

Avaliação Experimental do Robomind no Ensino de Programação com Estudantes do Curso Técnico em Informática Integrado ao Ensino Médio

Leonardo S. e Silva¹, Elmano R. Cavalcanti¹

¹Coordenação de informática – Instituto Federal de Pernambuco (IFPE)
Garanhuns – PE – Brazil

{leonardo.silva,elmano.cavalcanti}@garanhuns.ifpe.edu.br

Abstract. *To minimize difficulties in programming learning, new pedagogical strategies were proposed. Robomind is an example, this educational software promotes learning through play. This study aims to provide statistical evidence about the pedagogical effectivity of this tool, through an experiment with 65 students from technical high school. Statistical test did not found significant result (p -value < 0.05) in the learning outcomes among the group of students exposed to the intervention with Robomind and the non-intervention group. Effect size calculation resulted in $r = 0.08$. On the other hand, the tool proved to be adequate to avoid student evasion.*

Resumo. *Com o objetivo de minimizar as dificuldades no processo de ensino e aprendizagem de programação, novas estratégias pedagógicas foram propostas. O Robomind é um exemplo, esse software educacional promove o aprendizado por meio do lúdico. Este trabalho busca oferecer evidências estatísticas sobre a efetividade dessa ferramenta, por meio de um experimento com 65 estudantes do ensino técnico integrado ao ensino médio. Os cálculos estatísticos não identificaram diferenças, estatisticamente significantes (p -value < 0.05), nos resultados de aprendizado entre o grupo de estudantes expostos à intervenção com o Robomind e o grupo sem intervenção. O cálculo do tamanho de efeito resultou em um $r = 0.08$. Por outro lado, a ferramenta se mostrou adequada para minimizar evasões dos estudantes.*

1. Introdução

O Robomind¹ é um exemplo de Tecnologia da Informação e Comunicação (TIC) para o apoio ao ensino de programação. Por meio de uma interface visual o estudante programa o comportamento de um robô que deve ser capaz de resolver diferentes problemas e desafios (e.g., seguir uma linha de cor branca) e a interatividade presente neste processo cria uma experiência lúdica para os estudantes. Diferentes conceitos de programação podem ser trabalhados, como o pensamento lógico, condições, repetições, variáveis, entre outros.

Apesar dos aparentes benefícios, evidências estatísticas são necessárias para validar o uso pedagógico dessa ferramenta. Este estudo se insere nesse contexto e tem por objetivo avaliar os ganhos de aprendizado de programação por estudantes expostos à intervenção com o Robomind e sua capacidade em diminuir a evasão. Para realizar essa

¹<http://robomind.net/>

investigação foram propostas duas questões de pesquisa: a) o uso do Robomind no ensino de programação para estudantes de nível médio promove mais ganhos no aprendizado em comparação a um ensino tradicional? e b) o uso do Robomind reduz os índices de evasão de estudantes em comparação aos estudantes que receberam aulas em um formato de ensino tradicional?

Para responder a essas questões de pesquisa, conduziu-se um experimento com 65 estudantes do curso técnico em informática integrado ao ensino médio do Instituto Federal de Pernambuco - Campus Garanhuns. Os estudantes, matriculados na disciplina de introdução à programação, foram divididos em dois grupos: experimental e controle. O grupo experimental utilizou o Robomind em parte do ano letivo e o grupo controle não o utilizou, adotando uma metodologia tradicional de ensino, baseada nas linguagens português e java. Ao término do ano letivo, foram coletadas as notas finais dos estudantes na disciplina, que no contexto deste experimento serão usadas para mensurar o aprendizado em programação.

Testes estatísticos foram utilizados para responder as questões de pesquisa estabelecidas. Para a questão a), utilizou-se uma comparação das medianas das notas finais dos grupos, não sendo observada diferença estatisticamente significativa. O tamanho do efeito (*effect size*) do fenômeno investigado (aprendizagem) também foi calculado e o resultado foi de 0.08. Em relação a questão b), foi comparado o quantitativo de evadidos e não evadidos, o resultado indica que o grupo experimental apresentou menor evasão.

O presente estudo fornece evidências científicas e estatísticas que poderão subsidiar docentes no uso de tecnologia no ensino de programação. Esses resultados somados aos de outros estudos já publicados e que também avaliaram o Robomind, ajudam a construir evidências sobre a ferramenta.

