

<https://eventos.utfpr.edu.br/sei/sei2018>

## Inclusão digital de jovens pelo estudo da robótica

### Digital inclusion of young students by robotics

Álefe Felipe Gonçalves Pereira  
Dias

[alefe@alunos.utfpr.edu.br](mailto:alefe@alunos.utfpr.edu.br)  
Universidade Tecnológica Federal  
do Paraná, Curitiba, Paraná, Brasil

João Alberto Fabro

[fabro@utfpr.edu.br](mailto:fabro@utfpr.edu.br)  
Universidade Tecnológica Federal  
do Paraná, Curitiba, Paraná, Brasil

Lucas Eduardo Bonancio Skora

[skora.lucas@gmail.com](mailto:skora.lucas@gmail.com)  
Universidade Tecnológica Federal  
do Paraná, Curitiba, Paraná, Brasil

Thales Romelli Rodrigues Keps

[thaleskeps@gmail.com](mailto:thaleskeps@gmail.com)  
Universidade Tecnológica Federal  
do Paraná, Curitiba, Paraná, Brasil

#### RESUMO

**OBJETIVO:** Capacitar estudantes de ensino fundamental e médio, a compreender os conceitos básicos de programação, através de interação direta com dispositivos robóticos programáveis, físicos e simulados. **MÉTODOS:** Interfacear o resultado da programação através de robôs reais e simulados, aplicando esse modelo em alunos do ensino fundamental e médio, e também em professores. Mostrar as diferenças entre robôs físicos e simulados, incentivar a participação em olimpíadas de robótica. **RESULTADOS:** Metodologia se mostrou adequada, despertando interesse de alunos e professores para a área da robótica. Algumas escolas municipais de Curitiba pretendem utilizá-la em sala de aula. **CONCLUSÕES:** O fato de professores e alunos usarem as ferramentas desenvolvidas abre um leque para o ensino de robótica nas escolas, aumentando o interesse na área tecnológica como um todo.

**PALAVRAS-CHAVE:** Robótica. Educação. Fluxograma. Simulação

#### ABSTRACT

**OBJECTIVE:** To capacitate young students (from 6 to 17 years) to understand the basics concepts of programming, an approach is proposed, that uses direct interaction with programmable robotics devices (both physical and simulated robots). **METHODS:** To evaluate the result of programming through real and simulated robots, applying this model to allow young students to directly manipulate/program both real and simulated robots. One of the objectives is to highlight the differences between physical and simulated robots, and to incentive the participation in robotics olympics. **RESULTS:** The methodology showed adequate, stimulating students and teachers interest in the area of robotics. Some municipal schools in Curitiba intend to use it in the classroom. **CONCLUSIONS:** The fact that teachers and students use the developed tools open a range for teaching robotics in schools, increasing an interest in the technological area as a whole.

**KEYWORDS:** Robotic. Education. Fluxogram. Simulation

**Recebido:** 31 ago. 2018.

**Aprovado:** 29 set. 2018.

#### Direito autoral:

Este trabalho está licenciado sob os termos da Licença Creative Commons-Atribuição 4.0 Internacional.



## INTRODUÇÃO

A robótica educacional tem ganhado cada vez mais espaço no cenário da educação brasileira, sendo aplicado desde os primeiros anos do ensino fundamental até alguns cursos de graduação. Várias pesquisas [1][2] atestam que essas iniciativas possuem enorme aceitação por parte dos alunos, além de serem aplicadas com sucesso para incentivar o estudo e compreensão de diversas disciplinas, como a física, geografia, matemática, etc. Essas estratégias são especialmente eficazes pois permitem que o aluno visualize o resultado do que fez, além de ajudar a integrar as diferentes áreas de conhecimento de um curso (principalmente na engenharia), e despertar o interesse dos estudantes, melhorando seu entendimento do porquê ele deve estudar e compreender os conceitos de matemática, física, entre outras disciplinas, para ser capaz de então programar os robôs para realizar as tarefas pretendidas.

