

"Aperta o Play!" Análise da Interação Exploratória em um Jogo Baseado em Pensamento Computacional

Taciana Pontual Falcão¹, Rafael Santos Barbosa¹

¹Departamento de Estatística e Informática (DEINFO)

Universidade Federal Rural de Pernambuco (UFRPE)

Rua Dom Manoel de Medeiros, s/n, Dois Irmãos, CEP 52171-900, Recife – PE – Brasil

taciana@deinfo.ufrpe.br, s.rafaelbarbosa@gmail.com

Abstract. *The increasing importance given to the development of computational thinking in children is leading to the appearance of a number of digital games involving the logic of programming in exploratory environments. However, the lack of appropriate methods to determine the quality of these games hinders their adoption in schools. Through a case study, this paper presents formative and objective analyses of pedagogical and technical aspects that proved to be relevant in children's interaction with a game of the kind. These aspects can become parameters of a formal evaluation method specific for such games.*

Resumo. *A crescente valorização do desenvolvimento do pensamento computacional em crianças tem impulsionado o surgimento de jogos digitais que envolvem lógica de programação em ambientes exploratórios. Porém, a falta de métodos apropriados para avaliar a qualidade desses jogos dificulta a sua adoção em escolas. Através de um estudo de caso, este artigo apresenta análises formativa e objetiva de aspectos pedagógicos e técnicos que se mostraram relevantes no processo de interação de crianças com um jogo desse tipo. Tais aspectos têm o potencial de se tornarem parâmetros de um método de avaliação formal específico para jogos do gênero.*

1. Introdução

Existe uma crescente valorização da importância de se estimular o pensamento computacional no ensino básico, por se tratar de uma habilidade transversal a outras áreas do conhecimento, que vem se tornando essencial na sociedade contemporânea [Barr e Stephenson 2011][Wing 2006]. O pensamento computacional remete a estratégias para resolução de problemas através de raciocínio lógico e formal, em vários níveis de abstração, estendendo faculdades cognitivas [Wing 2006], o que foi apontado de forma visionária por Seymour Papert com a linguagem de programação Logo, dirigida a crianças [Papert 1980]. Entretanto, passadas décadas, grandes desafios permanecem em relação à exposição das crianças ao pensamento computacional, principalmente a maneira de integrá-lo à sala de aula e quais habilidades exigir dos alunos [Barr e Stephenson 2011]. Iniciativas precursoras incluem robótica educacional, computação desplugada, ferramentas visuais e jogos digitais envolvendo lógica de programação, sendo estes últimos o foco deste artigo [França et al. 2014][Gomes e Melo 2013].

Pesquisas apontam, de forma geral, o uso de jogos digitais como um poderoso instrumento no processo de ensino-aprendizagem [Felicia 2012]. Mais especificamente, jogos que estimulam o pensamento computacional envolvem o desenvolvimento do raciocínio através de resolução de tarefas com o auxílio de simulações passo-a-passo e são geralmente baseados em lógica de programação, sem necessariamente envolver uma linguagem formal [Medeiros et al. 2013]. Nesse processo, o jogador-aprendiz torna-se um agente protagonista na produção de conhecimento através de investigação e exploração [Felicia 2012]. Além disso, os jogos podem se tornar elementos motivadores por oferecerem momentos lúdicos e interativos no processo de aprendizagem, sendo atrativos ao aluno e aumentando assim a disposição para aprender [Medeiros et al. 2013]. Entretanto, jogos digitais ainda são muito pouco utilizados em contextos educacionais [Medeiros et al. 2013]. Dois fatores importantes constituem-se como obstáculos para a sua adoção: dificuldade dos professores em encontrarem jogos que se integrem ao seu planejamento didático; e desconhecimento sobre como avaliar a qualidade dos jogos [Medeiros e Schimiguel 2012]. De fato, apesar de uma quantidade razoável de propostas para avaliar software educacional, ainda não existem métodos consolidados específicos para tal, e menos ainda para jogos digitais educacionais [Brito Junior e Aguiar 2014].