2. Trabalhos relacionados

É fundamental para a área de ensino de programação que existam evidências que comprovem a eficácia pedagógica de ferramentas educacionais. Assim, professores poderão optar por intervenções que possuam comprovações estatísticas sobre sua efetividade.

Os resultados encontrados em [Yuana and Maryono 2016] indicam que 87% dos participantes consideram que o Robomind é motivador para o estudo de programação. Também foi identificado que 80% dos conteúdos de programação foram compreendidos. O trabalho apresenta algumas falhas, em especial na instrumentação experimental, pois não fica claro como o aprendizado em programação foi medido. Diferente de nosso estudo, não foi descrito o delineamento experimental, uso de grupos controle, entre outros importantes elementos de um estudo empírico, o que dificulta a comparação dos resultados entre estudos.

Os autores de [Nofitasari et al. 2017] realizaram um experimento no qual os alunos participam de uma intervenção com o Robomind para aprender programação. Apesar da ausência de testes estatísticos, os autores avaliam que a ferramenta atende aos propósitos de ensino de programação e também motiva os estudantes. Assim como no trabalho anterior, a ausência de dados estatísticos e informações sobre o experimento dificulta a comparação entre trabalhos.

Uma intervenção com objetivo de preparar estudantes para a Olimpíada Brasileira

de Informática (OBI) foi realizada em [Schoeffel et al. 2015], nesse processo os autores utilizaram as ferramentas Scratch² e Robomind. Os resultados indicam que os estudantes consideraram as aulas divertidas, que gostaram de criar programas e apresentaram percepções positivas sobre a área de computação. Os autores também analisaram as médias dos estudantes na OBI pós-intervenção, comparando com um grupo controle que não participou da intervenção. Houve diferença estatisticamente significativa entre os grupos, favorável ao grupo experimental. Esse resultado difere do encontrado por nós, o qual não foi observada significância estatística comparando os resultados do grupo experimental e controle. Entretanto, em razão da intervenção também utilizar o Scratch, isso inclui uma ameaça à validade interna do experimento, de modo que não é possível avaliar o impacto direto produzido somente pelo Robomind.

O estudo [Mafulah et al. 2017] apresenta um experimento com estudantes de um curso superior em multimídia, durante o aprendizado de programação. O grupo experimental utilizou a ferramenta Robomind, enquanto que o grupo controle adotou uma abordagem tradicional, por meio de uma linguagem de programação profissional. Comparando os resultados entre grupos, após a intervenção, observa-se que existe diferença significativa, favorável ao grupo experimental. Este trabalho apresenta um contexto experimental similar ao nosso estudo, porém com resultado diferente no que se refere aos ganhos de aprendizado.

Em [Durães 2015] os autores propõem um oficina para estimular o aprendizado em diferentes temas, como programação, matemática e geografia. O experimento possui um grupo controle e experimental, que fez uso do Robomind e Lego MindStorms. Os resultados da indicam que o grupo experimental apresentou melhores resultados nas atividades de programação, em comparação ao grupo controle. A ausência de testes estatísticos dificulta a comparação desses resultados com o encontrado por nós. Assim como em [Schoeffel et al. 2015], a adoção de outro recurso pedagógico no experimento é uma ameaça à validade interna do estudo.

Em [Benitti et al. 2009] o Robomind é adotado em uma oficina voltada ao ensino de disciplinas como matemática, geografia e programação. Dos estudantes que participaram da oficina, 75% avaliaram como ótima e 25% a consideraram uma boa experiência. Além disso, em 80% das questões, houve melhoria após a intervenção. Infelizmente não foram realizados testes estatísticos e assim a comparação entre estudos é prejudicada.

A intervenção proposta em [Conchinha et al. 2016] buscou trabalhar diferentes conceitos, dentre eles a programação, com estudantes com necessidades especiais. Observou-se que todos os estudantes que participaram da atividade sentiram-se satisfeitos e motivados com a intervenção. Diferente deste estudo, não foi avaliado o conhecimento em programação dos estudantes.

