Educational Robots as Collaborative Learning Objects for Teaching Computer Science

R. Burbaite, V. Stuikys, R. Damasevicius
Software Engineering Department, Kaunas University of Technology, Studentu St. 50–438, LT-51368 Kaunas, Lithuania
renata.burbaite@stud.ktu.lt, vytautas.stuikys@ktu.lt, robertas.damasevicius@ktu.lt

***Abstract*—We present the constructionism-based approach towards using collaborating educational robots for teaching the abstract concepts of Computer Science (CS) such as task decomposition. We present the collaborative robot-based e-learning environment, which enables us to implement the principles of constructionism and collaborative learning for teaching students how to solve CS problems using LEGO multi-robots as tangible Collaborative Learning Objects (CLOs). We extended the existing approaches by a) providing a framework of robotic CLO based learning environment; and b) demonstrating the use of task decomposition and allocation principles for teaching CS algorithms and programming.**

I. Introduction

In the last two decades, educational robots contribute to increase learning achievements in teaching of science, technology, engineering and mathematics (STEM) [1]. Technology-based models and methods of teaching and learning are very helpful, because they "*provide many alternative ways of teaching, which learners select as they engage in their educational experiences*" [2]. Particularly, the educational robotics technologies enable students to control the behaviour of a tangible model (robot) by means of a virtual environment in which students investigate everyday phenomena [3].

Robotics provides an ideal medium for advancing creativity and computing literacy in young people as it integrates both hardware and software and provides tangible means for representing highly abstract ideas of Computer Science (CS). From the pedagogical point of view, robot-based e-learning environments allow enforcing student engagement in problem solving, increasing student academic achievements, and enable the use of collaborative learning and teaching methods [1]. From the technological point of view, the approach has a strong technical support [4]. From the social point of view, educational robotics represents a shift in teaching curricula towards teaching students how to manipulate digital devices that surround them for their own needs [5]. Although robotics is an excellent tool for teaching and learning, and an engaging topic for all students, the pedagogy of teaching using robots is still in its infancy [6].

Out of a plethora of pedagogical theories and approaches available to a modern teacher, *constructionism* [7] has emerged as a promising approach for educational robotics [8]. Constructionism is a learning theory that considers design as a part of the building process that involves *de-construction* and *re-construction* of units of knowledge [9] rather than just construction. Here we interpret the "*unit of knowledge*" as it is understood in CS, i.e., as an algorithm or a computer (or robot control) program.

The main tenets of constructionism are 1) learning by designing meaningful projects, 2) manipulating with concrete tangible objects to build and explore the world, 3) integration of concepts from different realms of knowledge, and 4) self-reflection [10]. In CS education, the ultimate aim of constructionism is the advancement of computing literacy in the broadest sense, i.e. the ability to create and creatively adapt computer hardware and software to communicate and disseminate ideas rather than plain use [11]. Constructionism implies the need for setting up computational environments as a technological background to support learning-by-doing. Components of such environments such as tangible things, digital media and software tools are the basis of such knowledge de-construction and re-construction operations.

In this paper we demonstrate how an abstract concept of task decomposition can be visualized and made more understandable using robotics. Task decomposition is a common activity in CS. It is used in a task solving process and involves the division of a complex task into a set of smaller, less complex and more manageable sub-tasks to reduce complexity and to allow solving it iteratively. However, students often have difficulties with understanding task decomposition and allocation principles. We claim that tasks can be represented using robots and the relationship between tasks and sub-tasks can be represented and demonstrated tangibly using collaborating robots. Here collaborating robots are understood as *"autonomous robots that pursue joint goals"* [12]. Collaborating robots have been used before to explain key concepts in a multi-agent system [13], to combine and advance artificial intelligence and robotics [14], to teach object-oriented design and programming [15] as well as to promote enrichment of student cognition processes [16].

Our contribution is 1) development of the e-learning environment, which includes constructing architecture of the collaborating robots with communication channels and control software, 2) adaptation of the operating

978-1-4799-0009-1/13/$31.00 ©2013 IEEE

environment for teaching and learning purposes, and 3) demonstrating the use of task decomposition and allocation principles in teaching algorithms and programming at the high school.

