

Supporting the teacher in educational robotics classes: work in progress

Ilkka Jormanainen & Antony Harfield

Department of Computer Science and Statistics, University of Joensuu, Finland
{ilkka.jormanainen, antony.harfield}@cs.joensuu.fi

Abstract: This paper presents work-in-progress on developing an environment to support the teacher in educational robotics classes. Building on previous research that implemented an agent architecture to collect information about the interactions between student and robot, we explore approaches that the teacher can use to observe and to decide when and how to intervene in a classroom situation. As an alternative approach to traditionally developed programs, Empirical Modelling is introduced as offering an open-ended environment in which the teacher can construct an artefact that reflects their personal understanding of the rich situations that can arise in a robotics class.

Keywords: Educational robotics, Empirical Modelling, teacher support.

Introduction

Educational technology often focuses on making learning engaging and beneficial for the student. However, in most classroom situations it is the teacher who plays a critical role in making learning activities rewarding and meaningful for the students. Successful teachers know when and how to give students guidance based on observations made in the classroom. However, when ICT is introduced to the classroom, it is more challenging to observe the interactions between student and computer, and more difficult to determine when and how to intervene in the learning process.

Low-cost and highly accessible educational robot kits, such as Lego Mindstorms, are becoming more popular in classrooms and hands-on learning environments [7]. The use of educational robotics requires new kinds of classroom settings and teachers may have to change their teaching methods accordingly. Otherwise, the use of educational robotics may lead to negative results, as reported in [3]. In a typical classroom setting especially at the elementary level, a teacher might have 30-40 children to teach and the students are typically divided into groups of 3-4 students. Each group is provided with one set of educational robotics and group members working together within a given topic or project. A typical educational robotics project follows an iterative cycle of building, programming, testing, and evaluation. It is typical that groups proceed differently, being in different phases of the cycle at the same time. Furthermore, the open-ended nature of robot building can lead to students taking completely different approaches to an activity. These factors make it especially difficult for teachers to be aware of when intervention is required.

The main research agenda of our work is to answer: how can a teacher in an educational robotics classroom observe and understand the problems and progress of the students? In a previous paper [6], we have developed an architecture for collecting information on students' interaction with the computer when working on educational robotics projects. The architecture is designed to collect data about students' actions in the

robot programming environment and deliver these observations to the teacher. Our approach does this by utilising software agents that run in the background of the programming environment or the robot itself. The shaded area of Figure 1 illustrates the architecture (described fully in [6]). The current work, which is introduced in this paper, is concerned with answering: how can the teacher use the collected information to inform their intervention in the students interactions with the robots in the classroom?

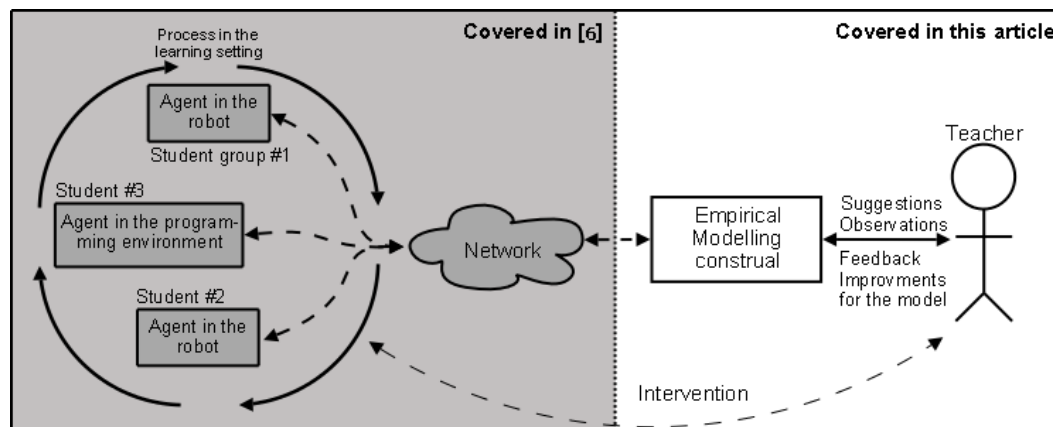


Figure 1. A framework for supporting teachers in educational robotics classes.

