

O DESENVOLVIMENTO DO PENSAMENTO MATEMÁTICO COM O USO DO SOFTWARE DE PROGRAMAÇÃO ROBOMIND

Marcelo Souza Motta¹
Universidade Tecnológica Federal do Paraná- UTFPR
marcelomotta@utfpr.edu.br

Resumo:

A utilização de ambientes de aprendizagem que criam micromundos possibilita o desenvolvimento do pensamento, através do aperfeiçoamento dos processos mentais essenciais a atividade matemática. Nesse direcionamento o ambiente criado por *softwares* de programação, exerce um papel fundamental ao auxiliar o processo de ensino-aprendizagem e o raciocínio criativo, abrindo perspectivas de trabalho que valorizam a resolução de problemas e tornam as ideias matemáticas significativas. Este artigo tem por objetivo apresentar algumas possibilidades de uso do *software* RoboMind ao Ensino de Matemática e suas contribuições ao desenvolvimento no pensamento matemático avançado.

Palavras-chave: RoboMind; Ensino de matemática; Pensamento Matemático; Tecnologias Educacionais.

1. Introdução

Ao definir os objetivos do ensino de matemática para a Educação Básica, os Parâmetros Curriculares Nacionais (PCN's) destacam que o aluno deve “[...] valorizá-la como instrumental para compreender o seu dia-a-dia, vendo-a como área que estimula o interesse, curiosidade, investigação e o raciocínio lógico.” (BRASIL, 2001, p. 15)

Nessa forma de pensar a aprendizagem matemática, o aluno deve aprender a utilizar os procedimentos matemáticos, os instrumentos tecnológicos disponíveis, desenvolver o pensamento e conseqüentemente os processos matemáticos mentais.

Assim, quando o aluno é solicitado a utilizar suas definições para criar significado a um conteúdo, ele cria estruturas formais apoiada numa grande variedade de processos cognitivos. Essa interação possibilita o desenvolvimento do pensamento essencial ao processo

¹ Doutor em Ensino de Ciências e Matemática, professor Adjunto da Universidade Tecnológica Federal do Paraná (UTFPR) – Câmpus Curitiba e professor permanente do Mestrado em Ensino de Matemática (PPGMAT/UTFPR) – Câmpus Londrina.

de ensino e aprendizagem da matemática.

Existem diversas metodologias que podem ser utilizadas para o desenvolvimento do pensamento matemático, dentre elas se destaca a utilização das tecnologias da informação. A informática está a serviço do ensino e aprendizagem da matemática, pois proporciona ao aluno a criação de uma imagem diferente da disciplina, bem como o enriquecimento de práticas pedagógicas que desenvolvem a exploração, a criatividade, a ludicidade, o raciocínio lógico, a interatividade, a socialização, a afetividade e a reflexão crítica.

Nesse contexto, e examinando as várias possibilidades de uso de tecnologias no ensino de matemática se destacam os *softwares* de programação que criam micromundos. Esses ambientes fazem com que o aluno integre a sua experiência pessoal, os conceitos e procedimentos necessários ao aperfeiçoamento das ideias matemáticas.

Neste artigo foi analisado o *software* de programação RoboMind que propicia a construção de uma aprendizagem significativa, facilita o saber e contribui para a constituição das estruturas mentais. Com o RoboMind os alunos têm a oportunidade de acertar ou errar e, quando erra, pode investigar o motivo do erro, tendo a oportunidade de “fazer” e “refazer” suas atividades.

Dessa forma, o presente artigo propõe uma análise das contribuições do *software* de programação RoboMind no desenvolvimento do pensamento matemático e na criação de um ambiente de trabalho favorável a superação de lacunas existente na assimilação de alguns conceitos matemáticos.

2. Softwares Educacionais e o Ensino de Matemática

O uso de *softwares* educacionais auxilia os alunos na aquisição de conceitos, procedimentos e simulações, dando novo significado às atividades pedagógicas. Proporciona, também, ao docente a oportunidade de repensar sua prática de forma inovadora. Giraffa (1999) defende a concepção de que todo programa que utiliza uma metodologia que o contextualize no processo ensino e aprendizagem pode ser considerado educacional. Dessa forma, qualquer *software* pode ser usado na educação, dependendo exclusivamente do enfoque dado pelo educador.

