Collaborative learning by modelling: Observations in an online setting

Peter Reimann, Kate Thompson and Miriam Weinel
CoCo Research Centre
The University of Sydney

A custom-designed combination of a chat tool and a wiki tool was used to engage postgraduate education students online in system dynamics modelling tasks. The purpose of the course was to familiarise students with core concepts of the complexity sciences, and to introduce them to modelling complex systems as a means to research processes of learning and organisational change. The rationale for the online course as well as the technology employed is described. Observations from two student teams using the Stella™ modelling software while cooperating in the online learning environment are reported, both with respect to their modelling activities as well as their team coordination behavior. We conclude with an identification of the main advantages of learning about a difficult subject area collaboratively and net-based.

Keywords: OLE, chat, wiki, system dynamics modelling, postgraduate students

Introduction

As Jacobson and Wilensky (2006) observe, although the study of complex systems "…is leading to the articulation of an integrated framework of ideas and methods that is generating excitement among scientists, policymakers, and segments of the public…" (p. 12), complexity sciences are, so far, not coherently integrated into school curricula nor are they systematically taught at the university level, with the exception of some areas of science. This is particularly disconcerting because complexity sciences are increasingly relevant for both societies and individuals who are dealing with an ever more complex socio-economic environment, and because problems that are beyond simple linear causality, such as climate change, have reached a global scale.

In order to get the science of complexity more firmly integrated into the educational systems, in addition to policy decisions, current and future teachers will need to be educated about relevant concepts (such as multi-scale hierarchical organisations, emergence, attractors) and relevant methods to analyse complex phenomena (such as cellular automata, agent-based modelling, system dynamics modelling). As part of a newly created Masters Program "Learning Sciences and Technologies" in the Faculty of Education and Social Work at the University of Sydney, we have developed a course that aims to develop an understanding of systems thinking and complexity theory concepts in educators. The course *Learning and Change: a Systems View* (LCSV) relates relevant concepts and methods as directly as possible to educational issues, such as models of learning and models of organisational processes occurring in educational systems. Our basic pedagogy in this course is one of 'learning by modelling': students analyse a range of educational processes by improving upon existing models of these processes and building new models from scratch.

The main model type used is system dynamics. System dynamics modelling is a powerful method used by both researchers and educators to provide opportunities to simulate and reason about complex systems. Very few studies, however, have examined how students participate in a collaborative modelling task. *LCSV* provided us with the opportunity to compare both the way in which students engaged in learning by modelling and learning with modelling activities collaboratively in an online learning environment, and also the ways in which they utilised the tools given in the Online Learning Environment. After having described this learning environment and the course approach in more detail, we report our first observations of students' modelling activities and interaction behavior, focussing on their synchronous communication.

The online learning environment

The OLE combines synchronous and asynchronous communication components. The main asynchronous collaboration medium used in this course was a wiki engine. A wiki is essentially a shared document (or set of documents), and a wiki engine is a collaborative text authoring tool. Using a wiki (engine) as a collaboration tool is somewhat different from situations where dedicated interaction technologies such as chat, discussion boards, newsgroups or email are used. However, it is quite typical for the communication that takes place between software developers (Ripochet & Sansonnet, 2006) or between authors of jointly written documents (Zacklad, 2006). In such groups, what typically happens is a combination of face-to-face meetings, synchronous remote communication such as phone conversations and an asynchronous textual medium such as a wiki. The artefacts created on wikis and in version-controlled collaborative document repositories can be seen as combining work on the task with interaction and coordination functions, to the extent that such artefacts are used not only to document work, but also to co-ordinate team members' activities and to structure their interactions (at least partly, in addition to what is not said and done in the synchronous communication). Using such document-like artefacts is convenient because they are often part of the groups' work anyway and hence constitute little communication overhead (MacMillan, Entin, & Serfaty, 2004). For instance, software designers often use wikis to document cases and to develop user manuals, and they use versioning systems to not only manage the code but also to distribute tasks amongst the team members (Layman, Williams, Damian, & Bures, 2006).

This convenience factor can easily lead to problems: Because interaction and coordination functions are not systematically separated from production tasks, and given that documents tend to grow quickly in size over a project's life, it can become difficult for team members to keep track of tasks and commitments. One way to address this issue is to separate the coordination aspects from the production aspects but keep them within the same basic medium. This is, for instance, possible in a system such as *Xplanner* [http://www.xplanner.org/] and in *Trac* [http://trac.edgewall.org] the group programming support tool employed in our study. Trac uses a ticket system for task management that is closely integrated, but separate from the wiki engine and the versioning engine.

Groups were required to use Trac, a tool designed for programmers working in teams to build software. It has three, tightly integrated parts:

- A *wiki* for collaborative editing of web pages for general group communication.
- An issue tracking system based on so-called *tickets* (see Figure 1), where one creates a ticket when a task needs to be done and this is allocated to a team member and, when the task has been completed, the ticket is closed. Tickets can be referred to from within wiki pages using the ticket number, and tickets can be grouped around *milestones*.
- A browsing interface to a repository based upon the version control system called *Subversion*, for storing documents like source code, including any versions.

Students in this course used the wiki engine and the ticket system, but not the file versioning module. All pages were accessible for editing by all students.