Os experimentos conduzidos por [Koorsse et al. 2015] buscaram avaliar a eficácia de diferentes ferramentas no aprendizado de programação: Scratch, Robomind e a linguagem B#. Estudantes foram alocados em grupos experimentais, no qual cada um utilizou uma ferramenta, e os resultados de aprendizagem foram comparados ao grupo controle que não utilizou essas ferramentas e adotou o uso direto de linguagens de programação profissionais. O Robomind foi avaliado em diferentes tópicos de

²<https://scratch.mit.edu/>

programação, como funções, condições, repetições e variáveis. O contexto experimental é similar ao nosso, a amostra é composta por estudantes do ensino médio, e o conteúdo curricular é similar. Assim como em nosso experimento, não foi observada diferença estatisticamente significativa entre os grupos controle e experimental.

Em razão das diferentes estratégias de avaliação utilizadas, ausência de testes estatísticos e delineamento dos experimentos, em alguns experimentos, a comparação dos resultados deste estudo com alguns dos trabalhos citados é difícil. Sugere-se que trabalhos futuros adotem instrumentos padronizados de avaliação, como também planejem seus experimentos e resultados estatísticos de modo a possibilitar a realização de comparações por meio de meta-análise.

3. Metodologia

Este estudo se caracteriza como quasi-experimental e busca prover respostas às questões de pesquisa por meio de intervenções controladas em uma amostra, não aleatória. A metodologia experimental adotada foi baseada na proposta de [Kirk 1982] e consistiu em: i) definição da questão de pesquisa, ii) definição das variáveis dependentes e independentes, iii) caracterização da amostra, iv) construção das hipóteses, v) definição das ameaças à validade, vi) planejamento estatístico e vii) execução do experimento.

3.1. Definição das questões de pesquisa

As questões de pesquisa definidas, foram: a) o uso do Robomind no ensino de programação para estudantes de nível médio promove mais ganhos no aprendizado em comparação a um ensino tradicional? e b) o uso do Robomind reduz os índices de evasão de estudantes em comparação aos estudantes que receberam aulas em um formato de ensino tradicional?

3.2. Definição das variáveis dependentes e independentes

Neste estudo foi utilizada uma variável independente denominada metodologia de ensino, com dois níveis: a) uso do Robomind e b) ensino tradicional. Três variáveis dependentes foram observadas: a) nota final (0 a 10) dos estudantes na disciplina ao término do ano letivo, b) o quantitativo de aprovados (média superior ou igual a 6) e reprovados e c) quantitativo de evadidos, que foram os estudantes que não compareceram às duas últimas avaliações da disciplina (de um total de quatro avaliações), para mensurar a evasão.

Alguns fatores foram controlados com o objetivo de minimizar a sua influência nas variáveis dependentes: a) os estudantes são todos do turno vespertino (as médias de notas de ingresso dos estudantes na instituição diferem entre turnos); b) apenas foi considerada a nota referente a primeira vez que o estudante cursou a disciplina, sendo descartada as notas de estudantes que cursaram novamente a disciplina em razão de reprovação no ano anterior, pois esse conhecimento prévio poderia enviesar os resultados do estudo; c) a mesma linguagem de programação foi utilizada em ambos os grupos; e d) foi lecionado o mesmo conteúdo programático na disciplina.

3.3. Caracterização da amostra e contexto pedagógico

Uma amostra de 65 estudantes de duas turmas do primeiro ano do curso técnico de informática integrado ao ensino médio do Instituto Federal de Pernambuco - Campus

Garanhuns, participaram do experimento. Os alunos do primeiro ano apresentam faixa etária entre 13 a 15 anos e da amostra que participou do experimento, 36 (55,4%) eram do gênero masculino e 29 (44,6%) do gênero feminino. Considerando o tamanho da amostra e o tamanho da população (estudantes de nível médio do Brasil) de 8,1 milhões [INEP - Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira 2017], a amostra obtida é representativa para um nível de confiança de 95% com margem de erro de 12,3%.

Foram criados dois grupos: experimental (n = 30) e controle (n = 35). O grupo experimental recebeu a intervenção com o Robomind, enquanto o grupo controle utilizou uma metodologia de ensino tradicional. No grupo experimental, 56,6% eram do gênero masculino e 43,4% do gênero feminino. No grupo controle, 54,2% eram do gênero masculino e 45,8% eram do gênero feminino.