Com a grande quantidade de *softwares* educacionais disponíveis, sua utilização oportuniza ao professor de qualquer área um “*fazer docente*” que cria situações enriquecedoras que contribuem com o raciocínio lógico, a autonomia, o levantamento de hipóteses, as inferências e, conseqüentemente, com o processo de ensino e aprendizagem.

Um *software* é desenvolvido para fins educacionais, quando em sua criação é definido o tipo de concepção pedagógica envolvida na sua construção. Para Ramos (1996), se um *software* é utilizado para fins educacionais, invariavelmente o mesmo (ou o uso que se faz dele) reflete um dos paradigmas educacionais: comportamentalista ou construtivista.

Os *softwares*, na concepção comportamentalista, baseiam-se na atuação passiva do aluno no processo de aprendizagem.

O aluno é direcionado a tomar algumas atitudes frente a estímulos apresentados, mas não há preocupação com o processo de raciocínio. São utilizados artifícios de reforço tais como notas e elogios e se em alguma atividade o aluno falhar, não há alternativa para que esse aluno possa refletir e reconstruir a sua resposta, já que a mensagem apenas notifica que o aluno falhou. (BONA, 2009, p. 2).

Já os *softwares* na base construtivista fundamentam-se na atuação interativa do aluno, frente a sua própria aprendizagem. Nessa concepção, “o conhecimento atual do aluno e as suas características para o aprendizado são levados em consideração.” (BONA, 2009, p.2).

Quanto à execução das atividades pelo aluno, um software pode ser algoritmo ou heurístico. Um software heurístico enfatiza aprendizagem pela descoberta, ou seja, a interação do educando com o programa cria um ambiente repleto de situações exploratórias. Já num *software* algoritmo, privilegia-se a transmissão do conhecimento, tendo como referência uma atividade planejada e estruturada.

Em relação ao direcionamento um *software* pode ter duas abordagens:

Na *abordagem dura* os planos são previamente traçados para uso do computador e as atividades dos alunos resumem-se a responder a perguntas apresentadas, registrando-se e contabilizando-se erros e acertos. Na *abordagem branda* a atividade e interação com o computador não parecem ter um objetivo definido, fazendo com que o aluno esteja no comando, fazendo uma série de atividades consideradas interessantes por ele, onde há desafio. Os erros são fontes de reflexão e desenvolvimento de novos projetos. (TEIXEIRA, 2004, p. 02, grifo nosso).

A partir da definição das abordagens, vários autores apresentam diferentes concepções para um *software* educacional. Para Valente (2000), os *softwares* educativos podem ser classificados de acordo com a maneira como o conhecimento é manipulado. Para ele, as

modalidades mais comuns destacadas pela maioria dos autores são: tutoriais, exercício e prática, jogos, simulação, programação, aplicativos, multimídia e ferramentas para resolução de problemas.

Portanto, a utilização de *softwares* educacionais auxilia no aperfeiçoamento da aprendizagem do aluno, criando um ambiente crítico no qual o educando tem a oportunidade de desenvolver a reflexão e o raciocínio lógico. A escolha do tipo de *software* depende exclusivamente do direcionamento proposto pelo professor.

Os *softwares* educativos matemáticos devem focar um trabalho na perspectiva construtiva, criando um ambiente no qual o aluno possa ampliar todos os processos mentais essenciais ao desenvolvimento de sua aprendizagem.

Nesses ambientes, podemos identificar duas formas de atividades, na direção de uma prática cognoscitiva:

Atividades de expressão: o aluno cria seus próprios modelos para expressão ideias e pensamentos. Suas concretizações mentais são exteriorizadas. [...] os ambientes são veículos de materialização de ideias pensamentos e mais geralmente de ações do sujeito;

Atividades de exploração: o aluno é apresentado a um modelo já pronto o qual deve ser explorado, entendido e analisado. [...] A própria compreensão do modelo, entendimento dos princípios de construção, já são por si só estímulos ao raciocínio, que favorecem a construções de relações e conceitos. (GRAVINA; SANTAROSA, 1999, p. 81, grifo nosso).

