Tecnologia Robótica e *Problem Based Learning* Aplicados ao Ensino de Lógica

Antonio Valerio Netto

DT CNPq
avalerionetto@gmail.com

Resumo: O artigo descreve o desenvolvimento de uma solução baseada em robótica educacional para promover o ensino de lógica e programação. As atividades envolveram a criação do *hardware* e a implementação da integração com Snap!, além da geração do material didático. No decorrer do desenvolvimento da solução, observou-se a necessidade de unir a aplicação do sistema robótico, a uma metodologia de ensino que pudesse apoiar o processo de disseminação junto aos professores e alunos da educação básica, especificamente, o sexto ano do ensino fundamental II. Diante disso, foi utilizada metodologia ativa, especificamente, *Problem Based Learning* (PBL) para a construção do material didático.

Palavras-chave: *robótica educacional; problem based learning; Snap!; ensino de lógica*.

1. Introdução

A robótica educacional é a aplicação da tecnologia na área pedagógica, sendo mais um instrumento que oferece aos alunos e professores, a oportunidade de vivenciar experiências semelhantes às que terão na vida real, dando a estes, a chance de solucionar problemas mais do que observar formas de solução. A robótica tem grande potencial como ferramenta interdisciplinar, visto que a construção de um novo mecanismo, ou a solução de um novo problema, frequentemente extrapola a sala de aula. A robótica assume o papel de uma ponte que possibilita religar fronteiras anteriormente estabelecidas, agindo como um elemento de coesão dentro do currículo das escolas do ensino básico [1].

Contudo, ainda persiste o desafio de implementar e promover corretamente a adoção da robótica educacional no Brasil. O primeiro problema está na falta de professores familiarizados com a sua aplicação em sala de aula. Posteriormente, ocorrem a existência de equipamentos sem manutenção, outros com tecnologia ultrapassada, além de quantidade insuficiente para atender o número de alunos [2]. Diante disso, é fundamental entender como realmente construir uma solução aplicando robótica educacional que tenha foco, metodologia de aprendizagem clara e que permita ao professor conduzir uma atividade de forma satisfatória para grande parte dos alunos e não somente para um grupo pequeno de interessados pelo assunto. O desafio está em como atingir a maioria dos alunos, incluído os desinteressados. Para isto, as plataformas educacionais envolvendo robótica devem se aprofundar no processo das metodologias ativas e encontrar o equilíbrio correto para atender os interesses tanto dos alunos quanto dos professores. No emprego da metodologia ativa [3], em oposição à aprendizagem passiva baseada na transmissão de informação, o aluno assume uma postura mais ativa, na qual ele resolve problemas, desenvolve projetos e, com isto, cria oportunidades para a construção de conhecimento.

Entre as metodologias ativas, a utilizada no projeto foi *Problem Based Learning* (PBL). Trata-se de uma proposta pedagógica que consiste no ensino centrado no estudante e baseado na solução de problemas, reais ou simulados. Os alunos, para solucionar esse problema, recorrem aos conhecimentos prévios, discutem, estudam, adquirem e integram os novos conhecimentos. Essa integração, aliada à aplicação prática, facilita a retenção do conhecimento. Portanto, o PBL valoriza, além do conteúdo a ser aprendido, a forma como ocorre o aprendizado, reforçando o papel ativo do aluno neste processo, permitindo que ele aprenda como aprender.

O PBL oferece diversas vantagens, como o desenvolvimento da autonomia, a interdisciplinaridade, a indissociabilidade entre teoria e prática, o desenvolvimento do raciocínio crítico e de habilidades de comunicação. Além disso, à medida que estimula uma atitude ativa do aluno em busca do conhecimento e não meramente informativa, como é o caso da prática pedagógica tradicional, o PBL caracteriza-se como uma metodologia formativa onde o elemento central é o aluno [4]. Os alunos são apresentados a um problema, pré-elaborado, e com a facilitação de um professor/tutor, são estimulados a discutir e elaborar hipóteses. Esta situação motivadora nos grupos leva a definição de objetivos de aprendizagem, que serão os estímulos para o estudo individual [5].