Esse artigo tem como objetivo geral contribuir para a difusão no ensino básico de jogos digitais educacionais, particularmente com foco no desenvolvimento de pensamento computacional e alinhados ao paradigma da aprendizagem por descoberta através de interação exploratória. Para tanto, a pesquisa visa construir uma melhor compreensão acerca de como avaliar a qualidade desses jogos, analisando através de um estudo de caso a possibilidade de aprendizagem por descoberta através de interação exploratória com um jogo do gênero, e fazendo uma reflexão sobre as dificuldades dessa interação à luz de heurísticas de usabilidade de interfaces de sistemas digitais.

2. Interação Exploratória e Aprendizagem por Descoberta

A aprendizagem por descoberta baseia-se na ideia de que o aprendiz utiliza experiência e conhecimento prévio em um processo de exploração através de manipulação de representações externas [Bruner 1961]. Assim, os conceitos não são explicitamente passados ao aprendiz, pois é esperado que ele mesmo os descubra através de investigação autônoma com os artefatos providos. A aprendizagem por descoberta é baseada no construtivismo [Piaget e Inhelder 1969], segundo o qual o aprendiz deve ser um participante ativo no seu processo de aprendizagem, experimentando, descobrindo, errando e depurando, e assim construindo um conhecimento mais robusto e significativo [Bruner 1961]. Porém, apesar das raízes construtivistas e de uma visão pedagógica moderna, os benefícios da aprendizagem por descoberta em relação ao método de instrução direta ainda não foram claramente demonstrados [Mayer 2004]. A aprendizagem por descoberta é criticada por deixar os aprendizes sujeitos a interpretações errôneas, sobrecarregar seu espaço de trabalho cognitivo e deixar lacunas no conhecimento que deveriam adquirir [Alfieri et al. 2011]. Assim, deixar os aprendizes totalmente desassistidos pode não ser eficiente para o surgimento de oportunidades de aprendizagem [Mayer 2004]. Buscando minimizar estes problemas, surgiram os métodos de descoberta guiada, que propõem a integração de técnicas de *feedback* e *scaffolding* para melhor estruturar a atividade do aprendiz, e assim prover,

por um lado, liberdade suficiente para que ele seja cognitivamente ativo e, por outro, orientação suficiente para que ele construa conhecimento significativo [Mayer 2004].

Feedback e *scaffolding* são, em geral, providos pela mediação tanto do educador quanto dos artefatos. As tecnologias digitais são ferramentas cujo papel mediador vem se ampliando em todos os contextos da atividade humana contemporânea, e com especial importância em contextos educacionais, onde elas introduzem formas de pensamento diferentes do raciocínio linear, baseadas em simulação e experimentação, e envolvendo uma nova linguagem de comunicação instantânea [Borba e Villareal 2005]. Além disso, elas fornecem meios de se explicitar ligações entre representações concretas e abstratas, ajudando a externalizar ideias e processos, através de *feedback* dinâmico e interativo, rastreamento de ações e possibilidade de desfazê-las [Clements 1999]. Tais características favorecem a interação exploratória, com o potencial de promover uma aprendizagem por descoberta guiada, conforme recomendações da literatura [Mayer 2004]. Porém, as tecnologias digitais trazem novos aspectos a serem considerados, em particular em relação ao *feedback* ao aprendiz e às metáforas da interface. Esse artigo analisa esses e outros aspectos através de um estudo de caso com um jogo digital educacional exploratório envolvendo pensamento computacional.

3. Jogos Digitais e Pensamento Computacional

Apesar de sua integração aos processos de aprendizagem formal ainda ser polêmica nas instituições de ensino, jogos são apontados como ferramentas interessantes e inovadoras para a educação [Medeiros e Schimiguel 2012]. Além dos benefícios da motivação trazida pela ludicidade [Medeiros et al. 2013], jogos desafiam o aluno a tomar decisões com base em raciocínio lógico, definição de estratégias, e sistematização e avaliação de informações, desenvolvendo assim competências cognitivas ligadas à resolução de problemas [Felicita 2012]. Esse processo é facilitado pela interatividade e rápido *feedback* dos jogos, promovendo ciclos de reflexão, o que por si só já é um estímulo ao desenvolvimento de pensamento computacional baseado em interação exploratória. Ademais, segundo Johnson (2005), jogadores pensam de modo similar a programadores ao escreverem códigos com instruções aninhadas: pode-se dizer que a lógica algorítmica está presente implicitamente nos jogos. Ao se trabalhar pensamento computacional através de noções de programação e pensamento algorítmico, são desenvolvidos conceitos como abstração, modularização e recursão, que podem ser aplicados a outras áreas como técnicas de resolução de problemas [Nunes 2011].