Portanto, os *softwares* educacionais matemáticos se apresentam como ferramentas de grande potencial frente aos obstáculos inerentes ao processo de ensino e aprendizagem da matemática, pois oferecem recursos que viabilizam as ações mentais e favorecendo um modelo pedagógico construtivista.

2.1 Softwares Educacionais para Programação

Desde 1970 vários estudos relatam que o ato de programar é um importante processo de desenvolvimento cognitivo. Wirth (1986) destaca que a programação é uma atividade construtiva e abrange um grande campo de variedades, envolvendo, em geral, atividades intelectuais bastante complexas.

No ensino de programação, tradicionalmente, o professor apresenta uma cadeia de comandos e códigos, esse método é de difícil compreensão, principalmente, para as séries

iniciais da Educação Básica. Para que programar seja uma atividade construtiva o docente deve buscar proporcionar um ambiente motivador, criativo e colaborativo.

Os softwares de programação podem ser categorizados em ferramentas de visualização, ferramentas de avaliação automática, ambientes de programação, ferramentas de suporte e programação e micromundos.

Uma ferramenta de programação pouco pesquisada no Brasil e que apresenta grandes contribuições ao Ensino de Matemática é o *software* RoboMind. O ambiente é uma linguagem de autoria que cria um micromundo específico e particular, em que se destaca a facilidade de interação e o desenvolvimento de potencialidades educativas necessárias ao processo cognitivo da matemática.

O micromundo criado pelo RoboMind proporciona ao estudante gerenciar sua aprendizagem, em um ambiente que é caracterizado por “objetos de pensar” e “objetos de pensar com”, proporcionando a explorações de concepções. (PAPERT, 1985, p. 25).

Rezende (2004, p. 53) destaca que a ideia de micromundo proposta por Papert deve possuir atividades autênticas e uma diversidade de objetos, que por meio de ações os aprendizes realizem construções de projetos concretos, privilegiando a flexibilidade de pensamento e a interpretação múltipla dos resultados

Nessa perspectiva, o RoboMind desenvolve um ambiente facilitador, em que o aluno é o sujeito ativo no desenvolvimento de sua aprendizagem através da mediação do professor. Essa interação ocorrerá por meio de manipulações com a linguagem de programação, dando novo significado a conhecimentos matemáticos.

3 Conhecendo o software de programação RoboMind

O RoboMind foi criado por uma parceria entre Jan Van Oorschot, Ernst Bovenkamp e Arvid Halma, é um software de código aberto (*open source*), que possibilita a modificação por qualquer usuário, a distribuição livre, a integridade do autor e a não discriminação de áreas ou grupos.

O *software* possui como metodologia a criação de um micromundo (mapas) com uma inteligência artificial (robô) a partir de uma linguagem de programação simples.

O ambiente é caracterizado por um mapa personalizável que permite ao usuário resolver problemas variados que podem ser verificados por meio da execução de um *script*, possibilitando a verificação de ideias e conceitos. Se existir algo errado, ele pode analisar o programa e identificar o erro, que é tratado como uma fase necessária à sua estruturação cognitiva.

Por se tratar de um software que torna o aluno sujeito ativo de sua aprendizagem, suas principais características são: *Amigabilidade*, pois, é de fácil uso e facilmente assimilado pelo aluno; *Modularidade* e *Extensibilidade* permitindo ao aluno incluir ou excluir comandos; *Interatividade* permite que o aluno veja e pense sobre seu erro imediatamente após a execução dos comandos; *Flexibilidade*, pode ser utilizados em todos os níveis de ensino (Fundamental, Médio e Superior); *Capacidade*, pois, permite ao aluno desenvolver uma linguagem de programação específica do software desenvolvendo o pensamento matemático.

A interface do programa é dividida em três partes: A janela de edição dos *scripts*, o mapa de movimentação do robô e o controle de execução, conforme destacado na Figura 01.



Figura 01 – Tela inicial do RoboMind

Neste artigo, utilizou-se a versão desenvolvida pelo Laboratório de Robótica (RobLab) do Departamento de Sistemas e Computação da Universidade Regional de Blumenau (FURB). A escolha por esta versão levou em consideração a disponibilização do *software* em português.