Ademais, o protagonismo dos estudantes no processo de aprendizagem é um importante fator motivacional, levando a busca ativa do conhecimento e gerando um aprendizado mais eficaz. A interdisciplinaridade é outra importante vantagem do PBL sobre o ensino tradicional. A substituição de conhecimento fragmentado, oferecido em disciplinas, por situações reais, que envolvam vários aspectos do conhecimento, favorece uma aprendizagem significativa, contextual e, ainda, promove a integração dos conteúdos curriculares do ciclo básico [6].

Com relação ao Snap! trata-se de uma linguagem de programação educacional e ferramenta de autoria multimídia baseado no Scratch (*https://scratch.mit.edu*) [7]. O Snap! é uma linguagem visual que utiliza características de arrastar e soltar (*drag and drop*). O principal aspecto desse ambiente está na implementação de algoritmos por intermédio de blocos construtivos.

Essas ferramentas são adequadas para introduzir os estudantes na área de ciência da computação, visto que o processo de desenvolvimento de programas computacionais se torna mais intuitivo e direto. Um detalhamento sobre a utilização do Snap! pode ser encontrado no manual de referência acessado pelo *link*: *https://snap.berkeley.edu/snap/help/SnapManual.pdf*
Todos os componentes do sistema de programação Snap! são gratuitos e de código aberto, sob uma Licença Pública Geral.

2. Metodologia

O desenvolvimento foi dividido em atividades relacionadas à criação do *hardware* do robô SuperKid (projeto mecânico, eletrônico e do software embarcado), construção de um *middleware* para realizar a integração dele com o Snap! e a transmissão de comandos via *bluetooth*. Além disso, paralelamente, foi realizada a criação do material educacional para apoiar o uso do kit por parte de alunos e professores.

Baseado nas especificações do SuperKid (levantadas na fase de visão e validadas na fase de concepção) foi realizado um planejamento dos custos de produção por lotes. Utilizou-se a eletrônica do robô Kid (*http://www.xbot.com.br/educacional/kid*) como base para iniciar a criação do Super Kid. A meta era trabalhar com um microcontrolador de baixo custo para suportar o *software* embarcado responsável pelo controle tanto dos motores (mobilidade) quanto dos sensores (infravermelho) e atuadores (leds e buzina). A meta era que o robô tivesse um custo de fabricação baixo já prevendo um carregador e baterias no lugar das pilhas e a presença de uma carenagem. Essas características não existem no Kid.

Durante o projeto chegou-se a pesquisar a utilização do Arduino com o objetivo de trocar o processador original do Kid (microcontrolador PIC 16F876A). O Arduino é uma plataforma de prototipagem eletrônica de *hardware* livre, projetada com um microcontrolador Atmel AVR de placa única, com suporte de entrada/saída embutido que utiliza uma linguagem de programação C/C++. Uma típica placa Arduino é composta por um controlador, algumas linhas de E/S digital e analógica, além de uma interface serial ou USB, para interligar-se ao hospedeiro, que é usado para programá-la em tempo real. Ela, em si, não possui nenhum recurso de rede, porém é comum combinar com extensões chamadas de *shields*. Um desses *shields* poderia permitir a comunicação via *bluetooth*. Contudo, devido ao custo financeiro envolvido, a decisão final foi manter o *hardware* do Kid, acrescentando um *chip bluetooth* na eletrônica embarcada. A placa fabricada pelo Kid com a modificação era mais barata que a aquisição do Arduino + *shield* para comunicação via *bluetooth*. O projeto também considerou a questão da autonomia da bateria do *hardware* que deveria durar pelo menos entre seis a oito horas. Foi modificada a placa original do Kid para ser capaz de conectar um carregador de tomada para carregar a bateria que substituiu o “case” de pilhas.