The remaining parts of the paper discuss the concept of robots as collaborative learning objects and their application following the collaborative learning model (Section II), present a framework of the e-learning environment (Section III), describe its application for teaching topics of CS curricula (Section IV), provide an example of collaborating robots used as learning objects (Section V), and present conclusions (Section VI).

II. ROBOTS AS COLLABORATIVE LEARNING OBJECTS

Collaborative learning (CL) is a pedagogical approach wherein students work in small groups together towards a common learning goal with the help of computers [17]. Learning is driven by social interaction as a tool to share and construct knowledge among learners while technology serves for communication or as a common resource [18]. CL represents a shift away from the traditional teacher-centred learning, where a teacher is the principal source of knowledge and skills, to student-centred learning, where students actively solve problems and create knowledge by working together to accomplish shared learning goals [19]. Commonly, CL is achieved using virtual learning environments which provide virtual learning objects (LOs) (as digital or web-based chunks of knowledge + metadata) and tools to engage students in the learning process [20].

Educational robotics provides a technological background and opportunities to extend the ideas of CL and constructionism beyond virtual LOs to robots as *tangible LOs* [21]. Robots on their own are tremendously engaging for students, allow to achieve a wide range of learning goals in CS teaching, and provide a good foundation for learning computing [22]. However, the concept of *collaborating* or *social* robots [23] allows expanding opportunities for active project-based learning in educational robotics even more. Here we understand 'social robots' as two or more physical entities which interact with each other within the same context [24]. With social robots we may obtain an immense degree of engagement and involvement of students, and therefore greater knowledge learning and retention.

In [21], we have introduced the concept of Robots as Physical Learning Objects (PLO), which extends the notion of a traditional LO beyond the virtual domain (e-content, web page) to a physical domain (robot hardware and physical processes that are demonstrated by the hardware). A PLO is a smart thing (e.g., a mobile robot) that has sensors and/or actuators to interact with its environment and content (control program) to control its behaviour. If multiple robots are used together in the educational context to achieve the same goals while exchanging messages between themselves, the PLOs can become the CLOs, the Collaborative Learning Objects. A CLO is a LO that interacts with other LOs to facilitate the transfer of knowledge and the achievement of learning goals of the learners. We claim that students who, while working in a group, design, program (from scratch or by modifying provided examples), deploy robotic CLOs, and provide research of their behaviour, can achieve the promises of collaborative learning and constructionism (i.e., proactive de-construction and re-construction of knowledge) even more effectively than using virtual LOs only.

When designing instruction for robotic CLOs, we adopt and extend the collaborative learning model from [25]:

1) *Engagement:* learning should provide collaborative tasks that supports and ensures group learning activities.

2) *Exploration:* learner groups work on the critical analysis of provided concepts and information (i.e., de-construction of knowledge).

3) *Transformation:* learner groups modify given algorithms to achieve given tasks while at the same time they synthesise learning concepts and (re-)construct knowledge.

4) *Research:* learner groups test and evaluate implemented algorithms by directly observing the behaviour of robots and record the results.

5) *Presentation:* learner groups prepare presentations of their work and receive feedback from the teacher.

6) *Reflection:* learners analyse what they have learned and offer constructive ideas on how learning can be improved and provide feedback to the teacher.

Next, a framework of a collaborative robot environment based on the ideas elaborated in this Section is presented.

III. FRAMEWORK OF COLLABORATIVE ROBOT ENVIRONMENT

The architecture of the collaborative robot-based learning environment (as compared to the one that uses a single robot, e.g., in [26]) is based on a classical master-slave model and includes additional components required for robot orientation in its environment (sensors, wireless cameras), communication channels to ensure the exchange of messages between communicating robots and support for different communication protocols (Bluetooth, WiFi), and control hardware/software (PC). In the master-slave model, the execution of a parallel process can be seen as a sequence of parallel and sequential sub-processes, where slaves perform parallel computations and the master does sequential computations. Sub-processes are controlled using communication between the master and slaves either by single node broadcast from the master or by send/receive messages exchanged between the master and any slave [27]. The principle is similar to task decomposition so the master-slave model itself can be used as an illustrative example of practical implementation of task decomposition.