1. Approaches to constructing a teacher support tool

1.1 The traditional software development approach

In traditional software development it would be appropriate to consider the program requirements that satisfy the needs of the teacher. In this case, given that we have access to collected data from the students' interactions, the teacher may have some idea of what might be needed of the support tool, for example: a) detecting when a student has a problem; b) analysing the actions of a student in order to understand problems or mistakes; and c) observing the overall progress of the students in the classroom. A tool that fits these requirements (and implements a similar agent-based information collection architecture) has been developed by George & Despres for educational robotics classes [4]. When their system detects that a student has a problem, then either the system suggests a course of action to the student or the teacher is informed so they can offer guidance to the student.

However, as this tool (and others such as described by Trojahn dos Santos et al. [8]) are programs with predefined specifications, they only offer the teacher a set of preconceived options for observing actions, detecting problems and analysing progress of the student. If a teacher discovers a new problem that students are encountering then there is little chance that this could be incorporated into the tool unless the teacher had the time, inclination and programming skills. Although such tools may offer some teacher support, what a teacher really seeks is an environment that enables more of the open-ended exploration and sense-making that is evident in a traditional classroom situation. It is for this reason that Empirical Modelling is introduced as an alternative to using a tool that has been implemented by traditional software development methods.

1.2 The Empirical Modelling approach

Empirical Modelling (EM) is a collection of principles and tools that have been developed by Beynon and Russ at the University of Warwick over the last 20 years [9]. Whereas traditional computer scientists view the computer as a machine for executing formally

specified programs, the alternative approach to computing suggested by EM is based on the idea that the computer represents a state corresponding to a concrete situation in the world. EM activity is concerned with constructing and engaging with concrete situations using computer-based artefacts. Using EM, such artefacts offer much of the flexibility of human interaction in the world when compared to the constrained behaviour of a computer program. For this reason, EM artefacts often invoke personal, subjective, provisional and tacit interpretations that reflect the open-ended nature of human interaction [5].

Whereas a traditional program gives the teacher information about a predetermined set of observations, an EM artefact enables the teacher to build up their own set of observations on the concrete classroom situation. Typically a teacher in an educational robotics class depends on a personal, subjective, provisional and tacit understanding in order to guide students through a learning activity, and as such, the teacher may wish to record a wide range of observations in a computer-based artefact (e.g. personal problems that particular students are having, actions that the teacher sees subjectively as bad practice, provisional observations of situations where students may get stuck, or tacit knowledge of good robot building practice). The principles of EM offer some support for interacting with artefacts in this manner (as demonstrated, for example, in research on using EM for supporting the teaching of databases [1]) and therefore provide a suitable environment from which to consider supporting the teacher in educational robotics classes.

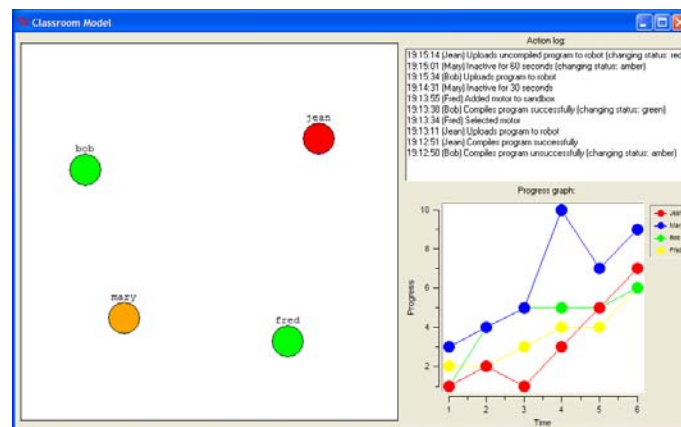


Figure 2. A screenshot of classroom model.

2. Initial results on applying EM

In order to demonstrate the potential for EM to support the teacher in an educational robotics classroom, a simple artefact has been constructed which integrates with the architecture illustrated in Figure 1. The artefact is a screen-based environment in which the teacher can observe possible problems, analyse the students' actions, and get an overall view for the progress of the students. These are typical functions that a similar program built with traditional programming tools could be designed to achieve. The focus of this discussion is demonstrating what a teacher would *not* be able to do with the program.