Posteriormente, o foco foi na construção da biblioteca com as funções de controle do robô como, por exemplo, movimentar para frente, para trás, parar, ligar e desligar buzina, acender e apagar leds. Dessa forma, os blocos desenvolvidos podem ser utilizados dentro do ambiente de programação do Snap! Os comandos que representam cada bloco são enviados via *bluetooth* para o robô executar. Para isto foi necessário realizar um estudo sobre o Snap! e entender de que forma a plataforma executada via *browser* poderia ser comunicar com o meio externo. Durante essa revisão foi identificado um caso de uso com o lego *mindstorm* (*https://scratch.mit.edu/ev3*) que serviu com o referência. Também foi encontrado um *site* que relata experiências de como realizar a conexão com o Scratch (*https://education.lego.com/en-us/support/mindstorms-ev3/scratch*), porém na época da pesquisa, não foi encontrado material para integração com o Snap!

Foi definida uma arquitetura que privilegiou a menor quantidade de interações para evitar consumo de energia no robô e problemas de erro de comunicação que poderiam desestimular o aluno a interagir com o kit educacional. Na Figura 1 é exibido o robô SuperKid. Na Figura 2 é apresentada a interface do Snap! com os comandos que foram criados (lado inferior esquerdo da figura – blocos construtivos em azul).

Figura 1. Visão geral do robô construído (SuperKid).

Inicialmente, foram montados manuais de uso que permitem, tanto o aluno quanto o professor, manipular os comandos do robô, instalar o sistema e executar exemplos de exercícios com os blocos construtivos. Também foram montadas videoaulas explicando detalhadamente cada atividade. Além disso, os professores envolvidos tiveram um *workshop* presencial de 16 horas (dois dias) para aprender a trabalhar com o material em sala de aula. Entendia-se que seria o suficiente para que pudessem utilizar o ferramental e montar suas próprias atividades, ou mesmo, utilizar aquelas que já tinham sido construídas. Contudo na fase de validação com três turmas do sexto ano do ensino fundamental II (total de 72 alunos), o processo de adoção da tecnologia não ocorreu de forma satisfatória com baixos índices de adesão por parte dos alunos.

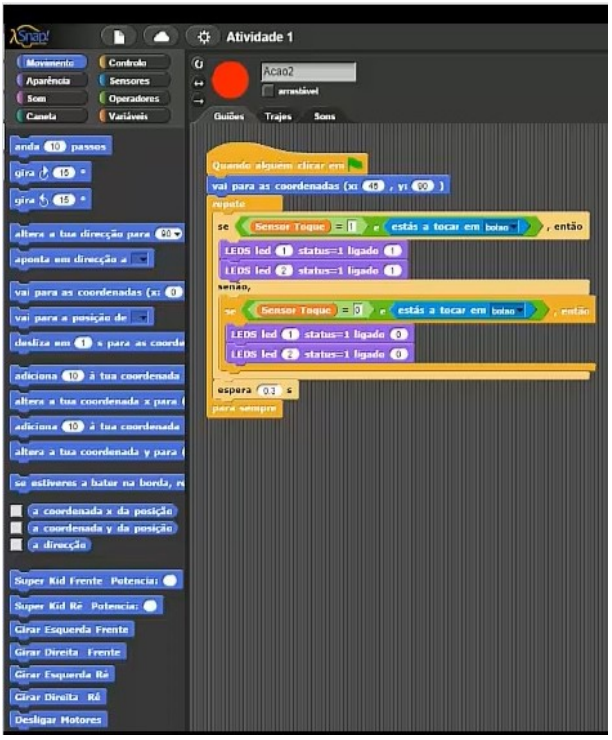

Figura 2. Interface do Snap! com os comandos desenvolvidos.