Fig. 1 presents a four-tiered framework to construct the collaborative robots-based environment as follows:

1) *Deliberative layer:* Central Coordinator (CC) receives initial tasks for robots from the teacher, then decomposes tasks into sub-tasks and uploads generated robot control programs (RCP) to the student PCs. In the simplest case, each task is divided into two sub-tasks (Master → Slave) and also we have two independent groups of students (GROUP1, GROUP2) assigned to work with the same task.

2) *Physical layer:* tangible mobile robots with wheels driven by servomotors.

3) *Reactive layer:* sensors allow a robot to receive information about its environment and react to its changes.

4) *Communication layer:* exchange of messages between robots and provision of feedback to teacher's PC for monitoring and evaluation.

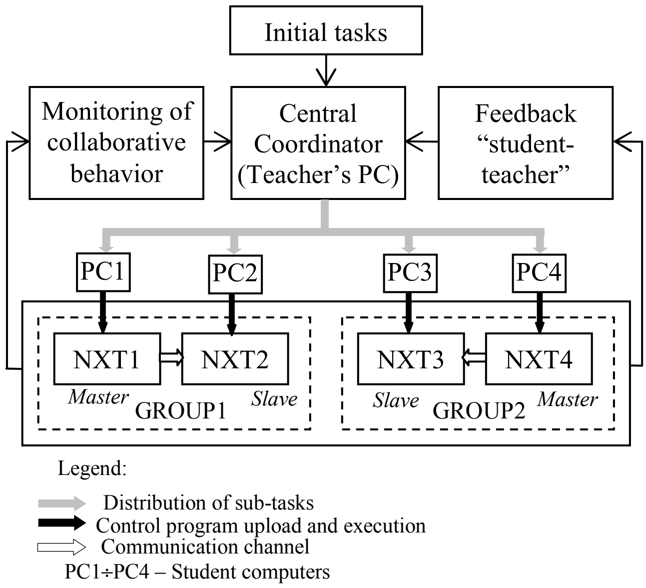


Figure 1. Framework of collaborative robots based environment for e-learning

In real setting, the number of collaborating robot groups depends on the technical capabilities (the number of available robots and PCs in the classroom) and educational needs (the number of students, teaching and learning objectives, etc.). In order to ensure satisfaction of educational needs and improvement of technical reliability, we provide a real-time "student-teacher" feedback and monitoring of collaborative behaviour of robots.

IV. USING E-LEARNING ENVIRONMENT TO TEACH CS TOPICS

The section presents an elaboration of the e-learning environment for teaching the CS course at high school.

Hardware. To implement the environment (see Fig. 1), we use heterogeneous (meaning robots with different sensors) LEGO MINDSTORMS NXT robots, which are named according to the tasks to be performed by them:

1. LINE FOLLOWER (Fig. 2, right) is a master robot that uses two light sensors to follow a black line on the floor, and the ultrasonic sensor to observe the distance to any obstacles, and sends messages to the DRAWBOT.
2. DRAWBOT (Fig. 2, left) is a slave robot that receives messages from LINE FOLLOWER and draws lines on the floor. It has been adopted from [26] to build this new environment. This robot is used to demonstrate visually the solution of the CS tasks.

The teacher's computer is used for formulation of tasks and distribution of sub-tasks to student computers. The teacher's computer also uses the mydlink-enabled Wireless N Network Camera DCS 932-L to monitor the collaborative behavior of robots and to receive a qualitative "student-teacher" feedback when students perform tasks.

Software. The software part of the system includes task-decomposition and allocation software in the teacher's PC and robot control programs in the student PCs.

Robot control programs are implemented in RobotC and uploaded from student PCs to Lego NXT robots.

For DRAWBOT, the students can select from a library of pre-programmed ornament templates and select values of template parameters to generate the ornament drawing program, while the robot executes the program and draws a figure on the floor using a pencil held by its gripper.

