

## O Modelo Cognitivo do Sistema Tutor Inteligente PAT2Math

**Henrique Seffrin, Geiseane Rubi, Patrícia Jaques**

Programa Interdisciplinar de Pós-Graduação em Computação Aplicada (PIPICA)

Universidade do Vale do Rio dos Sinos (UNISINOS) – São Leopoldo– RS – Brasil

hseffrin@terra.com.br, geiserubi@gmail.com, pjaques@unisinors.br

**Resumo:** *A fim de prover um acompanhamento individualizado aos alunos em sua aprendizagem, a arquitetura de um Sistema Tutor Inteligente (STI) é complexa. Um de seus componentes de maior importância é o Modelo Cognitivo. Este componente é responsável por corrigir todos os exercícios realizados pelo aluno, passo a passo, a fim de identificar habilidades desenvolvidas pelos alunos, assim como suas falsas concepções. Esse artigo apresenta o Modelo Cognitivo do STI PAT2Math, voltado ao ensino de equações algébricas de 1º e 2º grau. Também são apresentadas as avaliações qualitativas e quantitativas as quais este foi submetido.*

**Abstract:** *In order to provide an individual monitoring for students in their learning, the architecture of an Intelligent Tutoring System (ITS) is complex. One of its most important component is the Cognitive Model. This component is responsible for correcting all the exercises completed by the students, step-by-step, aiming at identifying the students' skills, and their common misconceptions. This paper shows the Cognitive Model of the Intelligent Tutoring System PAT2Math, focused on the learning of 1<sup>st</sup> and 2<sup>nd</sup> degree equations. This paper also describes the quantitative and qualitative evaluations for which PAT2Math was submitted.*

### 1. Introdução

A matemática é hoje considerada por muitos alunos uma das mais difíceis disciplinas. Dentre os conteúdos abordados, destaca-se a álgebra por introduzir os conceitos de incógnita e equivalência, que, até então, eram desconhecidos pelos alunos [Stacey, MacGregor, 1999]. Essa dificuldade também é percebida no contexto brasileiro: “Nos resultados do Sistema Nacional de Avaliação da Educação Básica (SAEB), por exemplo, os itens referentes à álgebra raramente atingem um índice de 40% de acerto em muitas regiões do país” [Brasil, 2001, p.115-116].

Gravina e Santarosa (1998, p. 8) afirmam que “Os ambientes informatizados apresentam-se como ferramentas de grande potencial frente aos obstáculos inerentes ao processo de aprendizagem”. Da mesma forma, Bloom (1984) e Borba (1999) reiteram os benefícios sobre o uso de ferramentas computacionais para o ensino, tendo em vista que é próximo ao ensino individualizado, em que há um professor para cada aluno, proporcionando um melhor aprendizado, pois o estudo está focado exclusivamente nas necessidades individuais de cada aluno. Os Sistemas Tutores Inteligentes (STI) são ambientes inteligentes de aprendizagem que buscam desenvolver esse ensino individualizado. Para tanto, a arquitetura desses sistemas é complexa, envolvendo a interação entre diversos componentes que buscam modelar informações sobre o conhecimento e perfil do aluno, decidir a melhor estratégia pedagógica a aplicar, entre outros. Um componente de maior importância em um STI é o Modelo Cognitivo. Ele

representa um especialista no conteúdo ensinado, sabendo resolver e corrigir qualquer exercício no tema ensinado pelo tutor. O desenvolvimento deste módulo pode ser um desafio em determinados domínios do conhecimento, como a álgebra, onde existem diversos caminhos para a resolução de uma equação.

Este artigo descreve o Módulo Cognitivo de um STI para álgebra, chamado de PAT2Math (*Personal Affective Tutor to Math*). PAT2Math é um STI com aplicação no ensino de álgebra, mais especificamente, voltado ao processo de ensino de equações algébricas de 1º e 2º grau com uma incógnita. Atualmente esse projeto, em andamento desde maio de 2008, possui dois aplicativos implementados, sendo um deles um protótipo funcional do módulo de resolução de equações do sistema tutor, chamado de PATEquation. O outro é uma ferramenta para demonstração de resolução de equações, denominado de PATSolver. Ambos estão funcionais e disponíveis para *download* em (<http://www.projeto.unisinos.br/pat2math/>). Outros artigos já foram publicados sobre PAT2Math. [Seffrin et al., 2009] descreve um Objeto de Aprendizagem (OA) que exibe a resolução de equações. [Seffrin et al., 2010] descreve o funcionamento geral do OA PATEquation. O presente artigo se difere dos anteriores por descrever especificamente o Modelo Cognitivo do tutor, bem como o módulo de detecção de falsas concepções (que integra o primeiro), implementado recentemente no sistema.