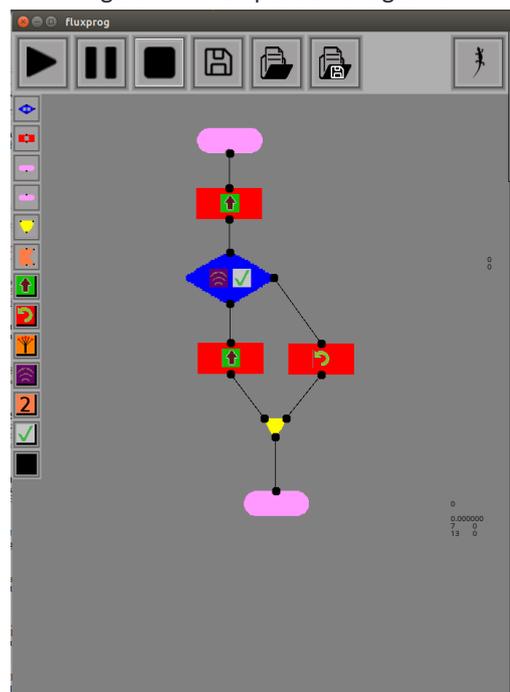
Entretanto, muitas dessas propostas envolvem alunos com experiência de programação [3] e/ou utilizam o kit de desenvolvimento da empresa Lego [4], que possui um custo relativamente alto (em 2018, por volta de 3 mil reais). Por isso se optou por desenvolver um kit de desenvolvimento de baixo custo, usando componentes eletrônicos facilmente encontrados em lojas de eletrônica, e usando como programação fluxogramas, já que se considera esta a mais simples maneira de introduzir alunos à programação [5].

Para testar a programação, é usado tanto um ambiente simulado, quanto um real. O ambiente de simulação é muito importante no estudo de robôs, já que permite testar a programação dos mesmos sem ter desgaste ou quebra do hardware do robô. Para este projeto, foi escolhido o simulador robótico V-REP (*Virtual Robotics Experimentation Platform*) [6], um simulador comercial muito utilizado e cheio de recursos como ambiente virtual de testes em 3D. Este simulador possui uma versão sem limitação de recursos, que é de uso gratuito para fins não-comerciais (educação ou pesquisa), e também possui portabilidade para diversos sistemas operacionais. O robô real de baixo custo, também desenvolvido dentro do escopo deste projeto, é baseado em uma plataforma padrão para robótica móvel (chassi 2WD [7]), controlado através de um microcontrolador Arduino Mega [8]. Este robô foi inspirado em abordagens de construção de robôs de baixo custo para uso em ensino, como os robôs descritos em [9-11]. Este projeto é desenvolvido no LASER, Laboratório Avançado de Sistemas Embarcados e Robótica [12], na UTFPR, Câmpus Curitiba, sede Centro.

## MÉTODOS

Para o ensino dos fundamentos da programação, foi desenvolvido um ambiente gráfico para criação dos fluxogramas e denominado *Fluxoprog*. Há blocos pré-definidos, onde o programador pode arrastá-los para formar o programa (Figura 1). O fluxograma final é composto basicamente por comandos simples que são interpretados pelos robôs físico e real: Andar para frente (um passo), virar para a direita, e virar para a esquerda. Um “passo” é definido como um caminho entre duas intersecções em um ambiente quadriculado, como o apresentado na Figura 2. Virar para ambos os lados é definido como ficar ortogonalmente à posição inicial. Os comandos do fluxograma mostrado na Figura 1 instruem o robô a seguir pra frente por um “passo”, testar um sensor de ultrassom, e dependendo da leitura realizar mais um passo à frente, ou virar 90 graus no sentido anti-horário. Todas as partes do projeto foram programadas em C++, por essa linguagem ser aceita tanto pelo robô simulado quanto pelo robô real. Para o ambiente de programação foi usada a biblioteca *Allegro* no desenvolvimento da interface gráfica.

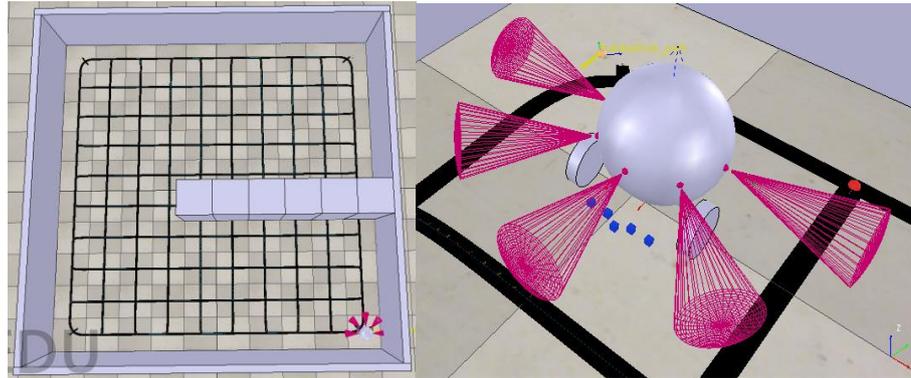
Figura 1 – Exemplo de fluxograma



Fonte: Autoria Própria (2018).