For LINE FOLLOWER robot, the students can select from a set of line following algorithms (One Inside, One Bounce, Straddle, Two Inside [28]), try and observe the behaviour of a robot, then modify robot control programs as needed and add functions for communication between LINE FOLLOWER and DRAWBOT. The One Inside and One Bounce algorithms use one light sensor in front of the robot that detects the edge of the line. The Straddle and Two Inside algorithms use two light sensors at the robot's front. In case of Straddle algorithm, sensors are positioned on the either side of the line, and in case of Two Inside algorithm, sensors are positioned side-by-side inside the line.

Communication. The NXT supports the Bluetooth application protocol called Serial-Port Profile (SPP). It is implemented on the top of a low-level RFCOMM (Radio Frequency Communication Protocol) protocol, which provides a simple reliable data stream to the user. Bluetooth in the NXT is used to ensure communication 1) between PCs and LEGO robots (up to three, but only with one at a time), 2) between teacher and students' PCs, and 3) between other Bluetooth-enabled devices (e.g., mobile phones, tablet PC). When multiple devices are connected together, a master/slave relationship is established. The master always creates the Bluetooth connection and initiates communication. To ensure reliability of Bluetooth connection we follow the rules formulated in [29].

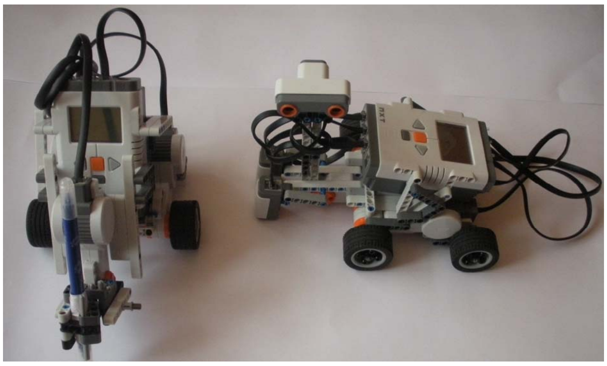


Figure 2. A full view of DRAWBOT and LINE FOLLOWER

The architecture of the implemented system is shown in Fig. 3.

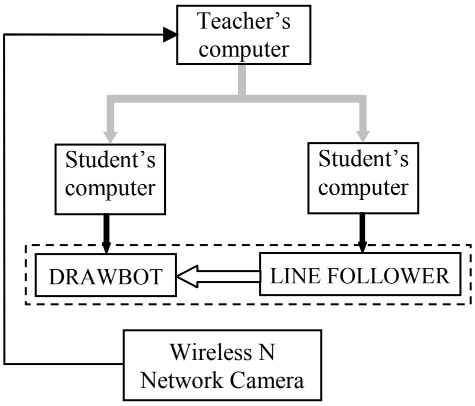


Figure 3. Example of environment with two collaborating robots

EXAMPLE OF COLLABORATIVE LEARNING OBJECTS

Using DRAWBOT and LINE FOLLOWER robots and principles described above, we have created a collection of robot-based Collaborative Learning Objects (CLOs) for teaching and learning of the CS course topics: "Loops", "Nested loops", "Conditional branching statements", "Task decomposition", and "Subtask aggregation".

We illustrate the use of the environment to implement the tasks described below using the collaborative learning model proposed by Reid *et al.* [25] by as follows.

1) *Engagement.* The students are introduced with the idea of complex behaviour as a problem that requires decomposition, allocation and coordination of tasks. A simple example of a robot that has to follow a line and avoid obstacles at the same time is given. The principle of task decomposition as a method to solve complex problems is explained. The conditions and constraints of the exemplary complex behaviour are explained such as static vs. dynamic obstacles, priority of tasks, etc.

2) *Elaboration.* The students gather in groups and elaborate on information given by the teacher. The sub-problems are identified and required sub-tasks are formulated for master and slave robots: line following, obstacle detection and message sending for master robot, roaming, message receiving and stopping for serve robot.

3) *Transformation.* The students analyse the library of small robot control programs/functions and adapt suitable functions for implementation of sub-tasks. Obstacle searching function is adopted for obstacle avoidance sub-task and ornament drawing function is adopted for roaming sub-task. Also the students select and adapt the variant of the line following algorithm and perform research of the behaviour of robots.

4) *Research*.

