. (eds.), *Educational Interfaces between Mathematics and Industry*, New ICMI Study Series 16, DOI: 10.1007/978-3-319-02270-3_38,

STEM difficult fields to learn and to teach. To develop an approach to STEM education which aims to be effective and in phase with the rapid scientific and technological development, it is of crucial importance to promote an early integration of computational modelling in learning environments which reflect the exploratory and interactive nature of modern research (Ogborn 1994). However, even in technologically advanced countries, computers, computational methods and software, as well as exploratory and interactive learning environments, are still not appropriately integrated in most STEM education curricula for the high school and undergraduate university levels. As a consequence, these curricula are generally outdated and most tend to transmit to students a sense of detachment from how science is currently made. These are contributing factors to the development of negative views about the education process and to an increase in student failure.

Physics education is a good example to illustrate this problem. Consider the general physics courses taken by first year university students. These are courses which usually follow a traditional lecture plus laboratory instruction approach and cover a large number of physics topics which students find particularly difficult. Due to a lack of understanding of the necessary fundamental concepts in physics and mathematics, the number of students that fail on the course examinations is usually very high. What is worse is that many of those students that do actually succeed also reveal several weaknesses in their understanding of elementary physics (Halloun and Hestenes 1985; Hestenes 1987; McDermott 1991).

Research in physics education has shown that this situation can be improved when students are involved in the learning activities as scientists are involved in research (Beichner et al. 1999; Mazur 1997; McDermott 1997). This is not a surprising result. Scientific research in physics is an interactive and exploratory process of creation, testing and improvement of mathematical models that describe observable physical phenomena. It is this cognitive process that leads to an inspiring understanding of the rules of the physical universe. As a consequence, physics should be expected to be more successfully taught in interactive and exploratory environments where students are helped by teachers to work as scientists do. In this kind of class environment knowledge performance is better promoted and common sense beliefs as well as incorrect scientific ideas can be more effectively fought.

The scientific research process is supported by a continuously evolving set of analytical, computational and experimental techniques. The same should be true for research inspired learning environments. Consequently, another important aspect of these learning environments is the possibility to balance the role of computational modelling methods and tools. This would set the learning process in phase not only with modern scientific research where computation is as important as theory and experiment, but also with the rapid parallel development of technology.

Several attempts have already been made to introduce computational modelling in research inspired learning environments. The starting emphasis was on professional programming languages such as Fortran (Bork 1967) and Pascal (Redish and Wilson 1993). Although more recently this approach has evolved to Python

(Chabay and Sherwood 2008), it continues to require that students develop a working knowledge of programming, a time-consuming task which hinders the process of learning physics. The same happens with scientific computation software such as Mathematica and Matlab. To avoid overloading students with such programming notions or syntax, and focus the learning process on the relevant physics and mathematics, several computer modelling systems were created, for example, Dynamical Modelling System (Ogborn 1985), Stella (High Performance Systems 1997), Easy Java Simulations (Christian and Esquembre 2007) and Modellus (Neves et al. 2013; Teodoro and Neves 2011).

A proper and balanced integration of computational modelling methods and tools in STEM learning environments is, thus, both a curricular and a technological development problem. In this work, we discuss how Modellus (a freely available software tool created in Java which is able to run in all operating systems, see the software webpage a http://modellus.fct.unl.pt) can be used as a central element of an approach to develop exploratory and interactive computational modelling learning activities relevant for STEM education. These activities can be adopted by high school and university curricula, and used as a valuable instrument for the professional development of teachers. To illustrate, we consider computational modelling with Modellus to teach physics, namely introductory mechanics, and discuss its impact on the student learning process.