Muitas são as dificuldades em processos de aprendizagem de lógica de programação [Rapkiewicz et al. 2006], algumas das quais amenizadas por ferramentas de programação visual e jogos. Interfaces gráficas de fácil uso possibilitam aos estudantes explorarem conceitos básicos de programação e algoritmos de modo mais intuitivo, sem se aterem a detalhes de sintaxe, concentrando-se na concepção da solução dos problemas [Medeiros et al. 2013]. Jogos como *Hour of Code* (hourofcode.com) (Figura 1, à esquerda), *The Foos* (www.thefoos.com) (Figura 1, à direita) e *Lightbot* (lightbot.com) (Figura 2) usam uma linguagem de programação visual em que o código é "escrito" arrastando e soltando blocos de comandos (sejam eles escritos em pseudo-código ou puramente visuais). Eles proveem um ambiente de interação exploratória em que o aprendiz pode experimentar diversas sequências de comandos e obter *feedback* através de uma representação visual dinâmica, podendo comparar os resultados obtidos com a solução pensada, e estabelecendo ligações entre a lógica abstrata de programação

e as consequências concretas que ocorrem no jogo [Santiago e Dazzi 2003]. Jogos desse gênero claramente têm potencial para promover a aprendizagem do pensamento computacional através de interação exploratória. Ainda assim, faz-se necessária uma maneira de avaliar esse potencial e verificar formalmente a adequação desses artefatos ao propósito em questão.



Figura 1. Jogos para ensino de pensamento computacional: Hora do Código (à esquerda) e The Foos (à direita)

4. Estudo de Caso: Avaliação do Jogo Lightbot

Existem várias propostas na literatura que tentam chegar a uma forma adequada para se avaliar um software educacional, sem no entanto apontarem para um consenso. De forma geral, sugere-se que é preciso considerar características técnicas (usabilidade, qualidade e ergonomia de software) e pedagógicas [Brito Junior e Aguiar 2014]. Uma das grandes dificuldades em se avaliar um software educacional refere-se ao fato da sua adequação depender da forma como eles são inseridos nas práticas pedagógicas [Brito Junior e Aguiar 2014]. Nesse sentido, Oliveira (2001) propõe uma distinção entre abordagens objetivas e formativas, que se complementam. Na abordagem objetiva, um especialista avalia o software verificando um conjunto de critérios pré-definidos, enquanto a abordagem formativa investiga a interação entre alunos, professores e ferramenta através de observação de atividades e/ou entrevistas com os mesmos. No estudo aqui apresentado, o *Lightbot* foi avaliado de maneira formativa com crianças, e de maneira objetiva através de um conjunto consolidado de heurísticas de usabilidade de software.

4.1. O Lightbot

O jogo *Lightbot* tem o intuito de apresentar ao jogador diferentes desafios de lógica de programação com um objetivo simples: movimentar um robô (utilizando os ícones de comandos) até o ladrilho azul e acender a lâmpada ali presente, deixando assim o ladrilho amarelo. As fases do jogo começam com poucos comandos, para que o jogador entenda cada funcionalidade. À medida que o jogador vai passando os estágios, os desafios vão aumentando em dificuldade, entrando em questão as propostas de aprendizado de controle de fluxo, funções, laços e maiores desafios de lógica com uma quantidade limitada de comandos que o jogador possui para resolver o problema. O desenvolvimento do "código" é sempre feito arrastando e soltando os blocos de comandos para que o robô se movimente. Assim, os jogadores podem ganhar prática e entendimento dos conceitos básicos de controle de fluxo, testando iterativamente

sequências de comandos e verificando as consequências na movimentação do robô. O *Lightbot* foi escolhido por ser um jogo de desafios, com mecânica baseada em conceitos de programação e portanto envolvendo pensamento computacional, com uma interface simples e objetiva com potencial para proporcionar uma interação exploratória.