Fig. 4 presents a view on a research environment. DRAWBOT (in the centre) draws a selected ornamental figure bounded by a black line, while LINE FOLLOWER follows the black line and at the same time observes the distance to DRAWBOT. If the distance between robots becomes too big or too small, LINE FOLLOWER sends a message to DRAWBOT and to stop. Two different routes are used (see Fig. 5): the elliptical (its radii are 21 and 32 cm) and rectangular (lengths of sides are 42 and 70 cm).

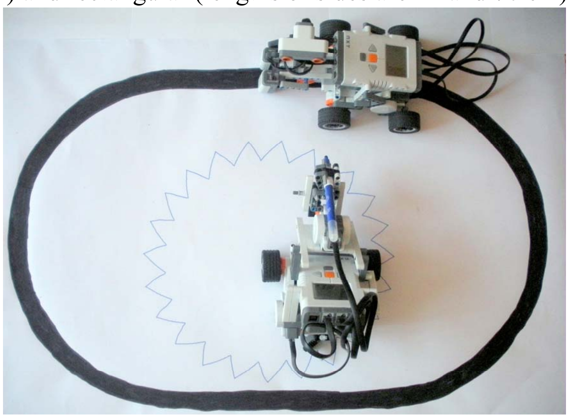


Figure 4. Collaborating robots: DRAWBOT (centre) and LINE FOLLOWER (above)

A dotted line presents the real path of LINE FOLLOWER obtained experimentally. When the robot's speed is 10 to 30 % of full power, the robot's trajectory coincides with the black line. When the robot's speed is larger than 40 % of full power, it is unable to follow the black line exactly and consequently deviates from it.

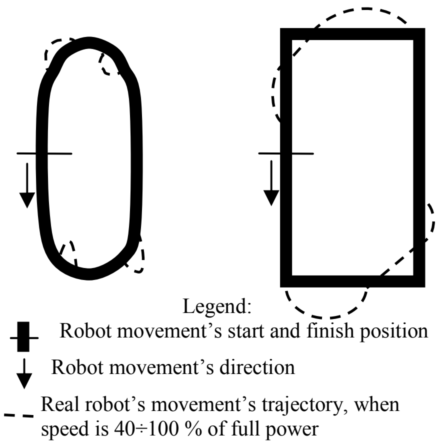
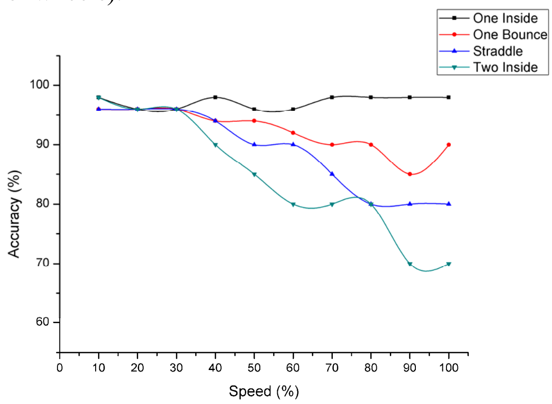


Figure 5. Elliptical and rectangular routes followed by LINE FOLLOWER

5) *Presentation.* Students present the results of their

team work and research to the teacher and receive feedback. As an example Fig. 6 presents the results of accuracy comparison of line following algorithms. Accuracy is calculated by estimating what part of path robot overcomes without leaving the black line while following the routes of different shape and driving at different speed (speed is expressed as a percentage of max power level of servomotors controlling the rotation of robot wheels).



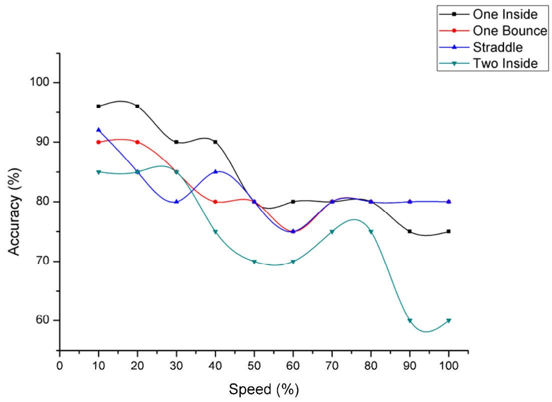


Figure 6. Results of research of line following algorithm using elliptical (top) and rectangular (bottom) routes

6) *Reflection.* Students fill surveys provided by the teacher and evaluate advantages and disadvantages of the course topic and its pedagogical delivery methods.