Este artigo está organizado como segue: as seções 2 e 3 apresentam o referencial teórico deste artigo: STI e Sistemas especialistas e Obstáculos Epistemológicos, respectivamente. A Seção 4 apresenta o trabalho proposto, detalhando cada componente separadamente. A seção 5 relaciona este trabalho com outros projetos. A seção 6 finaliza este artigo descrevendo as avaliações que foram realizadas ao longo deste projeto.

## 2. Sistemas Tutores Inteligentes

Sistema Tutor Inteligente (STI) é um ambiente computacional de aprendizagem que possui modelos de conteúdo instrucional que especificam o “que” ensinar e estratégias de ensino que especificam “como” ensinar [Wenger, 1987]. Esses sistemas são capazes de acompanhar o aluno durante todo o processo de solução de um problema. Desta forma, enquanto o aluno aprende o conteúdo, o tutor aprende sobre o aluno levantando diversas informações sobre o mesmo, como, por exemplo, facilidades que ele apresenta sobre determinado assunto e dificuldades apresentadas quanto a outros. Com este aprendizado, o tutor pode prover um acompanhamento mais individualizado, apresentando explicações e exercícios mais bem relacionados às habilidades que o aluno não se sente tão seguro ou possui maiores dificuldades.

Geralmente, a arquitetura de um sistema tutor é formada pelos seguintes componentes (Psołka et al., 1988; Woolf, 2009): (i) Base de Domínio (guarda o conhecimento procedural e/ou declarativo que representa o expertise do tutor); (ii) Modelo do Aluno (mantém informações sobre o conhecimento que o aluno possui); (iii) Tutor (aplica um determinado conteúdo de acordo com a estratégia selecionada e as informações no modelo de aluno), e (iv) Interface com o Usuário (interface gráfica com o usuário).

Em alguns tutores, a base de domínio é formada por 2 componentes principais: (i) o Modelo Cognitivo e o (ii) componente declarativo. O **componente declarativo** contém explanações, enunciados de exercícios, exemplos e outros tipos de conhecimento declarativo. O **Modelo Cognitivo** é um componente de extrema importância em STIs, pois ele é responsável por demonstrar e corrigir a resolução de qualquer exercício proposto ao aluno. Dessa forma, esse componente deve poder resolver qualquer exercício apresentado ao aluno. Em alguns domínios em que há diversas formas de solucionar um exercício, como na álgebra, onde há variadas formas de resolver uma equação, a modelagem desse módulo se torna ainda mais complexa. É

importante prever os diversos caminhos de solução que os alunos podem ter pensado. Para tanto o Módulo Cognitivo do STI PAT2Math foi implementado como um Sistema Especialista (SE), baseado em regras, do domínio ensinado: álgebra.

O funcionamento de um SE baseia-se na análise de certas informações; a partir destas, ele toma uma decisão ou realiza alguma conclusão. Basicamente, o SE trabalha com dois tipos de informações: os fatos e as regras. Os **fatos** são informações concretas, e certas que o sistema possui. Por exemplo, o paciente X apresenta marcas na pele, poderia ser um fato presente em um SE voltado para identificação de sintomas de doenças. As **regras** representam o conhecimento do sistema que será utilizado para analisar os fatos existentes e tirar uma conclusão, que poderá resultar na criação/remoção/atualização de alguns fatos existentes. As regras possuem o formato “se <condição> então <ação>”, como, por exemplo, “se o paciente apresenta marcas na pele, então é alergia”.

### 3. Obstáculos Epistemológicos na Aprendizagem de Álgebra

Na área de Educação, obstáculos epistemológicos [Brousseau, 1983; Silva et al., 2007] ou *misconceptions* são erros e falsas concepções que os alunos apresentam, frequentemente, ao estudarem um determinado conteúdo. Essas falsas concepções apresentadas pelo aluno impedem-no de aprender significativamente o conteúdo em questão.