4.2. Metodologia

A avaliação do *Lightbot* foi feita em duas etapas: formativa e objetiva. A avaliação formativa com crianças teve como objetivo geral analisar as possibilidades de aprendizagem por descoberta através de interação exploratória. Onze crianças (5 meninos e 6 meninas) foram recrutadas em uma praça pública da cidade, próxima à universidade e a um parque frequentado por famílias. Os pesquisadores abordaram as famílias de passagem para o parque, identificando-se aos responsáveis pelas crianças e explicando a atividade e seu objetivo. Quando dado consentimento, os pesquisadores sentaram-se com a criança em um banco da praça, usando os aparelhos de celular dos pesquisadores e o aplicativo *Lightbot*. A idade dos participantes concentrou-se entre 6 e 9 anos (com exceção de um menino de 5 anos e um de 13). Todos tinham familiaridade com jogos de celular. Os pesquisadores trabalharam em duplas, sendo um responsável pela condução da atividade e o outro pelos registros. Às crianças, foi explicado que o jogo estava sendo testado com a ajuda delas para identificar melhorias. Foi esclarecido que elas não deveriam preocupar-se em errar ou acertar, mas sentirem-se livres para explorar o jogo. O pesquisador forneceu instruções apenas quando solicitado pela criança, tendo como prioridade incentivá-la a explorar o jogo sozinha. Foram explorados os níveis 1, 2 e 3 do jogo (Figura 2). As sessões tiveram um tempo variável, entre 15 e 30 minutos, a depender do interesse da criança. Ao final, perguntou-se a opinião das crianças sobre o jogo, e foram entregues certificados de participação como “avaliadores de jogos digitais” em sinal de agradecimento.

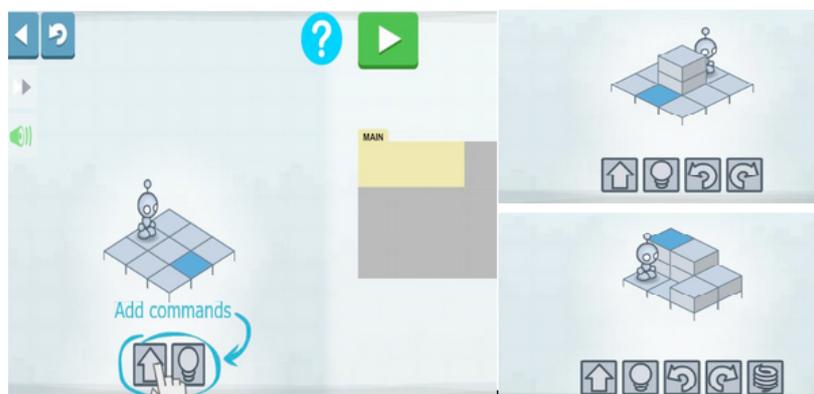


Figura 2. Estágios testados do *Lightbot*

A avaliação objetiva do *Lightbot* foi feita com base em heurísticas de usabilidade de software. A avaliação heurística é uma das formas mais tradicionais de avaliação de interfaces de software [Cybis et al. 2010], feita com base nas heurísticas propostas e refinadas por Jakob Nielsen e colaboradores [Nielsen e Mohlic 1990]. No entanto, o contexto de software educacional possui características próprias, formando uma categoria diferenciada à qual não necessariamente as heurísticas se aplicam. Por outro lado, apesar de uma quantidade razoável de estudos e propostas [Brito Junior e Aguiar 2014], ainda não existem métodos consolidados específicos para se avaliar um software educacional, e menos ainda um jogo digital educacional. Assim, partindo-se da

premissa de que as heurísticas de Nielsen são formalmente consideradas como um método efetivo de avaliação de interfaces, foi feita uma análise das categorias geradas pela avaliação formativa transpondo as heurísticas de Nielsen para um contexto de software educacional. A análise objetiva, portanto, foi complementar à formativa.