Weekly chat meetings were conducted including all students and a session moderator (one of the lecturers). The chat environment was developed by ourselves (Ullman, Peters, & Reimann, 2005), is accessible through any web browser, and combines the following elements (see Figure 2):

- A Chat area with a colour-coded chat history, access to audio conferencing (not used in this study), a polling tool and a participation awareness display, the "team radar" (on the left; each dot represents a user (unique colour for each user, same as text color in the Chat history window).
- A Notes area to display and edit text snippets. Used, for instance, to display and edit agenda items. Notes can be created and edited 'on the fly', for instance to capture ideas from a brainstorming activity. Editing of notes is "serial": When a note is edited, it is locked and can only be modified by the user who initiated editing. Changes are only visible to others once the author has pressed the "Send" button, which results in the lock being lifted and the changes becoming visible. Notes entries are versioned, with all users having access to the current and previous versions.
- A shared Whiteboard area. The Whiteboard supports "drawing" of simple graphical elements (arrows, rectangles, text boxes) in a parallel access fashion: all users can interact with the Whiteboard in parallel, and all changes (such as moving an arrow) become immediately visible. The software "remembers" who has drawn what and provides respective information when the mouse comes close to any graphics object. In addition to drawing, pictures can be loaded into the background, thus becoming available for graphical annotating.

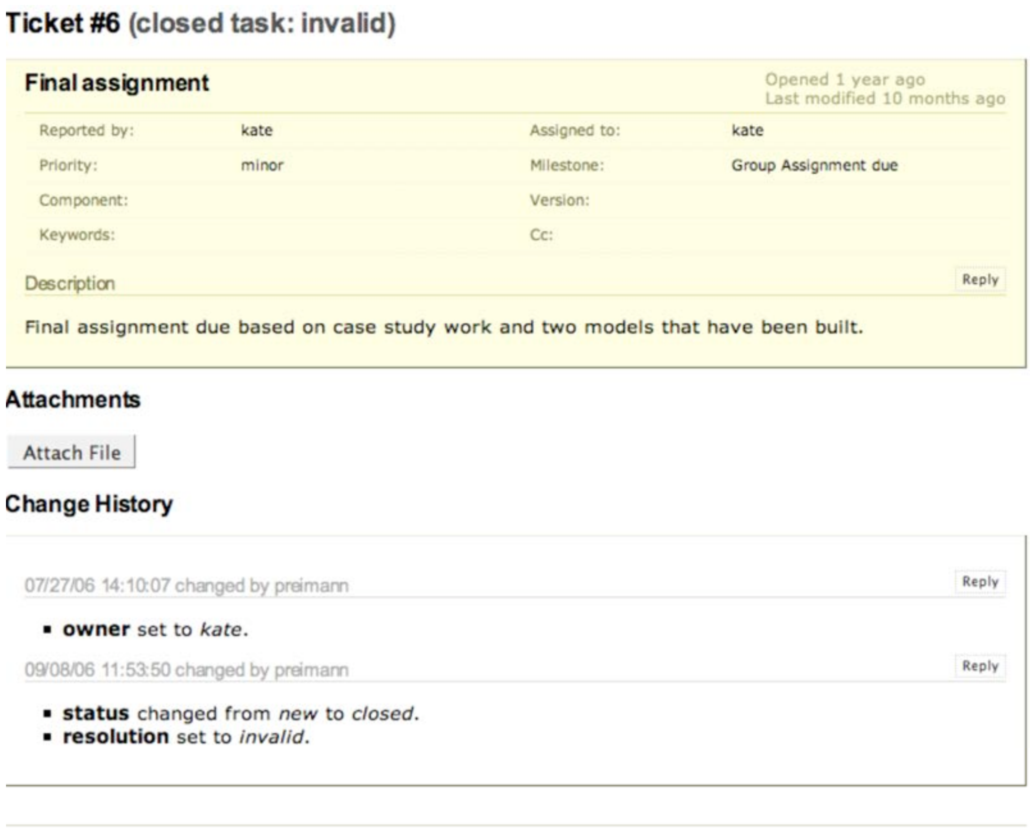
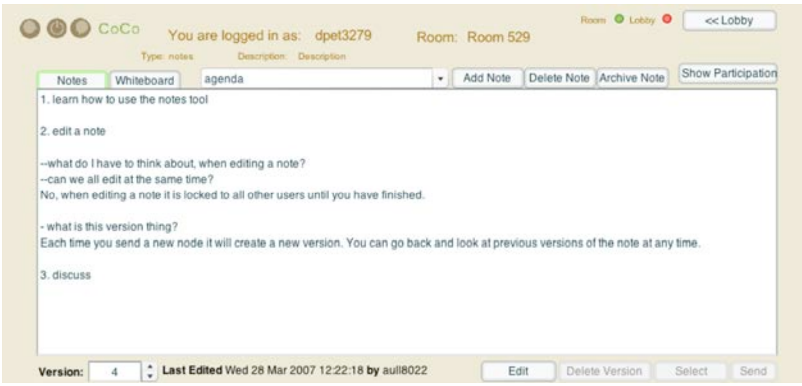
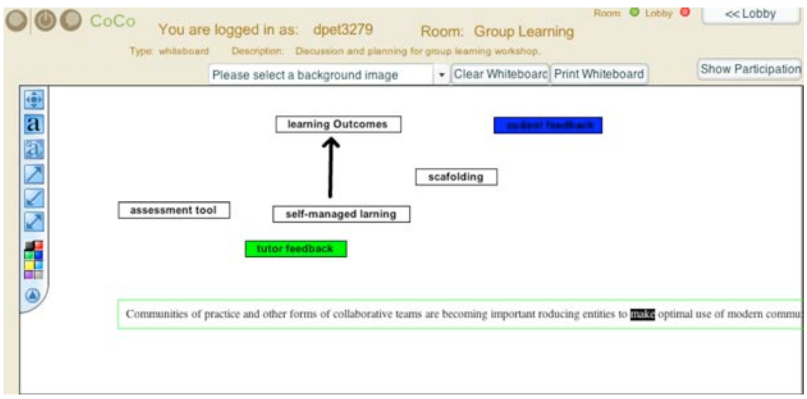