O ambiente simulado é feito no V-REP e apresenta um robô, com duas rodas, 5 sensores de proximidade em formato de cone simulando ultrassom, representados pelos cones vermelhos, e 5 câmeras simulando sensores de fita preta, representados pelos blocos azuis na figura 3. O V-REP uma ferramenta chamada *Remote API* para controle da simulação externa. Os sensores presentes no robô simulado foram pensados para realizar os seguintes propósitos: Seguir caminhos de fita preta, seguir um caminho com base num chão quadriculado de fita preta, seguir um labirinto com ou sem fita preta.

Figura 2 – Ambiente de Simulação e robô virtual em detalhe



Fonte: Autoria Própria (2018).

Foi desenvolvido um programa para controlar o robô simulado usando o *Remote API*. Este programa também envia comandos para o robô físico. Os comandos são recebidos do *Fluxoprog* e executados por um ciclo de tempo, enviando em seguida para o programa principal a leitura dos sensores. A comunicação entre o *Fluxoprog* e este programa é feita através de compartilhamento de memória, disponibilizado pela biblioteca *Boost:SharedMemory*. Todas as bibliotecas utilizadas foram escolhidas visando disponibilidade tanto nos sistemas operacionais *Linux* e *Windows*.

Após verificar que o robô simulado estava fazendo o que estava sendo requerido, foi projetado e montado o protótipo do robô real com as mesmas características. Como dito, a plataforma de desenvolvimento usada é o Arduino, já que é um hardware de fácil utilização e acesso pela maior parte da população, e o objetivo principal do desenvolvimento deste robô é ser de baixo custo e fácil construção por qualquer pessoa interessada. O ambiente físico usado é um banner impresso, contendo um caminho semelhante ao do V-REP. A ideia é que, se a programação gerada pelo fluxograma funcionar com o robô simulado, se execute a mesma coisa para o robô físico, de forma a se perceber as diferenças existentes entre um ambiente totalmente controlado e um semicontrolado, expostos a variações de terreno, luminosidade, entre outros. O caminho pode ser visto na Figura 3.

Para fazer a comunicação com o robô, foi escolhido o módulo HC-05, com base na tecnologia *Bluetooth*, que possui a limitação de ter uma comunicação um para um, mas como se espera usar *smartphones* no controle dos robôs posteriormente, o interfaceamento é facilitado. O mesmo programa que envia comandos aos motores do robô simulado foi usado para replicar o comando enviado do programa principal e enviar tais usando comunicação serial através do protocolo *Bluetooth*.

Os sensores de fita preta utilizados no robô fazem parte do módulo BFD-10005, que de extra traz um sensor de distância infravermelho e um detector mecânico de impacto. Este sensor se mostrou adequado para seguir fitas pretas de material fosco, porém para as de material brilhante é necessário isolar entre si o sensor central e os dois adjacentes, já que o material brilhante causa desvio dos infravermelhos, que acabam se cruzando entre si e atenuando o sinal do led receptor.

O sensor ultrassônico é o incluso no módulo HC-SR04, que possui distância

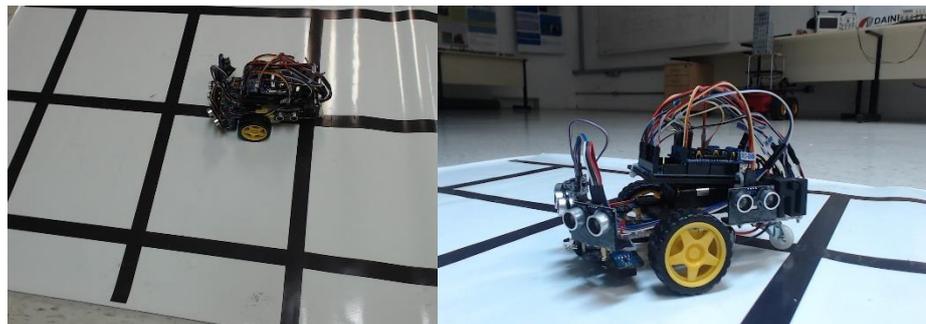
efetiva de captação dentro do limite de trabalho escolhido para o robô. Ele não capta alguns tipos de materiais não muito sólidos, que permitem a passagem das ondas de som. Nesses casos o sensor de distância do BFD-1000 entra em ação, mas ele também não capta alguns materiais, então em último caso é usado o sensor mecânico do mesmo.