4.3. Resultados

Na avaliação formativa, a maioria das crianças afirmou ter gostado do jogo. O ambiente ao ar livre e com barulho moderado e distrações não impactou na concentração das crianças. Em alguns casos percebeu-se que, por timidez, as crianças declararam terem gostado apesar das claras dificuldades de interação. A maioria jogaria novamente, "tentaria com mais calma para passar os estágios". Foi percebida uma maior afinidade dos meninos pelo jogo, demonstrando mais empolgação, interesse, e melhor desempenho do que as meninas. A interface do jogo foi considerada pelos pesquisadores como predominantemente masculina, o que pode justificar esse fato. Na maioria dos casos os pesquisadores precisaram interromper a atividade, informando à criança que ela precisava dar a vez a outra. Alguns responsáveis demonstraram interesse em incentivar as crianças a usar o jogo, por ser educativo e não ser violento. O fato dos responsáveis acompanharem as sessões causou algumas intervenções que geraram pressão e expectativa nas crianças. As intervenções incluíram: explicações sobre o objetivo do jogo; instruções ("leia as coisas, V., leia os quadradinhos!"); e até mesmo ordens ("o último, depois do último, G., é aqui, pelo amor de Deus. Aperta o play!!").

Além dessas impressões gerais, os registros feitos pelos pesquisadores foram analisados qualitativamente através de análise de conteúdo, o que gerou três categorias: interação exploratória; metáforas visuais da interface; e design de interação. Essas categorias são apresentadas a seguir e analisadas à luz das heurísticas de Nielsen relacionadas (indicadas através do código 'H', adaptadas e traduzidas livremente). Três das heurísticas não foram utilizadas na análise (consistência, flexibilidade e eficiência de uso, e design minimalista).

4.3.1. Interação Exploratória: Livre versus Guiada

Apesar das tentativas para descobrir como funcionava o jogo, foi percebida uma certa confusão das crianças ao tentar jogar, que as levava a pedir ajuda ao pesquisador. Mesmo acessando o tutorial, as crianças não compreenderam as instruções escritas. Na maioria dos casos, foi preciso auxiliá-las com dicas ao longo da interação. Assim, a frequência de intervenção dos pesquisadores foi alta por conta das solicitações de ajuda e das dificuldades das crianças. As heurísticas a seguir apontam algumas questões que precisam ser consideradas para promover uma maior autonomia das crianças.

[H] *Ajuda e documentação: se necessária, a ajuda deve ser facilmente encontrada e consultada, sendo objetiva e simples.* Apesar de necessária, a ajuda em jogos educacionais é um desafio para os projetistas, visto que as crianças têm pouca paciência para estudar como usar o jogo, e quando tentam acompanhar tutoriais, nem sempre conseguem compreender as instruções, como foi observado neste trabalho. Idealmente, um software educacional deveria ser intuitivo o suficiente para que os usuários interagissem sem a necessidade de ajuda documentada.

[H] *Visibilidade do estado do sistema: o sistema deve manter o usuário informado através de feedback apropriado.* Na presente análise, essa heurística complementa a anterior. Em contextos educacionais, a concepção de "feedback apropriado" deve

contemplar processos de *scaffolding*, através dos quais as tecnologias tornam-se ferramentas mediadoras do processo de aprendizagem [Borba e Villareal 2005]. Um *scaffolding* eficaz no *Lightbot*, por exemplo, ajudaria as crianças a explorarem o jogo de forma mais autônoma, com menos necessidade de orientações externas, apoiando processos de aprendizagem por descoberta.