Figure 1: Example of a ticket

Entries on all three areas can be archived by users on demand; by the press of a button, the current entries in the Chat, Whiteboard, and Notes window, respectively, are saved in a file and the file becomes available in the asynchronous team space. For instance, we encouraged students to save their chat sessions and analyse them for relevant information after the chat session had ended.



Notes view



Whiteboard view

Students in the *Learning and Change, A Systems View* class used all three tools in the Chat environment. The use of the Chat environment was quite straightforward, used for communication both within the class as a whole and within the two groups. The Notes area was used to plan the group's chat. Whoever was in charge of leading the group that week would use the Notes tool to plan an agenda for the group and to put any additional information for the group to read, questions etc. in a common space that everyone could see. It was used within the class to put the steps of the modelling process in their training sessions. The shared Whiteboard area was used for the whole class to demonstrate feedback loops, and to aid students in their identification of positive and negative feedback loops. The Whiteboard has a feature of being able to import a background image. So students also used this feature to upload screenshots of their model and have them visible for the group to see and discuss.

System dynamics modelling

"System Dynamics is a methodology for analysing complex systems and problems with the aid of computer simulation software" (Alessi, 2000, p. 1) and includes cause and effect relationships, time delays and feedback loops. Jay Forrester described the philosophy and method of the approach of system dynamics in 1961 with the publication of *Industrial Dynamics.* Systems can be represented by *causal loop diagrams* and by *stock and flow diagrams*. Causal loop diagrams are useful for demonstrating feedback (Sterman, 2000). Feedback is a defining element of a complex system. Forrester identifies feedback as the most important element in defining a system. "The feedback loop is the closed path that connects an action to its effect on the surrounding conditions, and these resulting conditions in turn come back as "information" to influence further action" (Forrester, 1971, p. 17). A positive loop occurs when the anticipation of a future reaction changes current behaviour (Daniels & Walker, 2001). Negative loops counteract change.

Stock and flow diagrams represent the quantitative nature of the system. A stock is defined as a "quantity of something (such as the quantity of heat in a cup of coffee)" (Alessi, 2000), and is a time-point related system variable. A stock is represented by a rectangle (see Figure 3). A flow represents the rate of change (the rate of increase and/or decrease) of a stock. Flows are represented by pipes into or out of a stock. A valve can be seen on the pipe that controls the flow. They are time-interval related system variables. The clouds at the ends of the flows represent the boundaries of the system.

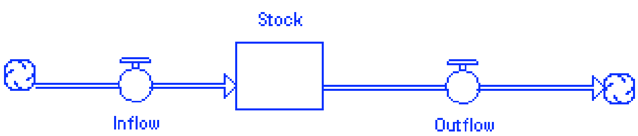


Figure 3: An example of a Stock and Flow Diagram (Sterman, 2000, p. 193)

System dynamics has been used to examine education (Ramsey & Ramsey, 2002), economics (Harvey, 2002), environmental sustainability (Saysel, Barlas, & Yenigun, 2002), to design interactive courseware (Spector & Davidsen, 1997), and to understand the process of implementing technology-enhanced learning environments in higher education (Stavredes, 2001). System dynamics has been used to study a variety of situations in natural resource management (e.g., Faust, Jackson, Ford, Earnhardt, & Thompson, 2004)); to model socio-environmental systems (Martinez Fernandez & Esteve Selma, 2004); situations of economy versus environment (Dudley, 2004; Woodwell, 1998); and systems involving the interplay of society, economy and the environment (such as tourism (Patterson, Gulden, Cousins, & Kraev, 2004) and tourist behaviour (Walker, Greiner, McDonald, & Lyne, 1999)). System dynamics modelling is suited to multidisciplinary problems because it can accommodate both quantitative and qualitative data. It is fundamentally interdisciplinary.

Models are representations of ideas, objects, events, processes or systems (Gilbert & Boulter, 1998), and are generally simplifications of reality (Coyle, 2000; D. Jonassen, 2000). Computer-based models allow complex systems to be represented efficiently and constructed in a relatively short amount of time. A computer-based model may be a better tool for learning because the assumptions of the system must be stated explicitly, allowing these assumptions to be criticised and compared (Forrester, 1971). In addition, parts of the system are able to be more easily visualised when using a model (Gilbert & Boulter, 1998).