No curso técnico em informática os estudantes recebem qualificação profissional para atuar como desenvolvedores de software, ao mesmo tempo em que cursam o ensino médio. No primeiro ano de curso os alunos são apresentados à disciplina de introdução à programação. Trata-se de uma matéria que é lecionada semanalmente por 1h30 minutos, por um período de um ano e totaliza 54 horas de aulas. As aulas ocorrem em um laboratório de informática com um computador por estudante. Nessa disciplina os alunos trabalham os conceitos de pensamento algorítmico, repetições, condições, variáveis, funções e arrays.

Em geral, os conteúdos são apresentados com uma metodologia tradicional de ensino com uso da linguagem de programação Java. Consideramos como ensino tradicional aquele que se baseia em linguagens convencionalmente usadas em disciplinas introdutórias de programação, tais como português, pascal, C, python ou java. Ademais, também julga-se como tradicional o uso exclusivo de pseudo-códigos e fluxogramas para explicação de conceitos como estruturas de condição e repetição.

Em razão do campus em que a intervenção foi realizada apresentar elevados índices de reprovação dos estudantes na disciplina (acima de 50%) [Silva 2018], optou-se pelo uso e avaliação da ferramenta Robomind. Ela possui alguns diferenciais como um ambiente de interatividade visual de programação e a possibilidade de escrever programas em português.

Em geral, o conteúdo programático não difere muito da sequência de assuntos normalmente vistos em uma disciplina de Introdução à Programação ou Algoritmos. Inicia-se com a apresentação do robô e seus comandos básicos de movimento, visão e ação (e.g., segurar objetos). Em seguida, pode-se partir para a definição de variáveis e operadores básicos (caso se use a versão 4.0 ou superior), estruturas de condição simples e encadeadas, estruturas de repetição, procedimentos e parâmetros. Após isso, pode-se explicar o conceito de recursividade. Após as primeiras aulas, já é possível apresentar a definição teórica e prática de depuração de código, de maneira que os estudantes possam testar e encontrar erros em seus algoritmos.

Em razão das limitações de estrutura e recursos humanos, optou-se por realizar o experimento em dois anos. No primeiro ano, o grupo experimental utilizou o Robomind por um período aproximado de seis meses e posteriormente fez a transição para a linguagem de programação Java. Em outro ano, a turma do grupo controle foi exposta

a uma metodologia tradicional de ensino de programação, utilizando diretamente a linguagem de programação Java. Em razão da ausência de aleatoriedade na seleção das turmas para cada grupo, definiu-se assim este experimento como quasi-experimental.

Em ambos os grupos, controle e experimental, os alunos realizaram no mínimo quatro avaliações, uma a cada bimestre. Quando não alcançaram a nota de aprovação nessas avaliações (nota superior ou igual a 6 de um total de 10 pontos), eles realizaram outra avaliação denominada de recuperação de aprendizagem, também com nota de 0 a 10. Quando a nota na avaliação de recuperação era maior que a nota de avaliação do bimestre, ela substituía a primeira nota. Ao término do ano letivo a nota final do estudante é calculada a partir da média das notas dos quatro bimestres. Nessas avaliações são utilizadas questões abertas e fechadas sobre os conteúdos de programação citados anteriormente. Os alunos fazem no papel, sem auxílio do computador.

3.4. Construção das hipóteses de investigação

Para responder as questões de pesquisas, três hipóteses nulas foram formuladas: H_{0a} : Não há diferença no desempenho em introdução à programação dos estudantes que participaram da intervenção com Robomind comparados ao método tradicional de ensino de programação. H_{0b} : Não há associação entre o número de aprovações/reprovações na disciplina de introdução à programação e o tipo de intervenção adotada. H_{0c} : Não há associação entre o número de evasões dos estudantes na disciplina de introdução à programação e o tipo de intervenção adotada.

3.5. Ameaças à validade

Foram identificadas três ameaças à validade interna do experimento. A primeira, por motivos institucionais, foi que diferentes professores lecionaram para os grupos experimental e controle, o que por consequência pode influenciar o resultado da variável dependente. A segunda ameaça refere-se à ausência de um pré-teste para mensurar a homogeneidade, entre grupos, no conhecimento prévio em programação dos estudantes. Por fim, a evasão, que foi uma variável analisada, pode estar relacionada à motivação intrínseca e extrínseca dos estudantes e este comportamento não foi mensurado com instrumentos específicos.