2 Modellus: Interactive and Exploratory Computational Modelling for STEM Education

The construction of STEM knowledge requires unambiguously clear declarative, operational and conditional specifications of abstract concepts and of the relations among such concepts. Of crucial importance in the understanding of the resulting models or theories is the interpretation process which involves operational familiarization and connection with the relevant referents in the observable universe (Reif 2008). In education, as in research, computers, computational methods, and software are cognitive artifacts (Teodoro 2005), which may amplify the learning cognition horizon due to more powerful calculation, exploration and visualization capabilities. As a consequence, these artifacts may play a key role in enhancing the operational familiarization and the connection with the real world referents, which must necessarily be involved in the STEM learning processes. To be able to fulfill such a potential key role, computational methods and tools should be used not only to display text, images or simulations but as mathematical modelling tools integrated in learning environments which reflect the exploratory and interactive nature of modern research. In addition, the modelling process should be focused on the meaning of models and avoid learning opacity factors such as too much programming and specific software knowledge.

To meet this educational challenge, it is not enough to simply choose a subset of programming languages and professional computational software. It is necessary to develop computer software systems with computational modelling functionalities, which contribute to nurture the progressive growth of solid STEM cognitive competencies. Among those systems which have already been created, Modellus stands out as a computational modelling tool for STEM education because of the following main advantages: (1) an easy and intuitive creation of mathematical modells using standard mathematical notation; (2) the possibility to create animations with interactive objects that have mathematical properties expressed in the model; (3) the simultaneous exploration of multiple representations such as images, tables, graphs and animations; and (4) the computation and display of mathematical quantities obtained from the analysis of images and graphs.

These are features that allow a deeper cognitive contact of models with the relevant real world referents and a deeper operational exploration of models as objects, which are simultaneously abstract, in the sense that they represent relations between mathematical entities, and concrete, in the sense that they may be directly manipulated in the computer. In a word, Modellus allows a deeper reification of abstract mathematical objects. Because of these characteristics, computational modelling activities built with Modellus can be readily conceived as exploratory and interactive modelling experiments performed by students in collaborative groups or individually. They can also be designed with an emphasis on cognitive conflicts in the understanding of STEM concepts, on the manipulation of multiple representations of mathematical models and on the interplay between the analytical and numerical approaches applied to solve STEM problems.

As a domain general environment for modelling, Modellus can be used to conceive STEM learning activities, which involve the exploration of existing models and the development of new ones (Bliss and Ogborn 1989; Schwartz 2007). As much as possible, such modelling activities should consider realistic problems to maximize the cognitive contact with the real world referents. This is a challenge because more realistic problems are generally associated with more complex analytic solutions, which are beyond the analytic capabilities of high school or first to second year university students. With Modellus and numerical methods, which are conceptually simpler and yet powerful, the interactive exploration of models for more realistic problems can start at an earlier age, allowing students a closer contact with the model referents, an essential cognitive element to appreciate the relevancy and power of models, necessarily a partial idealized representation of their referents.

Clearly, the development of the appropriate computational modelling activities for STEM research inspired learning environments is bound to call for a richer set of modelling functionalities, which are not yet available in Modellus. These events are seeds for technological evolution, which should be accomplished by a Modellus enhancement program. Currently under development and set to appear in forthcoming versions of Modellus is, for example, the following set of new functionalities: spreadsheet, data logging and curve fitting capabilities, advanced animation objects like curves, waves and fields, 3D animations and graphs, creation of a physics engine for motion and collisions, video analysis, and cellular automata models.

The simultaneous development of new functionalities to meet appropriate teaching goals is important because it reduces the learning opacity factor associated to an unnecessary proliferation of tools. However, there is a learning stage where it is advantageous to allow some diversity in the use of computer software tools and complement Modellus with other available tools. Indeed, in a research-based STEM learning environment one of the objectives is to make a progressive introduction to professional STEM computation methods and software. For example, Excel is a general purpose spreadsheet where modelling is focused on the algorithms. In addition, it already allows data analysis from direct data logging. On the other hand, Mathematica and Matlab (or wxMaxima, a similar but freely available tool) have powerful symbolic computation capabilities. Using these different tools to implement the same algorithm is an important step to learn the meaning of the algorithm instead of the syntax of a particular tool. If more realistic simulations are needed, Modellus animations can be complemented, for example, with EJS.