V. CONCLUSIONS

The e-learning environment based on collaborating robots supports constructionism-based teaching and learning paradigm (using project and problem-solving based methods) in a Computer Science (CS) course at high school.

The concept of robotic Collaborative Learning Objects (CLOs) allows us to step beyond stand-alone robots as educational tools, to extend the range and to raise the complexity of CS topics delivered to high school students thus increasing engagement and involvement of students in advanced group-based projects where using a single robot may be a too simple task for teamwork.

The proposed e-learning environment using robots for collaborative learning 1) provides the interdisciplinary aspects of teaching (the tasks considered are related to mechanics, physics, mathematics, and computer science); 2) increases student engagement in learning; 3) develops student abilities to critically analyse and compare different problem solving algorithms (e.g., line-following algorithms in our example); and 4) introduces to foundations of research and result presentation. As a result our approach, as compared to collaborative learning without using robots [17, 18], has much more capabilities for advanced e-learning.

The e-learning environment introduces also two basic challenges: 1) the need of the flexible communication infrastructure for groups of mobile robots; 2) the complexity of specifying collaborative behaviour.

Future work will focus on the extension of the architecture of the e-learning environment with multi-master / multi-slave model to allow using a larger number of communicating robots for learning at the same time and thus allowing to deliver teaching of more complex CS topics.

ACKNOWLEDGEMENT

The work has been carried out partially within the framework the Operational Programme for the Development of Human Resources 2007-2013 of Lithuania „Strengthening of capacities of researchers and scientists" project VP1-3.1-ŠMM-08-K-01-018 „Research and development of Internet technologies and their infrastructure for smart environments of things and services" (2012- 2015), funded by the European Social Fund (ESF).

REFERENCES

[1] F.B.V. Benitti, “Exploring the educational potential of robotics in schools: A systematic review”, *Computers & Education,* 58(3), 978–988, 2012.

[2] C. Dede, “Theoretical perspectives influencing the use of information technology in teaching and learning”, in J. Voogt, G. Knezek (eds.) *International Handbook of Information Technology in Primary and Secondary Education,* Springer Science + Business Media, 43–62, 2008.

[3] M. Resnick, F. Martin, R. Sargent, B. Silverman, "Programmable Bricks: Toys to Think With", *IBM Systems Journal,* 35(3-4), 443–452, 1996.

[4] A.-M. Eubanks, R.G. Strader, D.L. Dunn, “A comparison of compact robotics platforms for model teaching”, *Journal of Computing Sciences in Colleges*, 26(4), 35–40, 2011.

[5] N. Brittain, “IT education remains mired in uncertainty”, *Computing*, 20 Sep 2011. [Online]. Available: http://www.computing.co.uk/ ctg/analysis/2110259/education-remains mired-uncertainty

[6] M. Mataric, “Robotics Education for All Ages”, *Proc. of AAAI Spring Symposium on Accessible, Hands-on AI and Robotics Education,* Palo Alto, CA, 2004.

[7] S. Papert, “Situating constructionism”, in I. Harel, & S. Papert (Eds.), *Constructionism.* Norwood, NJ7 Ablex Publishing, 1991.

[8] D. DeLuca, “Robotics and Teaching: Promoting the Effective Use of Technology in Education”, *Honors thesis,* Tufts University, 2003.

[9] Y. Kafai, M. Resnick (Eds.), *Constructionism in Practice,* Lawrence Erlbaum Associates Publishers, Mahwah, New Jersey, 1996.

[10] M.U. Bers, I. Ponte, K. Juelich, A. Viera, J. Schenker, "Teachers as Designers: Integrating Robotics in Early Childhood Education", *Information Technology in Childhood Education*, 123-145, 2002.

[11] M. Berland, T. Martin, T. Benton, „Programming Standing Up: Embodied Computing with Constructionist Robotics", *Proceedings of Constructionism*, Paris, 2010.