A screen-shot of the artefact in a particular classroom situation is shown in Figure 2. On the left side of the screen is one possible view of the students, represented by circles, in a classroom where the colours indicate levels of difficulty a student may be experiencing (green for no problems and red for serious problems). It is likely that not all teachers would be content with this simple classification of student status into 3 colours. With an EM artefact, the teacher may change the definitions so that a new colour is introduced to reflect a student performing an action of which the teacher disapproves. Changes of this nature can be accomplished by the teacher in the classroom, while the artefact is running and without restarting the environment. The teacher is not restricted to only changing colours, they may

wish to introduce completely new dimensions to the visualisation, such as making the size of the circle dependent on the student's perceived progress or on the number of lines in their code. The teacher may also like to add indicators to detect completely new types of error they have encountered during the session. For example, the teacher might notice that students sometimes press the upload button after an unsuccessful compile (which is probably undesirable) and so they could add definitions to detect such occurrences. The nature of EM activity is such that any part of the artefact can be manipulated because the artefact is made up of definitions of a similar form. Thus the teacher could potentially extend the environment itself, perhaps by adding another graph to the right side of the screen to show the number of times the student has attempted to run or test their robot (hence giving some indication of how the students are progressing).

More comprehensive examples will be explored in future work, but this short introduction illustrates the flexibility of the EM approach that can potentially satisfy the demands of an educational robotics classroom. It is also important to note that an EM environment in no way forces 'artificial intelligence' on the teacher. Unlike a program, an artefact does not determine when a student has made an error, but instead, it is up to the teacher to use the artefact to decide how to guide the student. In this way, EM empowers the teacher to make good teaching decisions, rather than replacing the teacher.

3. Closing discussion

Teachers working in educational robotics classes often face difficulties to follow students' progress. Tutoring systems, which have been constructed with traditional programming environments can provide support for teachers in trivial cases, when students' errors have been preconceived. These systems do not offer very many possibilities to adapt the system when students proceed unexpectedly. Furthermore, all teachers work in different ways so we need to be able to let them to personalise their teaching support. EM provides tools to build an open-ended support system for teachers as introduced in this paper.

This paper completes the work on a framework for teacher support shown in Figure 1 and partly implemented in [6]. The first empirical experiments will be carried out in Kids' Club [2] at the University of Joensuu during May 2008, and further experiments and publications can be expected in Autumn 2008.

References

- [1] Beynon, W.M. & Harfield, A. (2005). Empirical Modelling in Support of Constructionist Learning: a Case Study from Relational Database Theory. In *Proc. of ICALT'05*, 396-398.
- [2] Eronen, P.J., Sutinen, E., Vesisenaho, M. & Virnes, M. (2002). Kids' Club as an ICT-based learning laboratory. In *Informatics in Education*, 1(1), 61-72.
- [3] Fagin, B.S. & Merkle, L. (2002). Quantitative analysis of the effects of robots on introductory Computer Science education. *J. Educ. Resour. Comput.* 2 (4).
- [4] George, S. & Despres, C. (1999). A multi-agent system for distance support in educational robotics. In *Proc. of International Conference on Telecommunication for Education and Training*, 344-353.
- [5] Harfield, A. (2008). Empirical Modelling as a new paradigm for educational technology. PhD thesis, Department of Computer Science, University of Warwick, UK.
- [6] Jormanainen, I., Zhang, Y., Kinshuk, & Sutinen, E. (2007). Pedagogical Agents for Teacher Intervention in Educational Robotics Classes. In *Proc. of DIGITEL 2007*, 49-56.
- [7] Sklar, E., Parsons, S. & Stone, P. (2004). Using RoboCup in University-Level Computer Science Education. *J. Educ. Resour. Comput.* 4 (2), 1-21.
- [8] Trojahn dos Santos, C., Frozza, R., Dhamer, A. & Gaspary, L. P. (2002). DÓRIS – Pedagogical Agent in Intelligent Tutoring Systems. In *Proc. Of ITS 2002*, 91-104.
- [9] The Empirical Modelling Website. (2008). URL: <http://www.dcs.warwick.ac.uk/modelling>

Student Perceptions of the Effectiveness System Dynamics-based Interactive Learning Environments: a Case Study

Qudrat-Ullah H^a

^a*School of Administrative Studies, York University, Canada*
hassanq@yorku.ca

Abstract: The aim of this article are (i) to describe the construction and the integration of an interactive learning environment in two educational settings, and (ii) to explore how undergraduate business students perceive SDILEs and SDILEs-based course approach. This research draws on data, 78 valid respondents, obtained from courses in undergraduate business program. Results of this study suggest that students enrolled in a SDILE-based course do indeed perceive important learning benefits and educational value. However, there is a need for additional resources to be developed and deployed to harness fully the benefits of experiential learning provided through SDILE-integrated course approach.