Por fim, para movimentação do robô são usados dois motores DC com tensão de operação entre 3V e 6V. Como são motores baratos, não há muita precisão no controle dos mesmos, e não possuem encoder integrado. Para ligar e controlar a rotação dos motores é usado o Motor Shield L293D, de fácil conexão com o Arduino e podendo controlar até 4 motores DC. Para o controle da velocidade e direção do robô, são utilizados dois sensores do tipo encoder, módulo MH-Sensor-Series, que simulam um encoder analógico usando infravermelho. Para alimentação, são usadas 6 pilhas AA de 1,5V para os motores, e 4 para o circuito digital.

A carcaça do robô, juntamente com os dois motores DC 3-6V, respectivas rodas e discos encoder, foram adquiridos pelo kit 2WD. O chassi deste kit possui vários furos para acomodar os diversos equipamentos do robô, como base de pilhas, sensores encoder, entre outros.

Para a confecção das bases dos diversos componentes do robô foi usado modelagem 3D, de tal forma que ao final essa modelagem pode ser impressa numa impressora 3D. O modelo final do robô físico pode ser visto na figura 3. Há uma estrutura que pode ser erguida, onde estão o microcontrolador, o módulo ponte H e o módulo *Bluetooth*. Os demais componentes estão dispostos de maneira acessível no chassi do robô.

Figura 3 – Ambiente e robô físico



Fonte: Autoria Própria (2018).

## RESULTADOS E DISCUSSÕES

As primeiras avaliações práticas ocorreram nos dias 04/06/2018 e 08/06/2018, através de sua apresentação para 4 turmas de 20 estudantes do ensino médio técnico em eletrônica e mecânica na cidade de Curitiba. O desafio proposto foi usar os fluxogramas para controlar o robô simulado de forma que ele fosse de um canto de um ambiente ao outro, desviando de obstáculos no caminho. Com um mínimo de explicações, mesmo estudantes sem nenhum conhecimento prévio de conceitos de programação ou robótica foram capazes de

cumprir o desafio, tendo considerado o ambiente de criação de fluxogramas intuitivo. O fato de se identificar o funcionamento dos comandos através da visualização direta do robô realizando as ações comandadas, tornou a identificação de erros bastante simples e direta, e ao final de duas aulas (1 hora e 40 minutos) todos os estudantes conseguiram desenvolver “programas” funcionais.

Iniciou-se também a apresentação do projeto para professores da rede de ensino municipal, na cidade de Curitiba. Os professores também não necessitam ter um prévio conhecimento na área, e espera-se que isso ajude a aumentar o interesse dos alunos nesta área de conhecimento. Após uma palestra sobre educação com o uso da robótica, os professores recebem um desafio semelhante ao aplicado aos alunos, de tal forma que eles captem boa parte da lógica de programação, e repassem para seus alunos. Espera-se que o método aqui exposto comece a ser usado nas escolas municipais, em breve.

### CONSIDERAÇÕES FINAIS

Apesar de estar em pleno estado de desenvolvimento, o ambiente de programação já está funcional, estando disponibilizado para download no repositório <https://github.com/AlefeFelipe/LASER-fluxograma-e-robos>. Neste link podem ser encontradas as instruções de download, compilação e execução do ambiente de programação desenvolvido, bem como instruções de execução do simulador, permitindo o uso completo em atividades introdutórias de programação. O ambiente está disponível para Windows e Linux. Após a finalização do protótipo do robô físico, serão disponibilizadas informações de construção completas na página do LASER. Entretanto, mesmo sem a construção física deste robô, deve ser possível utilizar o ambiente de programação para prover o aprendizado inicial de programação de uma forma direta, lúdica e interessante.