[H] *Erros: o sistema deve pedir confirmações das ações do usuário para prevenir ações indesejadas. No caso de erros, as mensagens devem informar a causa, as consequências e a solução.* As diretrizes relativas a erros se aplicam a software educacional tanto quanto a outros tipos de software. Porém, é fundamental expandir a reflexão sobre erros para incluir na avaliação de software educacional o erro conceitual. O tipo de feedback dado por um software educacional após um erro conceitual do usuário é definido pela orientação pedagógica desse software (essencialmente construtivista ou comportamentalista). Mas, de forma geral, o sistema pode, por exemplo, fornecer dicas após uma série de tentativas erradas do usuário. Nas sessões com o *Lightbot* as crianças faziam um giro completo com o robô ao descobrirem que estavam usando a rotação para o lado oposto, quando deveriam selecionar a rotação correta. Face a uma ação desse tipo, o jogo poderia sugerir à criança o uso da seta oposta, o que é uma forma de *scaffolding*.

[H] *Controle e liberdade do usuário: o sistema deve prover meios para o usuário realizar as ações que deseja com possibilidade de desfazê-las e refazê-las.* Essa heurística é de fundamental importância para a aprendizagem por descoberta, pois permite que o aprendiz explore a interface com segurança, sem temer o erro. Em termos pedagógicos, o desfazer-refazer é um dos grandes ganhos que as tecnologias interativas trouxeram ao processo de aprendizagem, permitindo que o aprendiz experimente diversas ações e interprete suas consequências.

4.3.2. Metáforas Visuais da Interface

Foi necessária explicação sobre os botões de ações (Figura 2), que as crianças não compreendiam com o tutorial apenas. Em particular, associar as setas rotatórias ao giro à esquerda ou à direita foi uma grande dificuldade. As crianças não conseguiam criar um mapeamento mental entre a noção de deslocamento do robô e a linguagem do tipo: "girar para o lado oposto"; "lado direito"; e "lado esquerdo". A linguagem compreendida pelas crianças era: "seta pra frente"; e "gira pra cá e pra lá" (mostrando com o dedo, visualmente, o lado para o qual o robô iria se dirigir).

[H] *Reconhecimento em vez de recordação: as informações e objetos relevantes devem estar visíveis para minimizar a carga de trabalho da memória do usuário.* No contexto aqui tratado é importante acrescentar a essa heurística que os elementos de interface além de identificáveis devem ser compreensíveis pelo usuário. Os comandos de setas rotatórias não foram compreendidos e conseqüentemente seu efeito não foi memorizado pelas crianças, que a cada nível do jogo precisaram redescobrir como usar as setas através de tentativa e erro, em vez de simplesmente reconhecer os objetos da interface e usá-los adequadamente.

4.3.3. Design da Interação

As crianças facilmente esqueciam como atingir completamente o objetivo do jogo: sabiam que precisavam chegar ao quadrado azul, mas não lembravam que tinham que inserir o comando para "acender a luz". Para elas, chegar ao ponto final já era suficiente.

A questão do salto também trouxe alguma dificuldade. Provavelmente devido à experiência com videogames, as crianças achavam que precisavam associar ao comando do salto o comando de movimentar-se para frente. Um dos pesquisadores forneceu ajuda através de associação com metáforas do mundo real: "para pular da calçada pra rua, você somente pula, não precisa pular e andar pra frente". Este tipo de metáfora facilitou o raciocínio das crianças. Ainda em relação ao deslocamento do robô, as crianças achavam que ele iria sempre continuar do ponto onde ele estava, e não retornar ao início. O retorno ao início foi um ponto de dificuldade na interação. As crianças acumulavam os comandos na caixa de controle, achando que o robô iria continuar a partir do último ponto, e não achavam intuitivo remover os controles.

[H] *Correspondência entre o sistema e mundo real: o sistema deve falar a linguagem do usuário e usar conceitos familiares e convenções do mundo real.* Em simulações como o *Lightbot*, propõe-se pensar em termos mais amplos de correspondência entre o mundo virtual, mundo real e contexto / perfil do aprendiz. No mundo virtual onde se desloca o robô *Lightbot*, por exemplo, o salto leva o robô para cima e para frente, o que não foi uma ação intuitiva devido ao funcionamento da maioria de jogos de videogames que as crianças conhecem, embora possa-se considerar que uma pessoa que salta no mundo físico geralmente também se desloca para frente. Isso implica em uma decisão importante do projetista, a ser pensada com cuidado. As crianças também não esperavam que o robô voltasse constantemente ao início de seu percurso (pois no mundo real as pessoas costumam continuar de onde pararam). Finalmente, o modelo mental das crianças considerava que o objetivo é cumprido chegando-se ao ponto final, sem a necessidade de uma ação desprovida de sentido metafórico dentro do contexto do jogo (acender uma lâmpada).