Keywords: System dynamics, simulations, ILEs, classroom, action research

Introduction

In essence, Simulations in general and computer simulation-based interactive learning environments (ILEs) in particular are developed and used to improve people's decision making in the context of the dynamic complexity of business settings by facilitating user learning [2, 15, 19, 21]. People learn from experience. Learning is the process whereby knowledge is constructed by the transformation of experience [13]. Simulations in general and ILEs in particular are one form of experiential learning. In an ILE session subjects make a series of decisions and have access to the instantaneous feedback. Subjects also have the opportunity to evaluate and reflect on their performance in the after-the-simulation debriefing session. Despite the promising potential and an increasing interest, ILEs benefit to the learning process in the classroom has rarely been investigated empirically.

From a methodological point of view, this study draws on data gathered based on action research in authentic educational settings [22].

1. SDILEs and Learning with SDILEs

1.1 Background Concepts

We use "ILEs" as a term sufficiently general to include microworlds, management flight simulators, DSS, learning laboratories, and any other computer simulation-based environments – the domain of these terms is all forms of action for the facilitation of learning in complex, dynamic environments. An ILE consists of three components (i) a computer simulation model to adequately represent the domain [4, 12, 14], (ii) a user interface capable of allowing the decision makers to make decisions and access the

feedback on interactive basis, and (iii) a human coach responsible for conducting briefing and debriefing sessions [4, 5, 16]. When an ILE's underlying simulation model is based on system dynamics methodology [8] we call that ILE as SDILE. Some popular SDILEs are People Express [25], FishBankILE [20], and Healthcare Microworld [20].

1.2 *Learning with SDILEs*

Learning about complex, dynamic tasks does not happen easily. There are some fundamental barriers to developing expertise in dynamic tasks: (1) *dynamic complexity*: our limited ability to understand the impact of time delays between our actions and their consequences coupled with the interactions between feedback loops that are multiple and non-linear in character and are ever present in most of the real world managerial tasks, (2) *information availability and quality limitations*: information we estimate, receive, and communicate is often oversimplified, distorted, delayed, biased, and ambiguous, (3) *information processing limitations*: when it comes to decision making people generally adopt an event-based, open-loop view of causality, ignore feedback processes, perceive flawed cognitive maps of the causal structure of the systems, fall prey to judgmental errors and biases, defensive routines [24].

1.2.1 *Learning and Decision Support through System Dynamics Simulation Model*

The core of SDILE is a system dynamics based simulation model [8]. System dynamics based models have strengths to map (i) the multiple stakeholders' interests, (ii) available but limited resources, and (iii) decisions at different levels in the organization—a general characterization of the most of the task systems e.g., health care system, education system, energy system etc. The significance of the modeling capabilities of this methodology lies in its contribution to our understanding of the structure and behavior of task systems [23].

1.2.2 *Learning and Decision Support through the User Interface Design*

For effective decision making in complex, dynamic tasks, decisions makers must acquire some reasonably precise notions of relationships among key task variables and develop an understanding of the most influential delays and feedback loops in the task system [6]. System dynamics methodology provides powerful tools like causal loop diagrams and stock and flow structures, to represent qualitatively the link between structure and behavior of the task (for excellent illustrations please see [3]).

1.2.3 *Learning and decision support through human support*

In an SDILE session, decisional aids can be provided at three levels: pre-task, in-task, and post-task levels. Pre-task level decisional aids can be conceptualized as information provided by the human tutor to a decision maker regarding the model of the task prior to performing the task [5, 9]. In-task decisional aids attempt to improve the individuals' decision-making performance by (i) making the task goals explicit at early stages of learning, (ii) helping them keep track of goals during the task, and (iii) providing them with diagnostic information. Post-task level decisional aids aim at improving performance by providing the decision-makers an opportunity to reflect on their experiences with task [4, 5].

2. Method

This study draws on data gathered through action experiments in authentic educational settings [1, 13, 22]. Data captured by the FishBankILE program, and questionnaire and qualitative data obtained in these settings serve the purpose to explore whether the integration of SDILEs in classroom settings generates learning benefits. Data collection occurred between March 2006 and March 2007.

2.1 Measurements

Subjects' task performance (TP) was measured by the cumulative profits made by each team over a period of 30 years plus the remaining resource (fish) value in the final year. TP was automatically recorded by a programmed module of FishBankILE. To explore the perceived value of SDILEs, a questionnaire was adopted from Romme's work on the evaluation of the learning benefits of microworld simulation [21].