3 Computational Modelling Learning Activities: An Illustrative Example from Rotational Dynamics

Let us consider a computational modelling activity about rigid body rotational dynamics and angular momentum, a topic in general physics which in a course program for first year university students should be introduced after computational modelling activities covering vectors, kinematics and Newton's fundamental laws of motion, including simple numerical and analytical solutions (Neves et al. 2009, 2011). A rigid body is a system of particles whose relative distance does not change with time. When a rigid body rotates around a fixed axis, each one of its particles has a circular motion around the axis, which is characterized by a rotation angle, an angular velocity and an angular acceleration. The kinetic rotation energy is the sum of the kinetic energies of all the particles of the body and is given in terms of the moment of inertia of the body relative to the rotation axis. From a dynamical point of view, the rotational motion of a rigid body around a fixed axis is characterized by two vectors, the angular momentum of the rigid body and the moment of the sum of all the forces acting on the body, the latter also called the net applied torque. Newton's laws of motion imply that the instantaneous rate of change of the angular momentum is equal to the net applied torque.

A real world system, which may be considered as a rotating rigid body is a wind turbine. With Modellus it is possible to model the action of the wind on the rotor blades and analyze their motion using at the same time different representations such as graphs, tables, and object animations. In this model, the fundamental equations of the rotational motion are written in the form of Euler iterations. Students are thus taught to apply this numerical method in a new realistic context, extending the applicability range of knowledge already acquired with a previous analysis of analogous numerical solutions of Newton's equations in translational motion settings (Neves et al. 2009). In this new application, students can determine the angular velocity and the rotation angle knowing the net applied torque, the moment of inertia of the system and the motion initial conditions. The model animation is constructed with three objects: a bar representing the rotor blade, a vector representing the angular momentum and a vector representing the net wind torque. Because the coordinates of the net torque are independent variables and the model is iterative, students can manipulate this vector at will and in real time control the motion of the rotor blade. With this activity students can confirm that the choice of the time step is an important one to obtain a good simulation of the motion and that this is the same as determining a good numerical solution of the equations of motion. While exploring the model, students can determine, for example, what are the values of the angular momentum, the angular velocity and the rotational kinetic energy, 8 s after increasing the net wind torque to 2,000 Nm. The possibility to change the mathematical model and immediately observe this action on the animation, graphs and tables is a powerful cognitive element to enhance the students learning process. Students can also change the model. Introducing a vector to represent the wind force they can explore the effect of the wind direction on the net applied torque and on the motion of the rotor blades.

4 Field Actions, Discussion and Outlook

In this chapter, we have shown how Modellus can be used as a key element of an approach to develop exploratory and interactive computational modelling learning activities for research inspired STEM education. As an example, we have discussed the modelling of the rotational dynamics of a wind turbine. This was one of the activities which was tested on the field when we implemented computational modelling activities in the general physics course offered in 2008 and 2009 to the first year biomedical engineering students at the Faculty of Sciences and Technology of the New Lisbon University (Neves et al. 2008, 2009, 2011). More field tested activities are discussed, e.g., in Neves et al. (2013) and Neves and Teodoro (2012). For other earlier educational applications and evaluation tests of Modellus-based computational modelling activities see, e.g., Araújo et al. (2008), Dorneles et al. (2008) and Teodoro (2002).

The 2009 edition of the general physics course for biomedical engineering involved a total of 115 students. Of these, 59 were taking the course for the first time and only they were enrolled in the computational modelling classes. To build an interactive collaborative learning environment, we organized the students in groups of two or three, one group for each computer in the classroom. During each class, the student teams worked on a computational modelling activity set conceived by us to be an interactive and exploratory learning experience with Modellus, built around a small number of problems in mechanics connected with easily observed real world phenomena. The teams were instructed to analyze and discuss the problems on their own using the physical, mathematical and computational modelling guidelines provided by the activity set documentation, a set of PDF documents with embedded video support. To ensure a good working pace with appropriate conceptual, analytical and computational understanding, the students were continuously monitored and helped during the exploration of the activities. Whenever it was felt necessary, global class discussions were conducted to clarify doubts on concepts, reasoning, or calculations. Online support in class and at home was provided in the context of the Model platform where links to class and homework documentation was provided.