4. Resultados

Para fins de possibilitar a replicação desse estudo, todos os dados coletados e cálculos estatísticos estão disponíveis em: <https://dx.doi.org/10.6084/m9.figshare.7088594>. Todos os cálculos estatísticos foram feitos por meio da ferramenta R³.

4.1. Resultados na disciplina

O grupo experimental apresentou nota final média de 3,85, enquanto que o grupo controle apresentou nota final média de 3,69. A mediana do grupo experimental foi de 3,7 e no grupo controle foi de 2,2. O desvio padrão do grupo experimental foi de 2,39, e no grupo controle foi de 3,03.

No grupo experimental houve um total de 11 aprovações (36,6%) e no grupo controle 16 (45,7%), que são estudantes que alcançaram média final superior ou igual à 6 na

³<https://www.r-project.org/>

disciplina. No grupo experimental houve 2 evasões (6%), enquanto que no grupo controle houve 16 (45,7%). Foram considerados evadidos os estudantes que não realizaram as avaliações dos últimos dois bimestres, de um total de quatro bimestres.

4.2. Testes de hipótese

O procedimento de teste da hipótese H_{0a} consistiu de quatro etapas: i) verificação da normalidade dos dados, ii) escolha do teste estatístico, iii) execução dos testes e iv) análise dos resultados.

A análise de normalidade dos dados foi verificada por meio do teste de Shapiro-Wilk, uma vez que este teste é adequado considerando o tamanho da amostra deste estudo [Razali et al. 2011]. Em razão do teste de normalidade indicar a ausência desta distribuição nos dados, optou-se pela escolha de testes de hipótese não paramétricos, pois nessas situações eles apresentam maior poder estatístico que os testes paramétricos [Cleophas and Zwinderman 2011].

Para testar a hipótese H_{0a} , considerando o quantitativo de variáveis dependentes e independentes, como também os tipos de dados, utilizou-se o teste de Wilcoxon. O resultado do teste indicou que não há diferença estatisticamente significativa, de aprendizado de programação, entre os grupos ($W = 550.5$, $p\text{-value} = 0.7419$). Assim, não é possível rejeitar a hipótese H_{0a} . Para calcular o tamanho de efeito utilizou-se a equação proposta em [Pallant 2007]: $r = Z/\sqrt{N}$. O resultado encontrado foi de $r = 0.08$, que de acordo com a literatura citada é considerado insignificante.

Os testes das hipóteses H_{0b} e H_{0c} foram feitos por meio do teste de Qui-Quadrado de Pearson, que possibilita avaliar a associação de dependência da variável independente: tipo de intervenção, nas variáveis dependentes: resultado final e a evasão na disciplina [McHugh 2013]. Ou seja, verificar se o resultado de aprendizagem e a evasão são dependentes do tipo de intervenção adotado. Esta escolha se deu em razão do uso de variáveis nominais que representam os valores das variáveis dependentes: resultado final (aprovação e reprovação) e situação do estudante (evadido ou não evadido), como também da variável independente (uso do Robomind e metodologia tradicional).

O resultado do teste de Qui-Quadrado indicou não haver significância estatística na associação entre o resultado final do estudante (aprovado ou reprovado) e a intervenção a qual eles foram expostos ($X\text{-squared} = 0.23569$, $df = 1$, $p\text{-value} = 0.6273$). Assim, não é possível rejeitar a hipótese H_{0b} .

Com relação à evasão, a aplicação do teste de Qui-Quadrado indicou haver associação de dependência com significância estatística entre a evasão e a intervenção a qual os estudantes foram expostos: $X\text{-squared} = 10.428$, $df = 1$, $p\text{-value} = 0.001241$. Assim, é possível rejeitar a hipótese H_{0c} .

5. Análise dos resultados

A análise dos resultados será apresentada seguindo a ordem das questões de pesquisa estabelecidas anteriormente.

5.1. a) o uso do Robomind no ensino de programação para estudantes de nível médio promove mais ganhos no aprendizado em comparação a um ensino tradicional?

Para avaliar ganhos de aprendizado utilizou-se duas informações dos estudantes: a nota final e o número de aprovações e reprovações na disciplina de introdução à programação.

Utilizou-se um teste estatístico para verificar se resultados dos grupos experimental e controle, diferem. A comparação das medianas das notas finais dos estudantes, entre os grupos, por meio do teste de Wilcoxon, não indicou haver diferença estatisticamente significativa para um nível de significância de 95%. Assim, não é possível rejeitar a hipótese nula H_{0a} . O tamanho de efeito encontrado (0.08) é considerado irrelevante, comprovando que de fato não há diferenças entre os dois grupos.