Davies (2002) found that the features of a simulation that were important for student engagement were the complexity of the situation, the learning environment as a whole, navigational opacity, allowing sufficient time for engagement to develop, and allowing for cooperative learning. Some studies have shown that both model building exercises and learning with models can promote systems thinking, improve learning outcomes and student attitude toward the class (Friedman & McMillian Culp, 2001; Kiboss, Ndirangu, & Wekesa, 2004; Kurtz dos Santos, Thielo, & Kleer, 1997).

Models used in education have two main roles. The first is an analytic role, when models are used to simplify complex structures and the model is applied directly to a situation (Harre, 1999). The second role is an explanatory one, where models are used as representations for anything that cannot be observed naturally, such as theories (Harre, 1999). In either case, the use of models and technology provides an authentic learning experience for the students (Kelleher, 2000). Some studies have shown that both model building exercises and learning with models can promote systems thinking, improve learning outcomes and student attitude toward the class (Friedman & McMillian Culp, 2001; Kiboss et al., 2004; Kurtz dos Santos et al., 1997). Simulations are effective particularly in science because they allow students to develop hypotheses and test them (Woolsey & Bellamy, 1997). Once a model is created, it can be used as a trigger to explain behaviour or identify how the system relates to a larger system (Coyle, 2000).

The course

The majority of the course followed a collaborative, project-oriented pedagogy. Davies (2002) suggested that one of the features of a simulation that was important for student engagement was allowing for cooperative learning, prompting the investigation of collaborative learning in this study. Studies have found that students in cooperative learning groups outperformed individual learners in a biology subject (Singhanayok & Hooper, 1998), and in our own research learning about the environment with system dynamics models (Thompson & Reimann, 2007). Cooperative learning encourages interaction with the tool (Singhanayok & Hooper, 1998); supports a range of learning styles (Wang, Hinn, & Kanfer, 2001); and allows group members to explain concepts to each other (Kramarski, 2004), which is an important metacognitive strategy.

The course *Learning and Change a Systems View* was conducted for the first time in Semester 2, 2006. Postgraduate students participating in this subject learned about a number of different methods used to analyse and describe learning and change. The second half of the course had a focus on one of these methods – system dynamics modelling (SDM). In preparation for their group work students had three online training sessions on the basics of system dynamics modelling and systems thinking and on how to use Stella™. Instructions were pasted into the whiteboard available as part of the OLE described above, and students worked on this using the runtime version of Stella™. They could ask questions in the chat environment by referring to the number of the instruction. Students were then divided into groups of six and four and given a research-based and design-based task respectively. The research-based task consisted of a unit from *Modelling Dynamic Systems: Lessons for a First Course* (Fisher, 2005). The unit they were given was the Pronghorn Antelope, age-specific population study. This was chosen because the last section of this activity involves testing the model against real data. This was thought to be an excellent opportunity for the research team to experience how system dynamics modelling can be used in research. The design-based task was to find an existing Stella™ model online and to use the instructional elements of the software to make it suitable for use in an educational setting. This involved finding Stella™ models in a domain about which at least one of the team members was already knowledgeable, and then the team would decide which one to work on. Students were encouraged to investigate the educational options available in Stella™, such as the storytelling tool. Finally students were to implement the decision. Their final piece of work showed they had added features such as instructions, experiments, and storytelling (including sound). In order to do this well, students needed a good understanding of the system being modelled so that they could explain it in the storytelling feature and construct appropriate experiments for the users.

Students were expected to work on this in their groups online (synchronously and asynchronously). The output was to be a finished model representative of their group work, although they would not receive marks based on the model itself. Students were encouraged to take ownership of those elements that they felt most comfortable with (for example some students were obviously more comfortable with the modelling side and using Stella™, others with quality control or group management tasks). Students were given instructions regarding management of their teams. Part of this was that the students would be expected to coordinate their own work within their team. All of this was to be documented in Trac, (as discussed above). The emphasis was for students, whether in their individual or shared work, to show the processes that they were going through. This environment also allowed students to keep track of the hours that they were spending on this subject each week as it allowed work reports to be generated.

Observations

Modelling work

Due to the limited number of Stella licenses, not all students could be the modellers. The internal management of the teams meant that there was much discussion with regards to who took ownership of the modelling part of the activity. The aim of the activity had been that students would specialise in the aspects that they were comfortable with, and it was not intended that all students would embrace system dynamics modelling. However, it seemed that this did not happen, and, particularly in the research team, students were initially unable to understand what the modellers were doing unless they had done the same thing themselves. It was not until approximately half way through the activity when the tutor intervened, that the idea of providing a narrative to help with understanding the system was taken up by the team. This helped those students who were not actually modelling to understand the relationship between the stocks and flows in the model, to understand the decisions that had been made by the modellers, and to participate in the discussion. This was an enormous help to the problem solving that occurred within the online group.

Due to space limitations, we can only report here on the modelling outcomes of one team, the design team. The final product that the design team submitted had three main features (see Figure 4). The first is that they provided background information to help students understand the system before experiments were done using the model. Students who would use this resource were given the role of a manager of the area, and given an aim. The second feature was the development of the storytelling tool available in STELLA™. This involves the (complex) model being explained step by step. In this way, students using the resource would be able to understand the relationships between stocks and flows in the model, the implications, and the links.