5. Considerações Finais

Neste artigo, foi apresentada uma avaliação de um jogo exploratório envolvendo pensamento computacional, através uma análise da interação de crianças com o jogo enriquecida por heurísticas de usabilidade. Essa avaliação permitiu identificar aspectos técnicos e pedagógicos relevantes em jogos educacionais do gênero.

Por proporcionar uma interação exploratória, jogos como *Lightbot* têm o potencial de promover a aprendizagem por descoberta, por sua vez apoiada em preceitos construtivistas e construcionistas [Piaget e Inhelder 1969] [Papert 1980]. Porém, a análise mostrou que há obstáculos no processo de aprendizagem por descoberta que ainda não foram superados mesmo com a mediação dos recursos inovadores inerentes às tecnologias digitais. Alguns desses obstáculos são dificuldades conhecidas na literatura sobre aprendizagem por descoberta, relacionadas principalmente à quantidade e qualidade do *feedback* necessárias para guiar os aprendizes. Neste artigo, mostrou-se que o ambiente de experimentação de comandos com visualização dos resultados (movimentação do robô) não foi suficiente para guiar as crianças, fazendo-se necessária a mediação de um facilitador. As razões específicas para tal precisam ser melhor investigadas, mas a análise revelou algumas delas à luz de heurísticas de usabilidade. Os principais obstáculos identificados foram: (i) a ajuda disponível (em forma de tutorial) mostrou-se pouco eficaz, enquanto que dicas durante a interação, baseadas no histórico de ações do usuário, poderiam constituir um *scaffolding* mais apropriado; (ii) o significado das representações visuais e metáforas associadas a elementos gráficos cruciais (setas rotatórias) e formas de movimentação do robô (salto e retorno ao início)

não foram internalizados pelas crianças, dificultando assim sua interação e compreensão conceitual; (iii) percebeu-se uma desconexão entre o contexto das crianças e o mundo simulado onde “vive” o robô (por exemplo, o objetivo de acender uma lâmpada - desprovido de sentido para as crianças - foi um dificultador).

As contribuições deste trabalho vão em duas direções. Uma é a identificação e compreensão de aspectos interacionais que devem ser levados em consideração no projeto e na avaliação de jogos digitais educacionais exploratórios, em particular envolvendo pensamento computacional. A outra contribuição refere-se à construção de um método de avaliação específico para jogos exploratórios envolvendo pensamento computacional. O método aqui utilizado aplicou heurísticas genéricas de usabilidade a dados empíricos. Embora essa abordagem tenha gerado resultados válidos, ela também apontou limitações dessas heurísticas para o contexto, e direções para criação de um conjunto de parâmetros específicos e/ou heurísticas adaptadas. Como trabalho futuro, pretende-se realizar outras avaliações com jogos similares, com o objetivo de gerar e consolidar critérios / heurísticas de avaliação através de um processo *bottom-up* (ou seja, a partir da utilização e análise de jogos do gênero), em oposição a métodos genéricos de avaliação de software educacional. Assim, a pesquisa pretende auxiliar a avaliação desses jogos por pesquisadores e professores, contribuindo para a sua difusão no ensino básico.

Agradecimentos

Este trabalho foi realizado como parte da disciplina de Informática na Educação 2013.2 do curso de Licenciatura em Computação da UFRPE. Agradecemos aos estudantes-pesquisadores Alcantara Rodrigues, Eugênio Maia, Fábio Menezes, Gustavo Barbosa, Igor Lucena, Pedro Silva e Verônica Santos, além de todas as crianças participantes.

