The use of a maze solver robot to support the teaching and learning process

João M. Vilas Boas[1], Allyson A. A. F. Soares[1], Ivanilson F. Vieira Júnior[1], Érica L. G. T. Gomes[1], Isaac B. L. da Silva[1], Samara R. da Silva[1], Justo E. Alvarez Jácobo[2], Luiz M. G. Gonçalves[2]

[1]Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte (IFRN)
CEP 5915-000 – Natal – RN – Brasil

[2]Universidade Federal do Rio Grande do Norte (UFRN) – Natal – RN - Brasil.

{joao.vilasboas,allyson.soares,ivanilson.junior,erica.torres}@ifrn.edu.br,
justo.alvarez@gmail.com,lmarcos@dca.ufrn.br

Abstract. *This article reveals the fundamental elements, the developments, and the results of a methodology based on the use of a maze solver robot for supporting teaching and learning in logical and programming disciplines, as an alternative proposal for solving learning difficulties experimented by students in programming and algorithms disciplines carried out at the Instituto Federal de Educação do Rio Grande do Norte.*

1. Introduction

Technology is becoming more and more present in daily activities of humans, in despite of the several conjunctures of society. In this scenario, robotics is one of the most promising areas of this technological progress, based on the fact that it has achieved a remarkable place on the technological innovations, with a major focusing on industry applications, initially. However, without getting apart our regular activities, robotics has becoming present in home appliances, electronics devices, lifts, and another lot of daily used systems, all of them with a "robot side" [Zilli 2004]. Nonetheless, robotics applications can also be found in environments and situations that are dangerous to human health, manufacturing processes to increase efficiency, even in organizations that accomplish robotic competitions with the focus of encouraging interest on areas like computer vision and artificial intelligence [Gomes *et al.* 2001], [Vilas Boas *et al.* 2014].

Puntel *et al.* (2013) corroborates with this postulate saying that this vision of "industrial robot" has been changed along the years, becoming more usual to interact with robots in our daily simple activities, for example, vacuum cleaner robot. Although, the word "simplicity" looks like hard to associate when it is time to try to understand all of the theory connected with construction and robots operation. It should not be ignored the fact that the system complexity is intrinsically linked to the objective for which the robot is being projected [Guimarães 2007].

Therefore, it is supposed that these new technologies can be used as valuable resources to help the teaching and learning process at schools, being it elementary, high or even college. Because of this panorama, mainly in the last decade, educational robotics area has succeeded a significant expansion in academic environments [Barbero & Demo 2011, Perez *et al.* 2013]. This justifies, in fact, the possible and promising introduction of these new technologies, even in a specific situation as the one described next. In fact, learning difficulties are usually identified in Brazilian educational institutions in disciplines inherent to programming and algorithms contents. In the *Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte* (IFRN), Center Natal campus, the reality of obstacles for the apprenticeship in logics and algorithms (basilar curricular component of computer science related courses) has a very expressive diagnosis. This situation is verified by way of a simple analysis on the student's grades in the Fundamentals of Logic and Algorithms discipline, in the last four years of a high school technical course in the career of Informatics for Internet.

According to the analyzed data, the percentage of students that fail in Logic and Algorithms discipline has an average of 24% in the last four years. This fact shows that almost a quarter of students are not developing the learning satisfactorily, on programming. It entails not only the possible psychological conflicts of a failed student, but also in the reality of permanence/quality in the curse, considering that the subsequent subjects use programming as a prerequisite/basilar content.

In order to overcome these difficulties, robotics appears as a promisor element in the learning process, with a proposal to changing the pedagogical praxis in the school, with the perspective of new teaching and learning model.

In this direction, this paper proposes to use educational robotics as a complementary tool for the improvement of the students learning in logic and programming. In this proposal, the basic activity is based on the construction of a maze arena that makes able to afford some possibilities of solutions by maze solver robots. Thus, through a simulated environment, the students have the possibility to apply and learn the theory of programming and logic that has been taught in their classroom.

Withal seeing the described problem, offering a project like this promotes a different pedagogical action for the teacher, applying the principle of action-reflection-action and linking with the theory and practice in a dialogical, contextualized, interdisciplinary and flexible way.

2. Maze Solver project description

The Maze Solver project consists in proposing to the students the challenge of building and optimizing an own algorithm to solve a maze of lines using the smallest number of decisions, and, consequently, the least possible time.