The third feature was the introduction of two scenarios. The first was one in which the aim of the manager is to manage the deer population which is starving due to a lack of food. Students were given a starting point "a colleague has suggested…", and some instructions in terms of how they should start, and a reminder of the desirable outcomes. Some questions were also provided that revolve around the outcomes, encouraging the students to think about the graphs that are produced and what they mean, rather than just reproducing graphs. The second scenario follows a similar pattern in that the starting point and instructions were given.

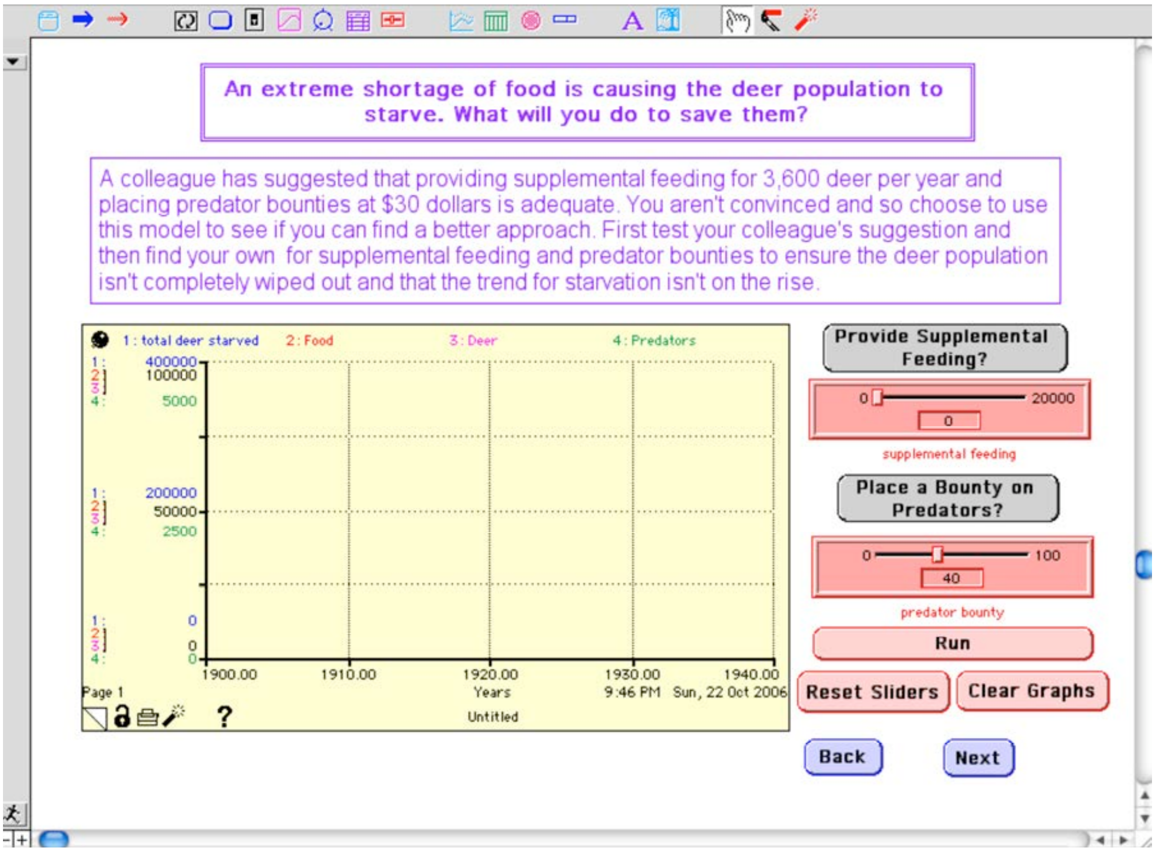


Figure 4: One of the scenarios developed

Issues that were apparent in their work concerned the practicalities of learning how to use unfamiliar software in a short amount of time. Small “tricks” such as the ability to have the two scenarios in the same file were too software specific for our students to be able to figure out in the time provided. Other issues concerned changing between mac and pc versions of the software, and their ability to add extra features such as pictures and sounds. Their ability, however, to understand the issues that were communicated by the model, and to be able to take that information and change it so that it was aimed at primary level students was outstanding. Their chats reveal that the decisions that they made were for those express purposes: “I thought that they got to do too much with the model - they had to interact with all the sliders. we could present them with maybe only one or two sliders to interact with given that these will be primary school learners”

The story telling tool was also used in a sophisticated way. The explanation of the model was done with a combination of detailed (where needed) and brief explanations of the links. There is no evidence in the chat logs that the students had any issues with understanding how the model fit together, or what the stocks and flows meant. These students were able to look at a model they had never seen before, understand the scenario, and alter it so that it could be explained to a primary school aged student, using software that they had had a brief experience with.

Analysing the chat conversations for group coordination activities

In order to elaborate on the decision making processes in the groups, we adopted the Decision Function Coding System (DFCS) from Poole and Holmes (1995). The DFCS presents a commonly established cognitive decision model with problem definition, orientation, solution and agreement/disagreement phases. While a lot of decision development theories build on models with fewer and simpler phases (Poole & Holmes, 1995), we chose the DFCS as it allows for more detailed decision path analysis. It has been applied in research in various modified forms to elaborate on decision function processes in different settings (Cho & Jonassen, 2002; D. H. Jonassen & Kwan, 2001).