Como os obstáculos epistemológicos são as dificuldades apresentadas pelos alunos na evolução da construção do conhecimento é importante salientar que o aluno possui conhecimentos prévios construídos ao longo da vida cotidiana. No entanto, não se trata de adquirir um conhecimento novo, mas sim de derrubar essas barreiras que estão impedindo a aprendizagem significativa para que o aluno possa evoluir cognitivamente. Conforme aponta os Parâmetros Curriculares Nacionais, “Os obstáculos apontados explicam em grande parte o desempenho insatisfatório dos alunos revelado pelas elevadas taxas de retenção em Matemática o que faz atuar como filtro social no Ensino Fundamental” [Brasil, 2001, p.23].

Baseado nos estudos de vários autores [Coxford and Shulte, 1995; Lima, 2007; Silva et al., 2007, entre outros], além de entrevistas realizadas com cinco professores do Ensino Fundamental, foram levantados os erros mais frequentes apresentados pelos alunos no processo de ensino e aprendizagem algébrica. São eles:

- **Primárias:** São seis tipos que envolvem operações primárias (adição, subtração, multiplicação e divisão) relacionadas a números e incógnitas.
- **Secundárias:** São cinco tipos: Operações com Frações, Propriedade Distributiva, Fator Comum, Raiz e Símbolos preferenciais como parênteses que envolvem 13 regras relacionadas aos conteúdos citados.
- **Operações Inversas:** Envolve 22 regras que observam as operações inversas na resolução de equações algébricas.
- **Fórmula de Bhaskara:** Envolve duas regras referentes a identificação de coeficientes de uma equação de 2º grau e quanto a aplicação da fórmula de forma equivocada.

Na situação das equações algébricas, essas barreiras epistemológicas, geralmente, derivam das “falsas regras” decoradas pelos alunos, ou ainda, pela incompreensão de conceitos básicos como igualdade entre membros, incógnita, princípios aditivos e multiplicativos, identificação de coeficientes.

Dentro do escopo do presente trabalho, foram definidas 56 *misconceptions* que

foram implementadas no Modelo Cognitivo para que o tutor possa identificar a falsa concepção do aluno e auxiliá-lo mostrando o conceito correto. Por exemplo, na resolução de uma equação de 1º grau com uma incógnita é necessário aplicar a operação inversa, de modo que a incógnita seja calculada. Os alunos frequentemente apresentam uma falsa concepção em relação a operação inversa de multiplicação. Por exemplo, para a equação “ $-2x=4$ ”, eles podem fornecer como resposta “ $x=4+2$ ”, enquanto a solução correta seria “ $x=4/2$ ”.

Assim, é importante que o sistema realize uma correção simultânea, explicando ao aluno o erro cometido. Esse tipo de retorno aos alunos é dado pelo PAT2Math. Na seção a seguir é apresentado o Modelo Cognitivo e explicado como este corrige e identifica as falsas concepções.

#### 4. Trabalho Proposto

Segundo VanLehn (2006), o funcionamento dos STIs pode ser resumido em dois *loops*: o *inner loop* e o *outer loop*, no qual cada *loop* é responsável por uma tarefa. O *outer loop* é responsável pela seleção dos exercícios para o aluno, tomando como base as informações do modelo do aluno. O *inner loop* é responsável por acompanhar o aluno durante a resolução do exercício, apresentado no *outer loop*, provendo a correção de cada passo de resolução da equação, ou indicando os erros cometidos.

Este trabalho encontra-se neste contexto do *inner loop*, uma vez que o Modelo Cognitivo (MC) do STI PAT2Math é responsável por verificar, validar e, se necessário, corrigir cada passo fornecido como resposta pelo aluno nos exercícios de resolução de equações. Na Figura 1 tem-se um exemplo de passos fornecidos pelo aluno na resolução de uma equação algébrica. Nesse exemplo, um passo seria a aplicação de uma ou mais operações para cada solução parcial da equação, como pode ser observado na Figura 1 (passos 1 a 3). O termo correção, aqui citado, se refere à detecção de erros comumente cometidos por alunos, ou seja, as *misconceptions*.