2.1. Robotics Platform

This project uses the Polulu 3PI robotic platform, a small robot with a circular shape with 9.5 cm diameter as seen in Fig. 1. The control of the robot is based on the Atmega 328P-PU microcontroller, which can be programmed through the AVR studio. Robot movements are allowed through two independents wheels that are activated by two CC motors, and by one omnidirectional supporting wheel. Because of its size, shape, and high accuracy, this robot has proven to be the ideal tool to solve the maze of lines, as proposed in this project. To accomplish this objective, the Polulu 3PI has a line following function, which is done through the use of five optical-reflective sensors that are put on the lower front part of the robot. Fig. 1 introduces the main electromechanical features of the used robotic device.

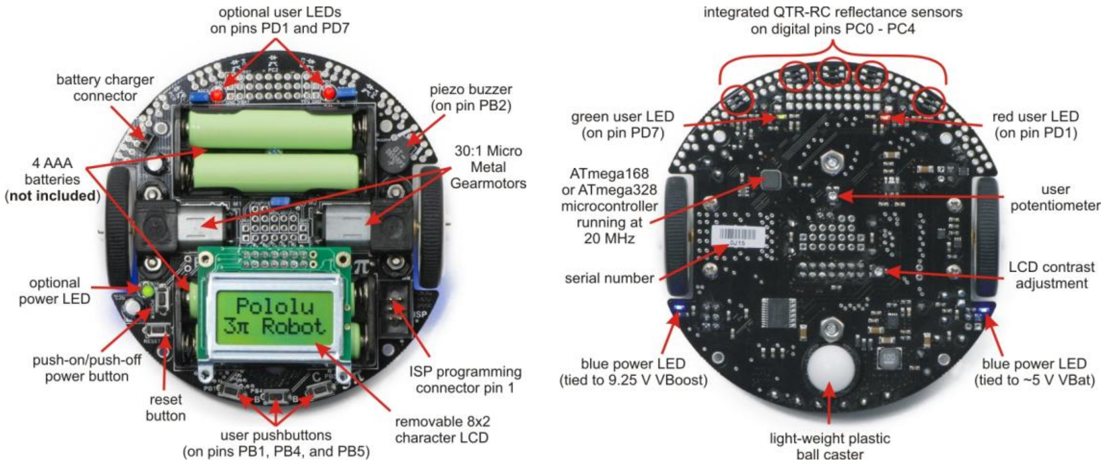

Figure 1. General features of Pololu 3pi Robot [Pololu 2014].

2.2 Arena tests

A robotic arena is built to simulate a real environment controlled and safe in order to make it easy the execution and repetition of tests with small robots. The arena is built with MDF and has 90 x 120 cm of free area. The maze of lines are delimited inside it following different solutions, depending on the logical programming utilized in the robot. With the intention of obtaining a better analysis of the student's algorithms progress, all of the thirty-seven points of decision-making of the arena are mapped. Thereby, parameters, as the number of decisions and the resolution time of the maze, are embraced to measure the efficiency of the implemented algorithms. Fig. 2 presents the arena being used by the robot during an attempt to finishing the maze.

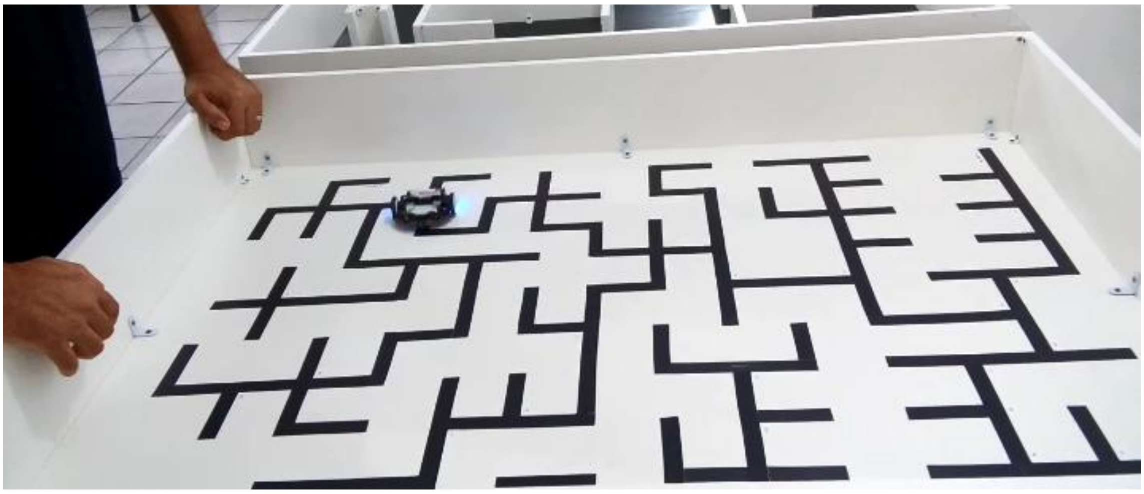

Figure 2. Robot and arena used for experiments.