The coding scheme in its original proposed form did not entirely suit our research questions and we customised it with regard to the omission of phasic markers and the simplification of categories. The modification resulted in a coding scheme with 6 main categories and 5 subcategories (see Table 1).

Table 1: Modified coding scheme on decision-making processes.

Code	Category	Definition
1	Problem definition	• Statements that define or state the causes behind a problem. • Statements that evaluate problem analysis statements.
2	Orientation	• Statements that attempt to orient or guide the group’s process. • Statements that reflect on or evaluate the group’s process or progress.
	Solution development	
3a	Solution analysis	• Statements that concern criteria for decision making or general parameters for solutions. • A direct reference to the solution must be given.
3b	Solution suggestion	• Suggestions of alternatives.
3c	Solution elaboration	• Statements that provide detail or elaborate on a previously stated alternative.
3d	Solution evaluation	• Statements that evaluate alternatives and give reasons, explicit or implicit for the evaluations – and therefore include a valuation.
3e	Solution confirmation	• Statements that state the decision in its final form or ask for final group confirmation of the decision. • Statements that concern decisions linked to intermediate results.
4	Nontask	• Statements that do not have anything to do with the decision task.
5	Simple agreement	• Only ‘on-topic’ agreement.
6	Simple disagreement	

One chat entry, which was approximately equivalent to a semantic unit, served as a unit of analysis. We coded 8 out of 14 chats that took place during the semester. After an initial coding was undertaken, a first inter-rater agreement was estimated and the coding scheme underwent a revision cycle. Training cycles constituted of coding, agreement and revision were done twice on data not retrieved from the course and once on the actual course data. Subsequently, the second rater coded about 25% of the coded data. The coding resulted in a inter-rater reliability of κ= .68, which is an acceptable agreement according to Banerjee et al. (1999). The frequencies are summarised in Figure 5.

A chi square test was performed on this data, and found to be significant ($\_^2$=240.15, p< .001). Further examination of the proportions of chats suggests patterns in the decision-making process. For one, a pattern of decision-making per session can be observed. For most sessions, non-decision tasks take up the highest proportion of chats, orientation the second highest proportion, the solution definition the third highest, then simple agreement, problem definition and simple disagreement. There are a couple of exceptions. The first is session 5, where the second highest proportion of chats revolved around the solution definition rather than orientation. The other is session 8, during which the highest proportion of chats was devoted to orientation, followed by non-decision task. In that week the simple disagreement came in fifth and problem definition sixth. During session 5, had almost completed the coordination phase of the activity, and tasks had been allocated to the members of the group. The second phase of their work was then begun. In this sense, session 5 can be seen as a completion week, much like session 8. In other words, for the majority of the time students had a particular pattern of communicating, which changed in the final stages of the activity.

We can also look at the proportion of time spent per session on each decision-function category. The greatest proportion of the chats were devoted to problem definition in the chat before the final chat in the first half, and also in the first chat after they organised the second part of their task. The least proportion of chat devoted to problem definition was in the first chat, and the last chat. Presumably, they were getting to know each other and completing their task respectively.

The greatest proportion of the chats were devoted to orientation in the third chat into the activity, followed by the first chat phase 2 of the activity. The least proportion of chat devoted to orientation was in the first chat, and the final chat in the first half. This may indicate that students took a couple of chats to get to a point of reiterating the problem, and then had a very intense first chat at the beginning of the second phase.

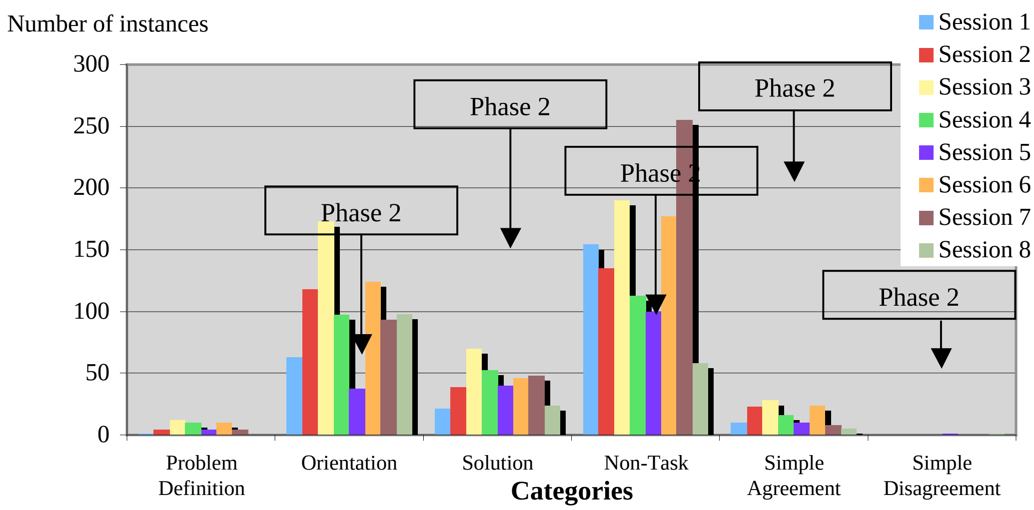


Figure 5: Chat data for the design team for each chat conducted

For the purposes of this analysis, the individually coded chats for the sections of solution development were combined. Students in the design group spent the highest proportion of chats discussing solution development on the third chat into the activity, with very little discussion before that. This was followed by the chat after (4^{th} session) and then the second chat in the second half of the activity. The least amount of time was spent in the final and first chats. This indicates that students took a couple of chats to get to a point of developing a solution to the problem, and then spent two chats developing the solution. There was little time spent on solution development in the first week or the last week, presumably they were engaged in other actions in the first chat, and had developed the solution by the final chat.