$2x - 24 = 0$ ; (Equação fornecida pelo Tutor) $2x = 24$ ; (passo 1) $x = 24/2$ ; (passo 2) $x = 12$ ; (passo 3)
---

**Figura 1: Exemplo de Resolução de Equação**

O MC foi implementado na *shell* de Sistema Especialista JBoss Drools 4.0 (<http://www.jboss.org/drools>), que permite o uso da linguagem de programação Java para a construção dos fatos e das regras. Neste contexto, fatos são as equações, e as regras representam um passo (operação) de resolução da equação, ou uma *misconception*. Há aproximadamente 33 regras representando as operações algébricas e 20 regras de falsas concepções representando as 56 falsas concepções obtidas. A maioria das regras trata duas ou mais falsas concepções por estas serem parecidas. Por exemplo, há uma única regra que trata os erros comuns de soma, subtração e multiplicação.

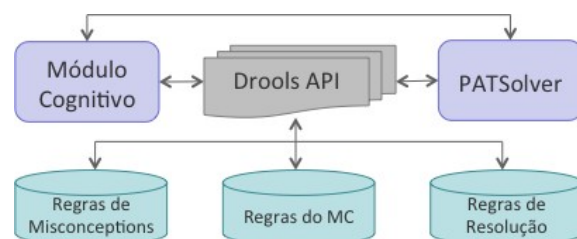
Cada equação é representada, no sistema, como uma árvore de expressões, pois este formato permite a representação da precedência de operadores sem o uso de parênteses. Essa representação torna mais simples a realização de cálculos e a manipulação dos termos (em relação a cadeias de caracteres), pois a movimentação de termos da equação limita-se à movimentação dos nós da árvore.

O MC está subdividido em três módulos: (i) o módulo resolvidor (PATSolver); (ii) o modelo cognitivo propriamente dito; (iii) e o módulo de falsas concepções (*misconceptions*). A arquitetura do MC é apresentada na Figura 2. Nesta figura nota-se que o módulo cognitivo, responsável pela correção das equações, utiliza a API da *shell*

Drools para poder acessar as regras. A interação entre o modelo cognitivo e a *shell* consiste em passar a equação para ela, seja para a correção da equação (Regras de Resolução) ou para a detecção das *misconceptions* (regras de *Misconceptions*). Há também uma comunicação entre o módulo cognitivo e o PATSolver para uso do modo de demonstração, a ser explicado nas seções que seguem.

#### 4.1. Resolvedor de Equações PATSolver

O módulo resolvidor, ou simplesmente PATSolver, é o módulo que detém o conhecimento relacionado ao domínio do sistema, ou seja, ele é responsável pela resolução de equações algébricas de 1º e 2º grau com uma incógnita. Como explicado anteriormente, para poder corrigir os passos de resolução fornecidos pelo aluno, o sistema deve também ser capaz de resolver as equações. Esse módulo possui esta função.



**Figura 2: Arquitetura do módulo cognitivo**

Este módulo pode operar tanto em *stand-alone* quanto junto com o modelo cognitivo. No modo *stand-alone*, ele possui uma interface própria para demonstração da resolução completa de uma equação. Já no modo em conjunto com o MC, este último encaminha ao resolvidor uma equação para ser resolvida em apenas um passo, de modo que seja possível futuramente compará-la com a equação do aluno, a fim de validá-la, ou não.

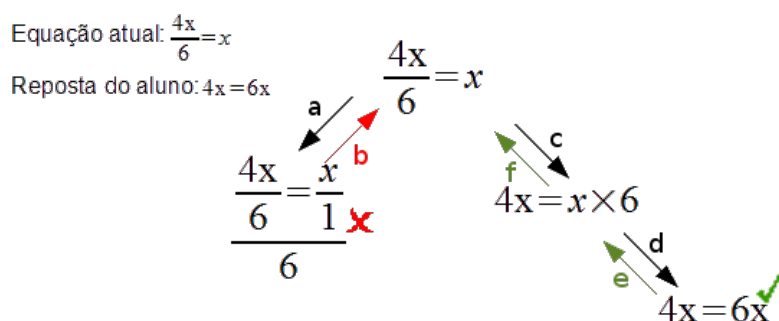
#### 4.2. Modelo Cognitivo

O Modelo Cognitivo (MC), conforme citado acima, é responsável pela aplicação do conhecimento do sistema de modo a verificar se a resposta apresentada pelo aluno está correta ou não. Para tanto, ele deve encontrar um caminho de resolução que o leve a resposta apresentada pelo aluno. Ele deve ter a capacidade de resolver a equação, pois desta forma é possível identificar o passo de resolução que o aluno possui dificuldade em resolver e desta forma auxiliá-lo com um *feedback* apropriado.