Os próximos passos do desenvolvimento envolvem a criação de novos ambientes simulados, e maiores desafios de programação, tais como um ambiente onde os estudantes devam construir um programa que permita ao robô acompanhar uma linha preta enviando comandos de velocidade diretamente a cada um dos motores, ou interagindo com os sensores tanto de fita preta quanto de distância simultaneamente, para desviar obstáculos presentes sobre a fita. Tais situações são encontradas na categoria prática da Olimpíada Brasileira de Robótica (OBR) [13], e pretende-se que este sistema seja uma introdução/motivação para que mais equipes se formem e participem desta competição, a partir do primeiro contato com a programação e a robótica adquirido pelos estudantes através do uso do ambiente de programação/simulação desenvolvido no âmbito deste projeto.

### AGRADECIMENTOS

Os autores agradecem à Fundação Araucária pela concessão de bolsas. O primeiro autor agradece ao segundo autor, pela escolha do primeiro como bolsista, e aos demais pela participação ativa na criação do *Fluxoprog* e das peças suporte que compõem o robô físico, além da ajuda em outros requisitos do projeto.

## REFERÊNCIAS

- [1] Vargas, M. N., Menezes, A. G. C., Massaro, C. M., Gonçalves, T. M. Utilização da robótica educacional como ferramenta lúdica de aprendizagem na engenharia de produção: introdução à produção automatizada. 2012. XL Congresso Brasileiro de Educação em Engenharia-COBENGE, Belém, 2012.
- [2] Oliveira, D., Ferreira, S., Celestino, H., Ferreira, S., Abrantes, P. Uma proposta de ensino e aprendizagem de programação utilizando a Robótica Educativa e Storytelling. 2012. II Congresso Internacional TIC e Educação, Lisboa, Portugal, 2012.
- [3] Santos, F. L., Nascimento, F. M. S., Bezerra, R. M. S. REDUC: A Robótica Educacional como abordagem de baixo custo para o ensino de Computação em Cursos Técnicos e Tecnológicos. 2010. XVI Workshop sobre Informática na Escola, Encontro da Sociedade Brasileira de Computação, Belo Horizonte – MG, 2010.
- [4] Aranibar, D. B., Gurgel, V., Santos, M., Araujo, G. R., Roza, V. C., Nascimento, R. A., Silva, A. F., Silva, A. , Nascimento, L. M. G. Roboeduc: a software for teaching robotics to technological excluded children using lego prototypes. 2006. Anais do 3RD IEEE Latin American Robotics Symposium, Santiago, Chile, 2006.
- [5] MASCHIO, E. , DIRENE, A. I. . Múltiplas Representações Externas para o Ensino de Programação de Computadores. 2007. I Workshop de Ambientes de Apoio à Aprendizagem de Algoritmos e Programação – Simpósio Brasileiro de Informática na Educação – SBIE, São Paulo, 2007.
- [6] Coppelia Robotics. V-REP. 2018 Disponível em: <http://www.coppeliarobotics.com/>. Acesso em: Agosto/2018.
- [7] RobotShop. 2WD Beginner Robot Chassis. 2018. Disponível em: <https://www.robotshop.com/en/2wd-beginner-robot-chassis.html> - Acesso em: Julho/2018.
- [8] ARDUINO. What is Arduino? 2018. Disponível em: <https://www.arduino.cc/en/Guide/Introduction>. Acesso em: Agosto/2018.
- [9] Arvin, F., Espinosa, J., Bird, B. et al. “Mona: an Affordable Open-Source Mobile Robot for Education and Research”, Journal of Intelligent & Robotic Systems. 2018. Disponível em: <https://doi.org/10.1007/s10846-018-0866-9>. Acesso em: Agosto/2018.
- [10] Gyebi, E., Hanheide, M., Cielniak, G., et al.: Affordable mobile robotic platforms for teaching computer science at African Universities. 2015. In: 6th International Conference on Robotics in Education, 2015.

- [11] Yu, J., Han, S.D., Tang, W.N., Rus, D.: A portable, 3d-printing enabled multi-vehicle platform for robotics research and education. 2017. In: IEEE International Conference on Robotics and Automation (ICRA), pp. 1475–1480, 2017.
- [12] LASER - Laboratório Avançado de Sistemas Embarcados e Robótica – UTFPR – Campus Curitiba. 2018. Disponível em: <http://laser.dainf.ct.utfpr.edu.br/doku.php>. Acesso em: Agosto/2018.
- [13] Olimpíada Brasileira de Robótica – OBR. 2018. Disponível em: <http://www.obr.org.br> - Acesso em: Julho/2018.