A inteligência artificial utilizada é o “robô” que executa um conjunto de instruções que permitem programá-lo para realizar diferentes tarefas. O robô possui as funções nativas que são: “pegar” que permite movimentar e retirar objetos do trajeto, “andar” descola o robô para frente ou para trás, “pintar” possibilita marcar um deslocamento realizado e a função “ver” que movimenta a parte superior do robô para a direita e esquerda.

Os comandos básicos para movimentação do robô são idênticos quando nos deslocamos, ou seja, anda-se para frente, para trás, para a direita ou para esquerda (ver Quadro 03).

Função	Comandos	Função	Comandos
Andar n passos para a frente	 andarFrente(n)	Anda n passos para trás	 andarTrás(n)
Vira 90 graus para a esquerda	 virarEsquerda()	Vira 90 graus para a direita	 virarDireita()
andarNorte(n)	Vira para o norte e anda n passos para frente	andarSul(n)	Vira para o sul e anda n passos para frente
andarLeste(n)	Vira para o leste e anda n passos para frente	andarOeste(n)	Vira para o oeste e anda n passos para frente
Pega o pincel e pinta o chão de branco	 pintarBranco()	Pega o pincel e pinta o chão de preto	 pintarPreto()
Para de pintar e esconde o pincel	 pararPintar()	Pega o objeto em frente do robô	 pegar()
Solta o objeto em frente ao robô	 soltar()	sortear()	Aleatoriamente determina um valor. Sortear() pode retornar verdadeiro ou falso.

Existem outras instruções avançadas das quais se destacam: repetições de instruções, repetições condicionadas, sair, voltar, pegar, soltar, condicional, expressões lógicas e operadores.

Uma das principais ferramentas do RoboMind consiste o desenvolvimento de projetos utilizando a linguagem de programação específica do *software* que se denomina *script*. A criação de um *script* permite a realização de um conjunto de

4 Contribuições do RoboMind ao pensamento matemático

O desenvolvimento de conceitos matemáticos com a possibilidade de utilização de um software de programação proporciona uma experiência matemática significativa criando situações que proporcionam a assimilação do conhecimento e a formação de habilidades. A possibilidade de realização de várias atividades, ordenadas de forma independente, cria momentos de reflexão, investigação e resolução de problemas.

Ponte (1991) destaca que a construção de programas exige do aluno um esforço suplementar de compreensão dos conceitos, muitas vezes obrigando a encará-los sob novas formas, e exige a elaboração de uma estratégia semelhante, em muitos aspectos, às que se usam para enfrentar situações problemáticas.

Nessa perspectiva de trabalho que valoriza a construção do conhecimento matemático pelo próprio aluno se destaca o RoboMind. O software apresenta um estilo próprio para abordar conceitos matemáticos dos quais se destacam: figuras planas, figuras espaciais, polígonos, ângulos, numeração, equações, coordenadas cartesianas, lógica, dentre outros.

O programa contribui no desenvolvimento do pensamento matemático, pois “[...] envolve todo ciclo de atividade matemática, que vai desde o ato criativo da consideração de um problema, passa pela elaboração de conjecturas e estratégias de resolução, até chegar ao refinamento e prova.” (TALL, 1991, p. 10).

Dentre os principais pensamentos matemáticos desenvolvidos no *software* RoboMind se destacam os processos mentais de representar, deduzir, conjecturar, generalizar, abstrair e visualizar. Destaca-se a seguir uma descrição desses processos matemáticos:

- O processo mental de *representação* se caracteriza pela construção das notações dos *scripts* expressando um conhecimento implícito sobre os conceitos matemáticos;
- A *dedução* fica evidenciada quando o aluno necessita empregar as propriedades de um determinado conceito matemático;
- As *conjecturas* surgem com frequência em ambientes tecnológicos, principalmente em *softwares* que possibilitam a aprendizagem por descoberta;

- A *generalização* ocorre quando o aluno manipula o *software* em uma situação particular e expande seu conhecimento para situações semelhantes;
- O processo de *abstração* é desenvolvido à medida que o aluno interage com as estruturas matemáticas presentes no RoboMind, isto é, relacionando o caminho desenvolvido pelo robô com as propriedades dos objetos matemáticos;
- A *formalização* acontece quando o aluno relaciona o conceito matemático, presente na interação do ambiente, com conteúdos usuais da matemática;
- Outro processo envolvido na interação do aluno com o ambiente é a *visualização* que permite a criação de representações a partir da compreensão dos mapas criados.