Contrariando as expectativas iniciais dos autores, a abordagem tradicional promoveu menos reprovações que o grupo experimental. No entanto, ainda sim a diferença não é estatisticamente significativa. Portanto, não se rejeita a hipótese H_{0b} .

Esses resultados corroboram os resultados do trabalho [Koorse et al. 2015], que também não identificou significância estatística comparando o aprendizado de programação entre o grupo que usou o Robomind e o grupo que utilizou uma abordagem tradicional, com o uso direto de uma linguagem de programação profissional. Por outro lado difere dos estudos [Yuana and Maryono 2016], [Nofitasari et al. 2017], [Schoeffel et al. 2015], [Mafulah et al. 2017] e [Durães 2015], que identificaram ganhos no aprendizado com o uso do Robomind.

Observa-se que, com exceção do trabalho [Mafulah et al. 2017], os demais apresentam relevantes ameaças à validade ou não adotam testes estatísticos, o que dificulta a comparação entre estudos. Sugere-se, portanto, que novos experimentos sejam realizados para que novas evidências sobre a efetividade dessa ferramenta no aprendizado de programação sejam identificadas.

Os resultados sobre o aprendizado surpreenderam os autores do estudo, pois acreditava-se que o feedback imediato, interatividade e o ambiente lúdico do Robomind seriam facilitadores do aprendizado.

No entanto, os ganhos de aprendizado não devem ser os únicos parâmetros que definem o uso de uma ferramenta educacional. Motivar e engajar estudantes também são elementos fundamentais e as ferramentas educacionais devem proporcioná-los. Nesse sentido, os resultados dos estudos [Yuana and Maryono 2016], [Schoeffel et al. 2015], [Benitti et al. 2009] e [Conchinha et al. 2016], apontam que o Robomind atende esta demanda. No entanto, critica-se nesses estudos a ausência de instrumentos validados para mensurar a motivação e engajamento dos estudantes.

5.2. b) o uso do Robomind reduz a evasão de estudantes em comparação a um ensino tradicional?

O grupo experimental apresentou uma menor evasão, em comparação ao grupo controle. Devido a significância estatística desse resultado é possível rejeitar a hipótese nula H_{0c} .

Esse resultado é bastante positivo, pois a ferramenta pode ter contribuído para a permanência dos estudantes até o final da disciplina, mesmo quando seus resultados

não eram favoráveis à aprovação. Acredita-se que essa persistência é benéfica por dois motivos: primeiro, porque pode favorecer a assimilação de parte do conteúdo que será útil ao cursar a disciplina novamente; segundo, pois pode reduzir um tipo mais grave de evasão que é o abandono do ensino médio.

5.3. Avaliação geral

Observou-se também que o uso da ferramenta divide opinião entre os estudantes. Alunos que apresentavam desempenho mais avançado em programação ficaram ansiosos para ter contato com linguagem de programação de uso profissional e de alguma maneira não se sentiam plenamente realizados em utilizar o Robomind. Por outro lado, parte dos estudantes consideraram a ferramenta divertida e desafiadora.

É natural que os estudantes apresentem diferentes visões sobre a ferramenta, em especial a depender de seu conhecimento prévio em programação e/ou progresso na disciplina. Isto precisa ser considerado pelos professores e uma alternativa pode ser adotar diferentes estratégias pedagógicas, como, por exemplo, utilizar o Robomind para alunos com dificuldades na compreensão dos conceitos de programação e o uso de uma linguagem de programação real com estudantes mais avançados nos conteúdos.

Por fim, sugere-se que trabalhos futuros também mensurem a capacidade do Robomind em promover o aprendizado de depuração de código, pois os recursos visuais dessa ferramenta podem ajudar os estudantes na compreensão sobre a execução de um algoritmo. Também é importante avaliar os resultados dos estudantes que receberão essa intervenção, em cada conceito de programação (e.g., condicionantes), como realizado em [Koorse et al. 2015]. Assim, é possível avaliar em quais assuntos pode haver uma maior contribuição da ferramenta.