3. Results

A total of 33 students out of 36 students who took the undergraduate junior level course responded with completed questionnaires. Consequently, the data of only these 33 is stored and used in this study. The response rate for the undergraduate senior level is 96%. The responses from both groups are summarized in Table 1.

Table 1: Perceived values by juniors and seniors

	Juniors (n = 33)		Seniors (n = 45)		Difference between seniors and juniors
	Mean	SD	Mean	SD	Mann-Whitney (significance)
Using FishBankILE is					
1. fun	3.12	0.76	3.56	0.85	-1.155 (0.248)
2. pleasant	3.15	0.77	3.18	0.42	-1.964 (0.050)
3. exciting	3.10	0.86	3.12	0.41	-1.089 (0.978)
4. enjoyable	3.10	0.81	3.26	0.56	-1.000 (0.317)
5. is easy to use	4.10	0.66	4.66	0.72	-1.481 (0.139)
The SDILE, FishBankILE					
has well written user manual					
	3.66	0.56	3.87	0.80	-1.225 (0.221)
7. has user-friendly interface	3.80	0.66	3.96	0.80	-0.980 (0.327)
8. represents real business situation	2.98	0.46	2.90	0.44	-1.112 (0.266)
The information system					
9. has effective on-line help	4.10	0.66	4.23	0.34	-1.582 (0.114)
10. provides immediate & useful feedback	3.66	0.64	3.88	0.80	-0.671 (0.502)
The assignments related to FishBankILE were					
11. clear and relevant	3.68	0.44	3.98	0.56	-1.149 (0.250)
12. interesting	3.86	0.72	3.88	0.80	-0.872 (0.383)
We collaborated successfully by					

13. sharing ideas	3.12	0.44	3.88	0.56	0.000 (1.000)
14. appreciating ideas of fellow team members	3.10	0.51	3.97	0.66	-0.592 (0.554)
15. crafting decision strategy by consensus	3.55	0.62	3.78	0.44	-1.155 (0.248)
Feedback by the instructor at					
16. pre-task level was useful and aroused my interest	3.44	0.71	3.56	0.56	-2.309 (0.021)
17. in-task level was useful	3.02	0.45	3.24	0.80	-1.549 (0.121)
18. debriefing was very useful	3.44	0.44	3.66	0.52	-1.225 (0.221)
Overall, the SDILE course approach					
19. is more interesting than traditional textbook learning	3.88	0.44	4.12	0.46	-0.592 (0.545)
20. is more exciting than traditional textbook learning	3.91	0.61	4.06	0.44	-1.155 (0.248)
21. is more innovative than traditional textbook learning	4.01	0.45	4.22	0.66	-2.449 (0.014)
22. adds value for learning	4.22	0.56	4.44	0.46	-0.218 (0.827)
23. has my strong recommendations	4.20	0.76	3.45	0.66	-1.732 (0.083)
24. adds value relative to real world experience	3.61	0.66	4.34	0.46	-0.289 (0.773)

Subjects' actual task performance (TP) in the simulation task, FishBankILE, is given in Table 2.

Table 2: Actual task performance by teams of juniors and seniors

	Juniors (n = 71)	Seniors (n = 52)	Difference between seniors and juniors
TP	-0.65 (0.43)	-0.39 (0.33)	p= 0.009

TP = 0 means that the team's performance is equal to the bench mark. Standard deviation values are given in parenthesis.

4. Discussion and Conclusions

Although the majority of university classrooms are still dominated by traditional teaching methods such as lectures and discussions, the use of computer simulations in instruction is on the rise [7, 11, 15]. The widespread use of computer simulation based learning environments is concomitant with increasing attention to experiential learning methods.

The results described in the previous section also suggest that to accrue the learning benefits from a SDILE-based course a number of resources have to be created and provided. These resources include a well-written and self-explanatory user manual, a user-friendly interface with easy-to-use help and information systems, a related business case based simulation model, relevant simulation assignments that motivate students to design and test their decision strategies in the simulated environment, and effective feedback at all levels i.e., pre-task, in-task, and post-task level, by the facilitator. Other studies have reported similar observations that reinforce the need for these resources [11, 17].

The difference with regard to perception of use of FishBankILE as a pleasant experience can be explained in terms of the background education course work. Senior

students have accumulated substantial business knowledge through a variety of courses, case studies, project work and internships. By itself, this is sufficient incentive for college/university administrators and faculty to continue to emphasize the integration of simulation technology into most facets of postsecondary education.