Para Dreyfus (2002) esses processos são complementares e não existe uma hierarquização entre eles, podendo cada um existir de forma independente do outro ou de forma concomitante. Existem outros processos matemáticos que não foram elencados que podem aparecer durante a manipulação do RoboMind.

Nesse contexto, pode-se afirmar que o *software* RoboMind, “contribui para o desenvolvimento dos processos mentais não somente como um instrumento, mas essencialmente, de maneira conceitual, influenciando o pensamento das pessoas.” (PAPERT, 1985, p. 114).

O RoboMind desenvolve também habilidades intelectuais e corporais, ajudando no desenvolvimento da localização espacial. Isto é perceptível quando o aluno executa alguns comandos básicos. Ele tem que se imaginar na posição do robô e ao mesmo tempo, descobrir quais comandos deve executar. Esta habilidade é denominada de “[...] sintonidade cultural, que relaciona a ideia de ângulo à navegação. Atividade positiva e firmemente enraizada à cultura extracurricular de muitas crianças.” (PAPERT, 1985, p. 87).

Dentre as formas geométricas criadas com o RoboMind se destacam os polígonos regulares. Na construção dessas figuras os alunos identificam as características de cada figura e os ângulos para traçá-las. Durante a manipulação com o software são desenvolvidos conceitos de ângulos internos e externos e as relações estabelecidas entre o perímetro e a área.

Imagine um plano cartesiano que passa exatamente pelo local onde está parado o robô.

Essa posição consideraremos como sendo a origem $(0, 0)$ do plano cartesiano.

Crie um script que faça o robô se deslocar pelas as coordenadas $(1, -1)$, $(1,3)$, $(-4, 0)$ e retornar para a origem do plano cartesiano.

Nesse processo de construção de polígonos regulares, os alunos utilizam seus esquemas de ação, aplicando os conceitos geométricos existentes na sua estrutura cognitiva (ângulos internos, externos e suplementares, soma dos ângulos internos e externos, área e perímetro) para a obtenção da figura desejada.

Outra possibilidade de uso do *software* no ensino de matemática é no aprimoramento de conceitos algébricos. Com a criação de *scripts* é possível utilizar argumentos que representem valores desconhecidos, na interação com a ferramenta o aluno precisará definir esses parâmetros para criar seu mapa ou estabelecer o caminho a ser percorrido pelo robô.

Existem propostas de trabalho com o Robomind que utilizam conceitos de localização no plano cartesiano. Uma proposta estabelece os eixos x e y no mapa de movimentação e considerada cada vértice da malha quadriculada como sendo uma coordenada.

Apresenta-se a seguir uma proposta de exercício de localização no plano cartesiano.



Uma das maiores contribuições do *software* ao Ensino de Matemática é o desenvolvimento do raciocínio lógico do aluno, possibilitando que as atividades produzidas sejam estabelecidas por meio de uma linguagem de programação.

Durante a programação do computador, os alunos constroem um “diálogo com o *software*”, o educando digita um comando e o computador fornece um *feedback*. Com base nesse retorno é que o aluno oferece um novo comando ou reformula o comando anterior.

Na execução de uma atividade com o RoboMind, a resposta não é fornecida, o aluno reflete sobre os erros apresentados, analisando todas as etapas desde o início, num processo de compreensão e depuração. A maneira como ele vê o erro nessa perspectiva de trabalho proporciona uma melhor compreensão da situação e dos conceitos envolvidos, identifica o seu estilo de pensar e de se relacionar com o mundo, ou seja, esse processo proporciona a produção do conhecimento.