Diante dessa situação e baseados na experiência de um projeto de aprendizagem para disciplina de Introdução à Computação (IC) para graduação [8], que trabalhou com fuga de labirinto e conceitos envolvendo PBL, foram criados Planos de Aulas (PAs) com conteúdos para que os alunos pudessem utilizar o kit educacional desenvolvido. Além disso, eles serviram de base para os professores gerarem novas atividades que pudessem ser utilizados nos laboratórios de informática ou espaço *maker* das escolas. Os PAs foram montados com tempo estimado para serem realizados durante uma aula padrão (45 minutos). Cada PA tem um tema (disciplina) e um subtema. Para etapa de validação foram criados PAs com o objetivo de ensinar linguagem de programação com o subtema relacionado à instrução condicional. No caso do ensino de lógica foi montado um PA sobre aprendizagem de conjuntos (união, intersecção, etc).

3. Resultados

Neste projeto foi construída uma biblioteca de comandos para o controle do robô: (1) movimentar para frente, (2) para trás, (3) parar, (4) ligar buzina, (5) desligar buzina, (6) acender leds e (7) apagar leds. Esses blocos são utilizados dentro da programação normal do Snap! e os comandos que representam cada bloco são enviados via *bluetooth* para o robô executar. Os comandos foram criados no formato “xml”. A seguir é apresentado um exemplo: no caso o comando “Girar Esquerda Frente”:

```
<blocks app="Snap! 4.0, http://snap.berkeley.edu"
version="1"><block-definition s="superkid parar
motores" type="command" category="other">

<code></code><inputs></inputs><script><block
s="doRun"><block s="reportURL"><l>
localhost:3611/parar</l></block><list></list>
</block></script></block-definition>

<block-definition s="Girar Esquerda Frente"
type="command" category="motion"><code></code>
<script><block s="doReport"><block s="reportURL">
<block s="reportJoinWords"><list><l>
localhost:3611/gef</l><l>-</l></list></block>
</block></block></script></block-definition>
```

Com relação ao servidor de comunicação, foi desenvolvido um aplicativo que realiza a conexão ponto a ponto via *bluetooth* entre o robô e computador instalado com Snap! Este aplicativo é responsável por realizar o pareamento com o robô antes de iniciar as atividades didáticas. A Figura 3 mostra a interface desse aplicativo. O usuário deverá escolher uma porta serial previamente definida em uma lista disponibilizada pelo computador e salvar. Depois, basta clicar no botão “iniciar servidor” para ligar a comunicação, e posteriormente, clicar no botão “finalizar servidor” para desligar essa comunicação.

Figura 3. Aplicativo “porta.exe” responsável por habilitar o servidor de comunicação entre o robô e o Snap!.

Ao iniciar o servidor, o robô exibe um sinal visual quando o mesmo está conectado ao computador. Dessa forma, facilita ao usuário saber se a comunicação está funcionando corretamente. O aplicativo desenvolvido é responsável por encapsular as palavras e transmitir os comandos seguindo um protocolo de comunicação robusto que evita perda de comandos e sobreposição da pilha de execução. Ele é capaz de interpretar os comandos vindos do Snap! e traduzir para este protocolo. A comunicação é bidirecional, isto é, do Snap! para este programa e vice versa. Isto se deve aos comandos condicionais que necessitam da resposta do robô. Não basta apenas enviar os dados, o Snap! precisa também recebê-los para executar a programação em tempo real.

Foram realizadas 10 sessões com atividades para testar os PAs criados com um grupo de oito alunos. O objetivo era verificar se a nova abordagem pedagógica surtiria efeito positivo no engajamento dos alunos e no aprendizado do conteúdo em si. Os testes foram realizados com o subtema: Instrução Condicional. A seguir, exemplos de atividades para os alunos:

1. Utilizando os blocos construtivos, faça com que o robô, ao ser iniciado, verifique todos os seus sensores de distância. Se houver algum obstáculo próximo a algum destes sensores, o robô deverá girar para o lado oposto e "fugir" do obstáculo. Não utilizar instruções de laço.
2. Utilizando os blocos construtivos e somente o sensor frontal construa uma solução para que o robô possa sair do labirinto o mais rápido possível e no menor número de interações.