The evaluation procedures associated with the computational modelling activities with Modellus involved both group evaluation and individual evaluation. For each computational modelling class, all student groups had to complete an online test written in the Moodle platform answering the questions of the corresponding activity PDF document. The individual evaluation was based on the student solutions to two sets of homework activities and a final class test with computational modelling problems to be solved with Modellus. All students took a pre-instruction and post-instruction FCI test (Hestenes et al. 1992) which did not count for their final course grade. At the end of the semester, the students answered a questionnaire to evaluate Modellus and the new computational modelling activities of the general physics course. As reflected by the student answers to the 2008 and 2009 questionnaires (Neves et al. 2008, 2011), the activities with Modellus were helpful in the learning process of mathematical models in the context of introductory physics. For students, Modellus was easy to learn and user-friendly. Working in groups of two or three was also acknowledged to be more advantageous than working individually. In addition, the PDF documents with embedded video guidance were considered to be interesting and well designed.

During class and homework communications, it was clear that the computational modelling activities with Modellus were being successful in identifying and resolving several student difficulties in key physical and mathematical concepts of the course. The possibility to explore simultaneously several different representations like graphs, tables and animations was a key success factor, in particular the possibility to have a real time visible correspondence between the animations with interactive objects and the object's mathematical properties defined in the model. During the course, Modellus and simple numerical methods, such as the Euler and Euler-Cromer methods, allowed the introduction and exploration of advanced mathematical concepts such as integration in the context of real world physics problems, prior to the introduction of the corresponding analytic techniques. However, FCI test results for the whole general physics course lead to an average FCI gain of 22 %, a typical traditional instruction performance (Hake 1998). Moreover, students manifested clearly that the new content load was too heavy and that the available time to spend working on the computational modelling activities was insufficient. Improvement actions (Neves et al. 2011) are now being implemented.

Acknowledgments Work supported by Unidade de Investigação Educação e Desenvolvimento (UIED) and Fundação para a Ciência e a Tecnologia (FCT), Programa Compromisso com a Ciência, Ciência 2007.

References

Araújo, I., Veit, E., & Moreira, M. (2008). Physics student's performance using computational modelling activities to improve kinematics graphs interpretation. *Computers & Education, 50*, 1128–1140.

Beichner, R., Bernold, L., Burniston, E., Dail, P., Felder, R., Gastineau, J., et al. (1999). Case study of the physics component of an integrated curriculum. *Physics Education Research, American Journal of Physics Supplement, 67*, 16–24.

Bliss, J., & Ogborn, J. (1989). Tools for exploratory learning. *Journal of Computer Assisted learning, 5*(1), 37–50.

Blum, W., Galbraith, P., Henn, H.-W., & Niss, M. (Eds.). (2007). *Modellng and applications in mathematics education*. New York: Springer.

Bork, A. (1967). *Fortran for physics*. Reading, Massachusetts, USA: Addison-Wesley.

Chabay, R., & Sherwood, B. (2008). Computational physics in the introductory calculus-based course. *American Journal of Physics, 76*, 307–313.

Chalmers, A. (1999). *What is this thing called science?*. London, UK: Open University Press.

Christian, W., & Esquembre, F. (2007). Modelling physics with easy java simulations. *The Physics Teacher, 45*(8), 475–480.

Crump, T. (2002). *A brief history of science, as seen through the development of scientific instruments*. London, UK: Robinson.

Dorneles, F., Araújo, I., & Veit, E. (2008). Computational modelling and simulation activities to help a meaningful learning of electricity basic concepts. Part II: RLC circuits. *Revista Brasileira de Ensino da Física*, 30(3), 3308–1.

Feynman, R. (1967). *The character of physical law*. New York, USA: MIT Press.

Hake, R. (1998). Interactive-engagement versus traditional methods: A six-thousand-student survey of mechanics test data for introductory physics courses. *American Journal of Physics, 66*, 64–74.

Halloun, I., & Hestenes, D. (1985). The initial knowledge state of college physics students. *American Journal of Physics, 53*, 1043–1048.