5. Considerações Finais

Torna-se oportuno registrar que a utilização do *software* de programação RoboMind tem sido relatada em pesquisas sobre robótica educacional dentre as quais se destacam os trabalhos desenvolvidos pelo RobLab. Apesar da realização dessas investigações pouco se tem registrado sobre aplicações do ambiente no Ensino de Matemática, principalmente, no desenvolvimento do pensamento matemático.

O objetivo da presente pesquisa foi apresentar algumas contribuições do *software* no aprimoramento de conceitos matemáticos, identificando as principais contribuições ao desenvolvimento dos processos mentais.

Apesar das potencialidades do ambiente, vale ressaltar que não basta apenas repensar a aprendizagem e a informática. Faz-se necessário que se busque o conhecimento sobre o que se pretende no ato de ensinar, explicitando objetivos claros e não somente trabalhando o “*software pelo software*”.

O professor ao utilizar qualquer ferramenta tecnológica não deve tentar impor sua opinião e metodologia, mas proporcionar que o uso do computador seja um momento de reflexão e discussão da aprendizagem.

Nesse contexto, o aluno ao interagir com o RoboMind aprimora seus conhecimentos matemáticos, desenvolve seu pensamento reflexivo e cria um ambiente potencialmente favorável a ampliação dos processos mentais. Kawasaki (1981, p. 48) destaca que a atividade humana mediada pelo computador altera de forma qualitativa a estrutura da atividade intelectual possibilitando a reorganização da memória e as formas que armazenamos as informações.

6. Referências

BONA, B. O. **Análise de softwares educativos para o ensino de matemática nos anos iniciais do ensino fundamental**. Experiências em Ensino de Ciências. Rio Grande do Sul, 2009.

BRASIL. Ministério da Educação e do Desporto. Secretaria de Educação Fundamental. **Parâmetros Curriculares Nacionais**: Terceiro e Quarto Ciclos do Ensino Fundamental: Matemática. Brasília, 2001.

DREYFUS, T. Advanced Mathematical Thinking Processes. In: TALL, David. **Advanced Mathematical Thinking**. Holanda: Kluwer Academic Publishers, 2002.

- GIRAFFA, L. M. M. **Uma arquitetura de tutor utilizando estados mentais**. Tese (Doutorado em Educação). Universidade Federal do Rio Grande do Sul, 1999, Porto Alegre, 1999.
- GRAVINA, M. A.; SANTAROSA, L. C. M. **A Aprendizagem da Matemática em Ambientes Informatizados**. Informática na Educação: teoria & prática. Rio Grande do Sul, 1999.
- KAWASAKI, T. F. **Tecnologias na sala de aula de matemática: resistência e mudanças na formação continuada de professores**. 2008. Tese (Doutorado em Educação). Universidade Federal de Minas Gerais, 2008, Belo Horizonte, MG.
- PAPERT, S. **Logo: computadores e educação**. São Paulo: Brasiliense, 1985.
- PONTE, J. P. **O computador na educação Matemática** (Cadernos de Educação Matemática, Nº 2). Lisboa: APM, 1991.
- RAMOS, E. M. F. **Análise ergonômica do sistema HiperNet buscando aprendizado da cooperação e da autonomia**. 1996. 227f. Tese (Doutorado em Educação) – Universidade Federal de Santa Catarina, 1996, Florianópolis, SC.
- REZENDE, F. A. **Construção de um traçador gráfico para fins educacionais**. 2004. 185 f. Dissertação (Mestrado em Multimeios do Instituto de Artes) – Universidade Estadual de Campinas, 2004, Campinas – SP.
- TALL, D. O. (1991). **The Psychology of Advanced Mathematical Thinking**. In Advanced Mathematical Thinking, volume 11, p. 03-21. Kluwer Academic Publishers
- TEIXEIRA, J. F. **Uma discussão sobre a classificação de software educacional**. Campinas, Revista Unicamp, Informação & Tecnologia. 2004. Disponível em: <
<http://www.ccuec.unicamp.br/revista/infotec/artigos/jacqueline.html> > Acesso em: 01 de março de 2016.
- VALENTE, J.A. **O Professor no Ambiente Logo: formação e atuação**. Campinas: Graf. da UNICAMP, 2000.
- WIRTH, N. **Algoritmos e Estruturas de Dados**. Prentice Hall do Brasil, 1986.