Referências

- Alfieri, L., Brooks, P. J., Aldrich, N. J. and Tenenbaum, H. R. (2011) Does discovery-based instruction enhance learning? In: *Journal of Educational Psychology* 103(1), 1-18.
- Barr, V. and Stephenson, C. (2011) Bringing computational thinking to K-12: what is involved and what is the role of the computer science education community?, In: *ACM Inroads* 2(1), 48-54.
- Borba, M. C. and Villareal, M. E. (2005) "Humans-with-Media and the Reorganization of Mathematical Thinking", Springer, USA.
- Brito Junior, O. O. e Aguiar, Y. P. C. (2014) Análise de abordagens objetivas para avaliação de softwares educativos, Em: *Anais do 13º Simpósio Brasileiro de Fatores Humanos em Sistemas Computacionais - IHC'14*, Foz do Iguaçu, Brasil.
- Clements, D. H. (1999) 'Concrete' manipulatives, concrete ideas. In: *Contemporary Issues in Early Childhood* 1(1), 45-60.
- Cybis, W., Betiol, A. H. e Faust, R. (2010) "Ergonomia e Usabilidade - Conhecimentos, métodos e aplicações", Editora Novatec.
- Felicia, P. (2012) "Digital Games in schools: a handbook for teachers", European Schoolnet.

- França, R. S., Ferreira, V. A. S., Almeida, L. C. F. de e Amaral, H. J. C. do (2014) A disseminação do pensamento computacional na educação básica: lições aprendidas com experiências de licenciandos em computação, Em: *Anais do XXII Workshop sobre Educação em Computação - WEI*, Brasília, Brasil.
- Gomes, T. S. G. e Melo, J. C. B. de (2013) App Inventor for Android: Uma Nova Possibilidade para o Ensino de Lógica de Programação, Em: *Anais do Congresso Brasileiro de Informática na Educação - CBIE*, Campinas, Brasil.
- Johnson, S. (2005) "Surpreendente!: a televisão e o videogame nos tornam mais inteligentes", Elsevier, Rio de Janeiro.
- Mayer, R. E. (2004) Should There Be a Three-Strikes Rule Against Pure Discovery Learning? The Case for Guided Methods of Instruction, In: *American Psychologist* 59, 14-19.
- Medeiros, M. O. e Schimiguel, J. (2012) Uma Abordagem Para Avaliação De Jogos Educativos: Ênfase No Ensino Fundamental, Em: *Anais do 23º Simpósio Brasileiro de Informática na Educação - SBIE*, RJ, Brasil.
- Medeiros, T. J., da Silva, T. R. e Aranha, E. H. S. (2013) Ensino de programação utilizando jogos digitais: uma revisão sistemática da literatura, Em: *Revista Novas Tecnologias na Educação - RENTE* 11(3).
- Nielsen, J. and Mohlic, R. (1990) Heuristic evaluation of user interfaces, In: *Proceedings of ACM CHI'90 Conference*, Seattle, USA.
- Nunes, D. J. (2011) Ciência da Computação na Educação Básica, In *Jornal da Ciência*, 09/09.
- Oliveira, C. C. de, Costa, J. W. da e Moreira, M. (2001) "Ambientes informatizados de aprendizagem: produção e avaliação de software educativo", Papirus, Campinas.
- Papert, S. (1980) "Mindstorms. Children, computers and powerful ideas", Basic Books, NY.
- Piaget, J. and Inhelder, B. (1969) "The psychology of the child", Basic Books, NY.
- Rapkiewicz, C. E., Falkembach, G., Seixas, L., Rosa, N. dos S., Cunha, V. V. da e Klemann, M. (2006) Estratégias pedagógicas no ensino de algoritmos e programação associadas ao uso de jogos educacionais, Em: *Revista Novas Tecnologias na Educação - RENTE* 4(2).
- Santiago, R. e Dazzi, R. L. S. (2003) Ferramentas que auxiliam o desenvolvimento da lógica de programação, Em: *Anais do SEMINCO - Seminário de Computação*, 12, FURB, Blumenau.
- Wing, J. M. (2006) Computational thinking, *Communications of the ACM* 49(3), 33-35.