4. Discussões e Conclusões

Este trabalho buscou aplicar tecnologia robótica para promover o engajamento não somente dos alunos, mas dos professores, inclusive aqueles que não são da área de informática/computação. Os primeiros testes com o público alvo (fase de validação) demostraram nitidamente que se não fosse adaptada uma metodologia ativa para promover as atividades, qualquer ferramenta construída padeceria de adoção da mesma em sala de aula, entende-se, laboratório de informática. Algumas vezes, por conta do aluno, e quase sempre, por conta do próprio professor/ instrutor.

Mesmo contando com o arcabouço do conceito da Hora do Código (*https://hourofcode.com/br*), o mesmo não se mostrou suficiente para promover o envolvimento dos professores no processo educacional, pois muitos deles não eram da área de informática/ computação ou não estavam imbuídos da mesma doutrina. Diante disso, foi necessário estudar as metodologias ativas e identificar de que forma elas poderiam auxiliar no processo de adoção e engajamento da tecnologia educacional proposta.

As principais vantagens pedagógicas da robótica, principalmente associada ao método PBL que puderam ser observadas estão relacionadas ao desenvolvimento do raciocínio e na lógica da construção de algoritmos e programas para controle dos mecanismos. É importante salientar que nos anos escolares iniciais devem ser trabalhados conceitos relacionados às estruturas abstratas necessárias à resolução de problemas no eixo do "Pensamento Computacional". É importante que o aluno tome consciência do processo de resolução de problemas, e compreenda a importância de ser capaz de descrever a solução em forma de algoritmo. Nesta etapa, os alunos são expostos à noção básica de algoritmos quando, por exemplo, ensinam-se as operações aritméticas básicas. A expectativa é que isso seja enfatizado, de forma que os alunos entendam noções básicas de algoritmo, sendo capazes de, a partir de conjuntos de instruções diversas, conseguir elaborar algoritmos para solucionar diferentes tipos de problemas.

Agradecimento

Ao apoio do CNPq por meio do seu programa de Desenvolvimento Tecnológico e Extensão Inovadora (DT) e do programa de Formação de Recursos Humanos em Áreas Estratégicas (RHAE).

Bibliografia

[1] Valerio Netto, A.; Gonçalves, L. M. G. (2006). Robótica Computacional e Robótica Educacional: ferramentas para o conhecimento e inclusão tecnológica, Grandes Desafios da Computação no Brasil, SBC. disponível em <http://www.xbot.com.br/wp-content/uploads/2012/10/Grandes_1-0.pdf>, acesso em 09/12/2019.

[2] Peralta, D. A.; Guimarães, E. C. (2018). A robótica na escola como postura pedagógica interdisciplinar: o futuro chegou para a Educação Básica?. *Revista Brasileira de Informática na Educação* 26(01): 30.

[3] Rocha, H. M.; Lemos, W. D. (2014). Metodologias ativas: do que estamos falando? Base conceitual e relato de pesquisa em andamento. IX Simpósio Pedagógico e Pesquisas em Comunicação. Resende, Brazil: Associação Educacional Dom Boston, 12.

[4] Cabral, H. D. S. R.; Almeida, K. K. V. G. (2014). Problem based learning: aprendizagem baseada em problemas. *Revista Interfaces: Saúde, Humanas e Tecnologia* 2(4).

[5] Thomson J. C. (1996). PBL uma proposta pedagógica. Olho Mágico, v. 2, n.7.

[6] Santos, J. A. M.; Angelo, M. F. (2009). Análise de problemas aplicados em um estudo integrado de programação utilizando PBL. In: Anais: XI Workshop sobre Educação em Computação–XXIII Congresso da Sociedade Brasileira de Computação, RS.

[7] Snap! (2017). Disponível em <https://snap.berkeley.edu/>, acesso em 09/02/2017.

[8] Valerio Netto, A. (2020). Proposta de aplicação de robô móvel para apoio ao ensino de lógica de programação. *RECIT* 11(26): 47-69.