2.3 Maze solver algorithms

According to environment knowledge, the methods/algorithms for a maze solution can be split in two groups: i) without previous knowledge or mapless navigation*:* random mouse, wall follower, to the left or to the right [Rosa 2015], Pledge [Paperd 1973] and Trémaux [Sánchez 2010]; and ii) with the knowledge of the environment or map-based navigation: shortest path, dead-end filling (in French, *Cul-de-sac*), etc. The algorithms with a prior information has the best results [Cai *et al.*

2010]. Other important algorithms mentioned for Rosa (2015) are: the Dijkstra, the Flood Fill and a variation of the last one.

2.3.1 Implemented algorithm

Originally, the robot is programmed using the Atmel Studio software, an almost pure form of C++ language, with a few libraries and uncomplicated functions that execute the necessary tasks, what increases the student knowledge. In order to facilitate the programming, we decide to adapt the IDE used with Arduino. So, it is possible to enjoy the robot platform robustness, added to the simplicity of a friendly software. Fig 3 presents an excerpt of the developed IDE Arduino algorithm.

```
void loop() {
  while (1) {
    follow_segment();
    OrangutanMotors::setSpeeds(50, 50);
    delay(50);
    unsigned char found_left = 0;
    unsigned char found_straight = 0;
    unsigned char found_right = 0;
    unsigned int sensors[5];
    robot.readLine(sensors, IR_EMITTERS_ON);
    if (sensors[0] > 100)
      found_left = 1;
    if (sensors[4] > 100)
      found_right = 1;
    OrangutanMotors::setSpeeds(40, 40);
    delay(200);
    robot.readLine(sensors, IR_EMITTERS_ON);
    if (sensors[1] > 200 || sensors[2] > 200 || sensors[3] > 200)
      found_straight = 1;
    if (sensors[1] > 600 && sensors[2] > 600 && sensors[3] > 600)
      break;
    unsigned char dir = select_turn(found_left, found_straight, found_right);
    turn(dir);
    path[path_length] = dir;
    path_length++;
    simplify_path();
  }
  OrangutanMotors::setSpeeds(0,0);
  if (millis() % 2000 < 1000) {
    OrangutanLCD::clear();
    OrangutanLCD::print("Solved!");
    OrangutanLCD::gotoXY(0, 1);
    OrangutanLCD::print("Press B");
  } while (1) {
      if (currentIdx < MELODY_LENGTH && !buzzer.isPlaying()) {
        buzzer.playNote(note[currentIdx], duration[currentIdx], 15);
        currentIdx++;
      } if (OrangutanPushbuttons::isPressed(TOP_BUTTON)) {
          buzzer.stopPlaying();
          if (currentIdx < MELODY_LENGTH)
            currentIdx = MELODY_LENGTH;
          else
            currentIdx = 0;
          OrangutanPushbuttons::waitForRelease(TOP_BUTTON);
          break;
      }
    }
  delay(30);
}
```

Figure 3. Implemented algorithm.

In our case study, the algorithm called "left hand rule", based in the left wall follower, is used in order to conclude the route. Basically, the rule predicts when the robot finds an intersection to the left of the robot, then it must turn 90° towards the line and follow it, independently if exist lines in front or right lines. If there's not a left line, the robot must choose to go straight ahead, turn right or even turn around (180°), in this sequence. Under these circumstances, the end of the maze is eventually found, which is delimited by a black square.

In this scenario, the robot should use a logic proposal, as a first attempt, using the above algorithm and then execute the smallest track possible to the end of the maze, considering only optimized route decisions. Fig 4 presents a scheme of the paths traced by the robot with the algorithm used.