Hestenes, D. (1987). Toward a modelling theory of physics instruction. *American Journal of Physics, 55*, 440–454.

Hestenes, D., Wells, M., & Swackhamer, G. (1992). Force concept inventory. *The Physics Teacher, 30*, 141–158.

High Performance Systems. (1997). *Stella, version 5*. Hannover, NH: High Performance Systems.

Mazur, E. (1997). *Peer instruction: A user's manual*. New Jersey, USA: Prentice-Hall.

McDermott, L. (1991). Millikan lecture 1990: What we teach and what is learned–closing the gap. *American Journal of Physics, 59*, 301–315.

McDermott, L. (1997). *Physics by inquiry*. New York, USA: Wiley.

Neunzert, H., & Siddiqi, A. (2000). *Topics in industrial mathematics: case studies and related mathematical models*. Dordrecht, Holland: Kluwer Academic Publishers.

Neves, R., Neves, M. C., & Teodoro, V. (2013). Modellus: Interactive computational modelling to improve teaching of physics in the geosciences. *Computers & Geosciences*, *56*, 119–126.

Neves, R., Silva, J., & Teodoro, V. (2008). Improving the general physics university course with computational modelling. Poster presented at the 2008 Gordon Research Conference, Physics Research and Education: Computation and Computer-Based Instruction, Bryan University, USA.

Neves, R., Silva, J., & Teodoro, V. (2009). Computational modelling with Modellus: an enhancement vector for the general university physics course. In A. Bilsel & M. Garip (Eds.), *Frontiers in science education research* (pp. 461–470). Famagusta, Cyprus: Eastern Mediterranean University Press.

Neves, R., Silva, J., & Teodoro, V. (2011). Improving learning in science and mathematics with exploratory and interactive computational modelling. In G. Kaiser, W. Blum, R. Borromeo-Ferri & G. Stillman (Eds.), *International Perspectives on the Teaching and Learning of Mathematical Modelling Vol. 1, ICTMA14: Trends in Teaching and Learning of Mathematical Modelling* (pp. 331–341). Dordrecht: Springer.

Neves, R., & Teodoro, V. (2012). Improving science and mathematics education with computational modelling in interactive engagement environments. In T. Simos, G. Psihoyios, C. Tsitouras and Z. Anastassi (Eds.), *Numerical Analysis and Applied Mathematics 2012, AIP Conference Proceedings* 1479, 1806–1809.

Ogborn, J. (1985). *Dynamic modelling system*. Harlow, Essex, UK: Longman.

Ogborn, J. (1994). Modelling clay for computers. In B. Jennison & J. Ogborn (Eds.), *Wonder and delight, Essays in science education in honour of the life and work of Eric Rogers 1902–1990* (pp. 103–114). Bristol: Institute of Physics Publishing.

Redish, E., & Wilson, J. (1993). Student programming in the introductory physics course: M.U.P.P.E.T. *American Journal of Physics, 61*, 222–232.

Reif, F. (2008). *Applying cognitive science to education: Thinking and learning in scientific and other complex domains*. Cambridge, USA: MIT Press.

Schwartz, J. (2007). Models, simulations, and exploratory environments: A tentative taxonomy. In R. A. Lesh, E. Hamilton, & J. J. Kaput (Eds.), *Foundations for the future in mathematics education* (pp. 161–172). Mahwah, NJ: Lawrence Erlbaum Associates.

Slooten, O., van den Berg, E., & Ellermeijer, T. (Eds.). (2006). *Proceedings of the International Group on Research on Physics Education (GIREP) 2006 conference: modelling in physics and physics education*. Amsterdam, Holland: European Physical Society.

Teodoro, V. (2002). Modellus: learning physics with mathematical modelling. PhD Thesis. Lisboa, Portugal: Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa.

Teodoro, V. (2005). Cognitive artefacts, technology and physics education. *Interactive Educational Multimedia, 11*, 173–189.

Teodoro, V., & Neves, R. (2011). Mathematical modelling in science and mathematics education. *Computer Physics Communications*, *182*, 8–10.