The non-decision tasks took up the highest proportion of chats overall. The majority of their non-decision proportion was on the second last chat of the entire activity, followed by the third chat. The smallest proportion of chats spent on non-decision tasks were on the final chats in both phases. This indicates that students were not in decision-making modes on the second chat of the second half, or the third chat. They also spent more time on decision-making tasks in their two "final" chats, presumably on final decisions with regards to allocating tasks etc.

Simple agreement and simple disagreement are the final codes that we will discuss. Students spent the highest proportion of their time in simple agreement in the third chat followed by the 6th session (which was the first chat of the second phase). The least proportion of time spent in simple agreement was the two final weeks of the total activity. Simple disagreement was the area that students spent the least proportion of chats in, in almost every session (except the final session). Actually, there were only two sessions where any chat was coded under this heading – this was session 5 and session 8. Both of these were the "final" sessions for each phase.

Conclusions

This paper showed some of the advantages of combining system dynamics modelling with collaborative activities in an online learning environment. Collaboration helps students to come to grips with the powerful but abstract notions endemic to system dynamics in particular and complexity sciences in general; and division of labour helps students to cope with the high demands of modelling as a research and learning method. Performing most of the relevant activities online makes it possible not only for students to participate from a distance, but also helps students teams to keep track of their work and produce records of their learning and interactions.

Trac allowed students to both asynchronously upload versions of models and provide comments and feedback, but also to manage their tasks, allocating tickets to members of the group. The chat environment provided an area to learn in, initially, as weel as to meet each other. The tools also provided us, as researchers, with a rich source of data, and therefore a unique view into the group work. This allowed us to assess the progression of the group as they were working, but also to analyse the decision making patterns in the group afterwards. Preliminary results have identified a distinct decision function pattern (Reimann, Thompson, & Weinel, 2007). Further analysis of this, as well as extending this analysis to other complex tasks, are areas that our research will extend.

Acknowledgements

Thank you to Lena Hofmann for coding the data.

References

Alessi, S. (2000). The application of system dynamics in elementary and secondary school curricular. Retrieved 29 October 2003, from http://www.c5.cl/ieinvestiga/actas/ribie2000/charlas/alessi.htm

Banerjee, M. B., Capozzoli, M., McSweeney, L., & Sinha, D. (1999). Beyond Kappa: A Review of Interrater Agreement Measures. *The Canadian Journal of Statistics, 27*(1), 3-23.

Cho, K.-L., & Jonassen, D. H. (2002). The effects of argumentation scaffolds on argumentation and problem solving. . *Educational Technology, Research and Development, 50*(3), 5-22.

Coyle, G. (2000). Qualitative and quantitative modelling in system dynamics: some research questions. *System Dynamics Review, 16*(33), 225-244.

Daniels, S. E., & Walker, G. B. (2001). *Working through environmental conflict: the collaborative learning approach*. Westport: Praeger Publishers.

Dudley, R. G. (2004). Modeling the effects of a log export ban in Indonesia. *System Dynamics Review, 20*(2), 99-116.

Faust, L. J., Jackson, R., Ford, A., Earnhardt, J. M., & Thompson, S. D. (2004). Models for management of wildlife populations: lessons from spectacled bears in zoos and grizzly bears in Yellowstone. *System Dynamics Review, 20*(2), 163-178.

Fisher, D. M. (2005). *Modeling Dynamic Systems: Lessons for a First Course*.

Forrester, J. W. (1961). *Industrial Dynamics*. Cambridge, Massachusetts: The M.I.T. Press.

Forrester, J. W. (1971). *World Dynamics*. Massachusetts: Wright-Allen Press, Inc.

Friedman, W., & McMillian Culp, K. (2001). *Evaluation of the CoreModels Project*. Maryland: Education Development Center/Center for Children & Technology.

Gilbert, J. K., & Boulter, C. J. (1998). Learning Science through models and modelling. In B. J. Fraser & K. G. Tobin (Eds.), *International Handbook of Science Education* (Vol. 2, pp. 53-66). Dordrecht: Kluwer Academic Publishers.

Harre, R. (1999). Models and Type-hierarchies: Cognitive Foundations of Iconic Thinking. In R. Paton & I. Neilson (Eds.), *Visual Representations and Interpretations* (pp. 97-111). London: Springer-Verlag London Limited.

Harvey, J. T. (2002). Keynes' Chapter 22: A System Dynamics Model. *Journal of Economic Issues, XXXVI*(2), 373-381.

Jacobson, M. J., & Wilensky, U. (2006). Complex systems in education: Scientific and educational importance and implications for the learning sciences. *The Journal of the learning sciences, 15*(1), 11-34.

Jonassen, D. (2000). *Computers as mindtools for schools. engaging critical thinking.* (2nd ed.): Merrill.