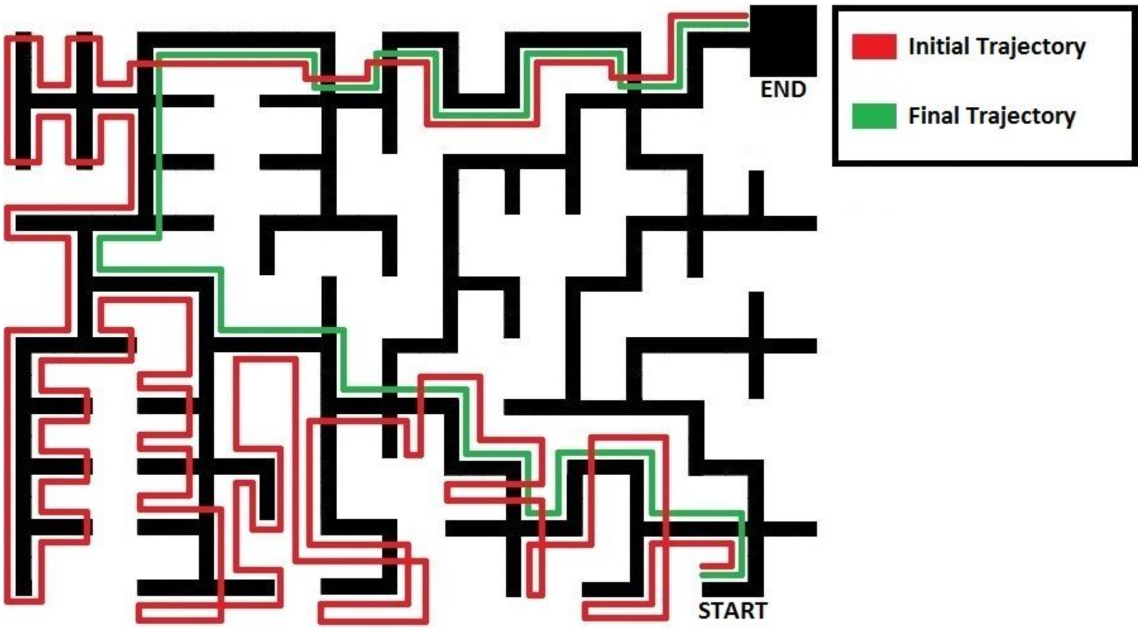

Figure 4. Labyrinth solution during the test.

4. Conclusion

Beyond the results obtained in this study case described in this paper, it is possible to observe that the use of this proposed scenario composed by a maze arena and the robot platform is really promisor regarding to the improvement of interest and learning in logic and programming of the students.

In addition, this project has being used as a didactic/pedagogic mechanism that encourages the integral formation of students, through the scientific, critical construction and search of knowledge.

Besides, the researcher attitudes will always be aroused to students, having the research as a knowledge source and the transformation of the academicals environment in a dynamic space. Besides being an educative, reflexive and formative act, research is a politician act, thus “should be a usual attitude of teacher and student" (Demo 2002).

Therefore, it is still necessary to develop and to stimulate researches that able to discuss the issues raised in the academic context and the particularity of teaching practices, creating opportunities for the evolved subjects, both critically and scientifically, that allow them to make questions and interpretations making possible their own growing.

5. References

Barbero, A.; Demo, G. B. (2011) The art of programming in a Technical Institute after the Italian secondary school reform. Proceedings ISSEP, Bratislava.

Cai, J. et al. (2010) “An algorithm of micromouse maze solving”. Computer and Information Technology (CIT), 2010 IEEE 10th International Conference on, 2010. 1995-2000.

Demo, P. (2012) Pesquisa: Princípio científico e educativo. 9ª ed. São Paulo, Cortez.

Gomes, M. M. *et al.* (2001) “Aplicação do sistema de futebol de robôs como ferramenta didática”, In: XXIX COBENGE, Porto Alegre/RS.

Guimarães, F. A. (2007) Desenvolvimento de robô móvel utilizado para a exploração de ambientes hostis. Dissertação (Mestrado em Engenharia de Processos Químicos e Bioquímicos) - Escola de Engenharia Mauá, São Caetano do Sul/SP.

Papert S. (1973) "Uses of Technology to Enhance Education", MIT Artificial Intelligence Memo No. 298, June 1973

Perez, A. L. F. *et al.* (2013) Uso da Plataforma Arduino para o Ensino e o Aprendizado de Robótica. In: International Conference on Interactive Computer aided Blended Learning. Florianópolis/SC.

Pololu (2014) Pololu 3pi Robot User’s Guide [Online], Disponível em: https://www.pololu.com/docs/0J21, Acesso em: 23/06/16.

Puntel, F. E. *et al.* (2013) “Desenvolvimento de um Robô Explorador para Ambientes Indoor”, In: 2º Simpósio de Integração Científica Catarinense, Araranguá/SC.

Rosa, S. A. et al. (2015) “Um algoritmo para solução de labirintos com uso de robô móvel”. In: XII Simposio Brasileiro de Automação Inteligente (SBAI). Natal/RN.

Sánchez, C. H. (2010) “Los diez mejores momentos matemáticos de Los Simpson”. Números: Revista de Didáctica de las Matemáticas, vol. 73, março de 2010, Pg. 35-40 ISNN: 1887-1984

Vilas Boas, J. M. *et al.* (2014) “Dispositivo GPRS para detecção de gases nocivos em ambientais industriais”, In: IX CONNEPI, São Luís/MA.

Zilli, S. R. (2004) A robótica no ensino fundamental: Perspectivas e Práticas. Dissertação (Mestrado em Engenharia de Produção) - UFSC, Florianópolis/SC.