References

- [1] Argyris, C. (1993). Knowledge for action: A guide to overcoming barriers to
- [2] Bakken, B. E. (1993). Learning and Transfer of Understanding in Dynamic Decision Environments. Unpublished doctoral dissertation. MIT, Boston.
- [3] Davidsen, P. I. (1996). Educational features of the system dynamics approach to modeling and simulation. *Journal of Structural Learning*, 12(4), 269-290.
- [4] Davidsen, P. I., (2000). Issues in the design and use of system-dynamics-based interactive learning environments. *Simulation & Gaming*, 31(2), 170-177.
- [5] Davidsen, P. I., & Spector, J. M., (1997). Cognitive complexity in system dynamics based learning environments. International system dynamics conference, Bogacizi University Printing Office Istanbul, Turkey.
- [6] Dörner, D. (1980). On the difficulties people have in dealing with complexity. *Simulations and Games*, 11, 8-106.
- [7] Faria, A. J. (1987). A survey of the use of business games in academia and business. *Simulation & Gaming*, 18(2), 207-224.
- [8] Forrester, J. W. (1961). *Industrial dynamics*, Productivity Press, Cambridge, MA.
- [9] Gröbler, A, Maier, F. H., & Milling, P. M. (2000). Enhancing learning capabilities by providing transparency in transparency. *Simulation & Gaming*, 31(2), 257-278.
- [10] Hirsch, G.B., Immediato, J.M., & Kemeny, M. (1997). *Creating Integrated Care and Healthier Communities Microworlds*. Cambridge, Mass: Innovation Associates.
- [11] Holweg, M., & Bicheno, J. (2002). Supply chain simulation – a tool for education, enhancement and endeavour. *International Journal of Production Economics*, 78, 163-175.
- [12] Homer, J. B., & Hirsch, G.B. (2006). System dynamics modeling for public health: Background and opportunities. *American Journal of Public Health*, 96 (3), 452-458.
- [13] Klabbers, J. H. G. (2000). *Gaming & Simulation: Principles of a science of design*.
- [14] Kriz, W. C. (2003). Creating effective learning environments and learning organizations through gaming simulation design. *Simulation & Gaming*, 34(4), 495-511.
- [15] Lane, D. C. (1995). On a resurgence of management simulations and games. *Journal of the Operational Research Society*, 46, 604-625
- [16] Ledrman, L. C. (1992). Debriefing: towards a systematic assessment of theory and practice. *Simulation & Gaming*, 23(2), 145-160.
- [17] Marsick, V. J., & O'Neil, J. (1999). *The many faces of action learning*. Management
- [18] Nagi, E. W. T. (2007). *Learning in introductory e-commerce: A project-based teamwork organizational change*. San Francisco: Jossey-Bass.
- [19] Qudrat-Ullah, H., & Karakul, M. (2007). *Decision Making in Interactive Learning Environments: Towards an Integrated Model*. *Journal of Decision Systems*, (forthcoming).
- [20] Qudrat-Ullah, H., Saleh, M. M., & Bahaa, E. A. (1997). Fish Bank ILE: An interactive learning laboratory to improve understanding of 'The Tragedy of Commons'; a common behaviour of complex dynamic systems. Proceedings of 15th International System Dynamics Conference, Istanbul, Turkey.
- [21] Romme, A. G. L. (2003). Learning outcomes of microworlds for management education. *Management Learning*, 34, 51-61.
- [22] Romme, A. G. L. (2004). Perceptions of the value of microworld simulation: Research note. *Simulation & Gaming*, 35 (3), 51-61.
- [23] Sengupta, K. & Abdel-Hamid, T. (1993). Alternative concepts of feedback in dynamic decision environments: An experimental investigation. *Management Science*, 39, 411-428. *Simulation & Gaming*, 34(4), 569-591.
- [24] Sterman, J. D. (1994). Learning in and around complex systems. *System Dynamics Review*, 10 (2-3), 291-323.
- [25] Sterman, J. D. (1988). *People Express Management Flight Simulator*. Software and briefing book. Available from J. D. Sterman, Sloan School of Management, 50 Memorial Drive, MIT, Cambridge, MA 02142, U.S.A.
- [26] Sterman, J. D. (1989). Modeling managerial behavior: Misperceptions of feedback in a dynamic decision making experiment. *Management Science*, 35, 321-339.