6. Conclusão

O presente trabalho teve como objetivo avaliar a ferramenta Robomind por meio de um experimento com sessenta e cinco estudantes do curso técnico em informática integrado ao ensino médio do Instituto Federal de Pernambuco - campus Garanhuns.

Foi identificado que a ferramenta não promoveu ganhos de aprendizado aos estudantes que a utilizaram (grupo experimental), quando comparados aos estudantes que utilizaram uma metodologia tradicional de ensino (grupo controle). Os resultados não foram estatisticamente significantes. Por outro lado, os estudantes do grupo experimental apresentaram menor índice de evasão, com um resultado estatisticamente significativo.

As evidências científicas fornecidas neste estudo, somada às demais já publicadas na literatura, ajudam a subsidiar os docentes nas escolhas de ferramentas para apoiar o processo de ensino e aprendizagem. Observa-se que a efetividade do Robomind na promoção do aprendizado de programação não está fortemente sustentada, mas que em outros quesitos, como motivação, estímulo ao interesse pela área e capacidade de reduzir a evasão, a ferramenta apresenta grandes potenciais.

References

- Benitti, F. B. V., Vahldick, A., Urban, D. L., Krueger, M. L., and Halma, A. (2009). Experimentação com robótica educativa no ensino médio: ambiente, atividades e resultados. In *Anais do Workshop de Informática na Escola*, volume 1, pages 1811–1820.

- Cleophas, T. J. and Zwinderman, A. H. (2011). Non-parametric tests. In *Statistical Analysis of Clinical Data on a Pocket Calculator*, pages 9–13. Springer.
- Conchinha, C., Leal, M., and de Freitas, J. C. (2016). Robots & nee: A robótica virtual como promotora de inclusão e da aprendizagem por projetos lúdicos robots & nee: Virtual robotics as a promoter for inclusion and learning by playful projects. In *IV Conferência Ibérica de Inovação na Educação com TIC*.
- Durães, D. A. (2015). Gaming and robotics to transforming learning. In *Methodologies and Intelligent Systems for Technology Enhanced Learning*, pages 51–56. Springer.
- INEP - Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira (2017). *Censo escolar da educação básica em 2016*. http://download.inep.gov.br/educacao_basica/censo_escolar/notas_estatisticas/2017/notas_estatisticas_censo_escolar_da_educacao_basica_2016.pdf [Online; Acessado em 28/06/2018].
- Kirk, R. E. (1982). *Experimental design*. Wiley Online Library.
- Koorsse, M., Cilliers, C., and Calitz, A. (2015). Programming assistance tools to support the learning of it programming in south african secondary schools. *Computers & Education*, 82:162–178.
- Mafulah, I. A., Efendi, A., and Basori, B. (2017). The effectiveness of robomind software in basic programming learning. *Indonesian Journal of Informatics Education*, 1(1):30–37.
- McHugh, M. L. (2013). The chi-square test of independence. *Biochemia medica: Biochemia medica*, 23(2):143–149.
- Nofitasari, A., Yuana, R. A., and Maryono, D. (2017). The use of robomind application in problem based learning model to enhance the students understanding on the conceptual programming algorithm. *Indonesian Journal of Informatics Education*, 1(1):9–18.
- Pallant, J. (2007). *Spss survival manual: A step-by-step guide to data analysis with spss*. New York, NY: McGrath Hill. Google Scholar.
- Razali, N. M., Wah, Y. B., et al. (2011). Power comparisons of shapiro-wilk, kolmogorov-smirnov, lilliefors and anderson-darling tests. *Journal of statistical modeling and analytics*, 2(1):21–33.
- Schoeffel, P., Moser, P., Varela, G., Durigon, L., de Albuquerque, G. C., and Niquelatti, M. (2015). Uma experiência no ensino de pensamento computacional para alunos do ensino fundamental. In *Anais dos Workshops do Congresso Brasileiro de Informática na Educação*, volume 4, page 1474.
- Silva, L. S. (2018). Análise do aprendizado de programação por estudantes do ensino técnico integrado ao nível médio do instituto federal de pernambuco. In *V Encontro Nacional de Computação dos Institutos Federais (ENCompIF) do Congresso da Sociedade Brasileira de Computação (CSBC)*.
- Yuana, R. A. and Maryono, D. (2016). Robomind utilization to improve student motivation and concept in learning programming. In *Proceeding of International Conference on Teacher Training and Education*, volume 1.