[12] J.A. Mèndez-Polanco, A. Munõz-Meléndez, "Collaborative Robots for Indoor Environment Exploration", *Proc. of 10th Intl. Conf. on Control, Automation, Robotics and Vision*, Hanoi, Vietnam, 359–364, 2008.

[13] N. Roy, G. Dudek, "Collaborative Robot Exploration and Rendezvous: Algorithms, Performance Bounds and Observations", *Autonomous Robots,* 11(2), 117-136, 2001.

[14] A. Witsch, "A ROS-based framework for collaborative robot team", Distributed Systems Group, Kassel University. [Online]. Available: http://www.betriebssysteme.org/Aktivitaeten/Treffen/2012-Koblenz/Programm/docs/slides_witsch.pdf

[15] T.P.J. Tse, S.C.F. Chan, G. Ngai, C.Y. Cheung, V.T.Y. Ng, "Dynamic Collaborative Robotic Platform – A Brief Introduction", *Proc. of the 2009 13th Int. Conference on Computer Supported Cooperative Work in Design*, 125–130.

[16] R. Mitnik, M. Nussbaum, M. Recabarren, „Developing Cognition with Collaborative Robotic Activities", *Educational Technology & Society*, 12(4), 317–330, 2009.

[17] A.A. Gokhale, "Collaborative learning enhances critical thinking". *Journal of Technology Education*, 7(1), 22−30, 1995.

[18] G. Stahl, T. Koschmann, D. Suthers, "Computer-supported collaborative learning: An historical perspective", In R.K. Sawyer (Ed.), *Cambridge handbook of the learning sciences* (pp. 409-426). Cambridge, UK: Cambridge University Press, 2006.

[19] M. Lakkala, "The Pedagogical Design of Technology Enhanced Collaborative Learning", *E-Learning Europe,* 2007. [Online] Available: http://www.elearningeuropa.info/files/media/3028.pdf.

[20] J. Fiaidhi, "Virtual SceneBean: a Learning Object Model for Collaborative Virtual Learning Environment", *Informatics in Education,* 3(2), 191–218, 2004.

[21] R. Burbaitė, R. Damaševičius, V. Štuikys, "Using Robots as Learning Objects for Teaching Computer Science", Accepted to *WCCE 2013,* Torun, Poland, July 2-5, 2013.

[22] R. Goldman, A. Eguchi, E. Sklar, "Using educational robotics to engage inner-city students with technology", *Proc. of the 6th Int. Conference on Learning Sciences* (ICLS '04). International Society of the Learning Sciences, 214-221, 2004.

[23] F. Hegel, C. Muhl, B. Wrede, M. Hielscher-Fastabend, G. Sagerer, "Understanding Social Robots," *Proc. of 2nd Int. Conf. on Advances in Computer-Human Interactions, ACHI '09,* 169–174, 2009.

[24] B. Duffy, C.F.B. Rooney, G.M.P. O'Hare, R.P.S. O'Donoghue, "What is a Social Robot?", *10th Irish Conference on Artificial Intelligence & Cognitive Science,* University College Cork, Ireland, 1999.

[25] J. Reid, P. Forrestal, J. Cook, *Small Group Learning in the Classroom.* Portsmouth, NH: Heineman, 1989.

[26] R. Burbaite, V. Stuikys, R. Marcinkevicius, "The LEGO NXT Robot-Based e-Learning Environment to Teach Computer Science Topics", *Electronics and Electrical Engineering*, 18 (2), 133–136, 2012.

[27] Y. Yalçinkaya, T. Steihaug, "An Analytical Model for a Class of Architectures under Master-Slave Paradigm", *Proc. of 8th Int. Conf. on High-Performance Computing and Networking,* Amsterdam, The Netherlands, Springer LNCS vol. 1823, 601–604, 2000.

[28] J.A. Gray. Toeing the Line: Experiments with Line-following Algorithms. [Online]. Available: http://www.fll-freak.com/misc/01-jgray_report.pdf

[29] S. Toledo, "Analysis of the NXT Bluetooth-Communication Protocol". Technical paper. [Online]. Available: http://www.tau.ac.il/~stoledo/lego/ btperformance.html