Jonassen, D. H., & Kwan, H. I. (2001). Communication patterns in comoputer mediated versus face-to-face group problem solving. *Educational Technology, Research and Development, 49*(1), 35-51.

Kelleher, R. (2000). A review of recent developments in the use of information communication technologies (ICT) in science classrooms. *Australian Science Teachers Journal, 46*(1), 33-38.

Kiboss, J. K., Ndirangu, M., & Wekesa, E. W. (2004). Effectiveness of a Computer-Mediated simulations program in school biology on pupils' learning outcomes in cell theory. *Journal of Science Education and Technology, 13*(2), 207-213.

Kramarski, B. (2004). Making sense of graphs: does metacognitive instruction make a difference on students' mathematical conceptions and alternative conceptions? *Learning and Instruction, 14*, 593-619.

Kurtz dos Santos, A. C., Thielo, M. R., & Kleer, A. A. (1997). Students modelling environmental issues. *Journal of Computer Assisted Learning, 13*, 35-47.

Layman, L., Williams, L., Damian, D., & Bures, H. (2006). Essential communication practices for extreme programming in a global software develpment team. *Information and Software Technology, 48*, 781-794.

MacMillan, J., Entin, E. E., & Serfaty, D. (2004). Communication overhead: the hidden cost of team cognition. In E. Salas & S. M. Fiore (Eds.), *Team cognition* (pp. 61-82). Washington, DC: American Psychological Association.

Martinez Fernandez, J., & Esteve Selma, M. A. (2004). The dynamics of water scarcity on irrigated landscapes: Mazarron and Aguilas in south-eastern Spain. *System Dynamics Review, 20*(2), 117-137.

Patterson, T., Gulden, T., Cousins, K., & Kraev, E. (2004). Integrating environmental, social and economic systems: a dynamic model of tourism in Dominica. *Ecological Modelling, 175*, 121-136.

Poole, M. S., & Holmes, M. E. (1995). Decision Development in Computer/Assisted Group Decision Making. *Human Communication Research, 22*(1), 90-127.

Ramsey, D. C., & Ramsey, P. L. (2002). Reframing the perfectionist's Catch 22 dilemma: a systems thinking approach. *Journal for the education of the gifted, 26*(2), 99-111.

Reimann, P., Thompson, K., & Weinel, M. (2007). *Decision making patterns in virtual teams.* Paper presented at the International Conference on Computers in Education, Hiroshima.

Ripochet, G., & Sansonnet, J.-P. (2006). Experiences in automating the analysis of linguistic interactions for the study of distributed collectives. *Computer Supported Cooperative Work, 15*, 149-183.

Saysel, A. K., Barlas, Y., & Yenigun, O. (2002). Environmental sustainability in an agricultural development project: A system dynamics approach. *Journal of Environmental Management., 64*(3), 247-260.

Singhanayok, C., & Hooper, S. (1998). The effects of cooperative learning and learner control on students' achievement, option selections, and attitudes. *Educational Technology, Research and Development, 46*(2), 17-32.

Spector, J. M., & Davidsen, P. I. (1997). Creating Engaging Courseware Using System Dynamics. *Computers in Human Behavior, 13*(2), 127-155.

Stavredes, T. (2001). A system dynamics evaluation model and methodolgy for instructional technology support. *Computers in Human Behavior, 17*, 409-419.

Sterman, J. D. (2000). *Business Dynamics. Systems Thinking and Modeling for a Complex World.*: McGraw-Hill Higher Education.

Thompson, K., & Reimann, P. (2007). *Do school students learn more about the enviornment from a system dynamics model by themselves or with a partner?* Paper presented at the The 25th International Conference of the System Dynamics Society 50th Anniversary Celebration, Boston.

Ullman, A., Peters, D., & Reimann, P. (2005). *Developing a research supportive web-based learning System.* Paper presented at the ODLAA Conference 2005, Adelaide.

Walker, P. A., Greiner, R., McDonald, D., & Lyne, V. (1999). The tourism futures simulator: a systems thinking approach. *Environmental Modelling & Software, 14*, 59-67.

Wang, X. C., Hinn, D. M., & Kanfer, A. G. (2001). Potential of computer-supported collaborative learning for learners with different learning styles. *Journal of Research on Technology in Education, 34*(1), 75-85.
Woodwell, J. C. (1998). A simulation model to illustrate feedbacks among resource consumption, production, and factors of production in ecological-economic systems. *Ecological Modelling, 112*(2-3), 227-247.
Woolsey, K., & Bellamy, R. (1997). Science education and technology: opportunities to enhance student learning. *The Elementary School Journal, 97*(4), 385-399.
Zacklad, M. (2006). Documentarisation processes in documents for Action (DofA): The status of annotations and associated cooperation technologies. *Computer Supported Cooperative Work, 15*, 205-228.

Peter Reimann
p.reimann@edfac.usyd.edu.au
Kate Thompson
k.thompson@edfac.usyd.edu.au
Miriam Weinel
m.weinel@edfac.usyd.edu.au

Please cite as: Reimann, P., Thompson, K. & Weinel, M. (2007). Collaborative learning by modelling: observations in an online setting. In *ICT: Providing choices for learners and learning. Proceedings ascilite Singapore 2007*. http://www.ascilite.org.au/conferences/singapore07/procs/reimann.pdf