Para se encontrar um caminho de solução, de determinada equação, o MC irá realizar uma busca em profundidade, através de recursão, utilizando as regras do PATSolver. Uma vez que o resultado é obtido, é realizado o caminhamento de volta e, a cada nível retornado, é armazenada qual regra foi utilizada naquele nível. Chegando ao fim da recursão, o caminho de solução é obtido. Este algoritmo implementa a técnica conhecida como *model tracing* [Anderson et al., 1992], que descreve o uso do conhecimento do módulo especialista para determinar qual foi o conhecimento aplicado pelo aluno para resolver o exercício, testando todas as regras possíveis até que um resultado equivalente seja obtido. O algoritmo, implementado como uma função recursiva, pode ser resumido nos seguintes passos, sendo estes ilustrados na Figura 3:

1. Verificar a equivalência entre a equação do aluno e a equação do exercício. Se forem iguais retornar erro, pois a equação já está sendo utilizada.
2. Obter o próximo passo de resolução, através do PATSolver.
3. Se não há mais passos, sair da função indicando que a resposta ainda não foi encontrada (Figura 3 (b)).

4. Caso contrário, comparar as equações.
  - 4.a) Se forem equivalentes, armazenar a descrição do passo e a equação, e retornar avisando que a equação está correta (Figura 3 (e)).
  - 4.b) Senão, chamar recursivamente a função utilizando a equação obtida como parâmetro e repetir o passo 2 (Figura 3 (a,c,d)).
5. Se a função retornou indicando que a equação não está correta, repetir o passo 2(Figura 3 (b)).
6. Se a função retornou indicando que a equação está correta, armazenar a descrição do passo atual e a equação e sair da função indicando que a equação está correta (Figura 3 (e,f)).
7. Se for o fim da recursão e não foi encontrada uma equação equivalente, retornar erro ao usuário.



**Figura 3: Exemplo de execução do algoritmo do modelo cognitivo**

A base de regras foi estendida com um outro conjunto de regras que levam em conta a resposta do aluno. Essas novas regras verificam quais foram as alterações que o aluno realizou na equação, a partir do passo anterior, para decidir que operações poderiam ter sido aplicadas. Isso permite otimizar o processo de verificação que, sem essa, poderia ter uma complexidade maior, devido a testes desnecessários envolvendo outros termos da equação. De acordo com as alterações realizadas, uma regra diferente é aplicada sobre a equação. Por exemplo, o sistema forneceu a equação  $1+2+3=x$  para o aluno resolver, e o aluno informa como próximo passo de resolução a equação  $1+5=x$ . Ao MC comparar as duas equações, ele constata que os termos 2 e 3 não aparecem na equação fornecida como resposta pelo aluno, e o termo 5 não consta na equação do exercício. O MC verifica também que os termos estão ligados pela operação de soma e que ambos são valores inteiros, portanto pode ser tratado pela regra de soma. O MC aplica então a regra de soma sobre estes termos na equação.

Para cada equação obtida do PATSolver, o MC verifica se esta é equivalente a do aluno. O processo de comparação consiste em “resolver” a equação, atribuindo um valor arbitrário à incógnita e realizando as operações (avaliação da equação). Uma vez resolvidas as equações, restará um valor em cada lado da equação, por exemplo  $a=b$  onde “a” e “b” são números reais quaisquer. Se ambas as equações, fornecidas pelo PATSolver e pelo aluno, produzirem resultados iguais, então a equação é equivalente. Como pode haver a possibilidade de termos em equações diferentes produzirem o mesmo resultado, é realizada anteriormente uma verificação nos termos, no qual “termos” inclui números inteiros e incógnitas. Tal verificação consiste em uma comparação termo a termo entre as equações, ou seja, os termos da equação do PATSolver devem ser os mesmos da equação do aluno.

O MC oferece ao aluno alguns recursos que podem ajudá-lo na resolução da

equação, os quais:

- **Dica:** apresentação, de forma textual, das possíveis operações que podem ser aplicadas sobre a equação atual.
- **Mostrar Passo:** resolve a equação em um passo.
- **Mostrar Resolução:** resolve a equação, a partir do ponto em que o aluno parou, até o resultado final ser obtido.

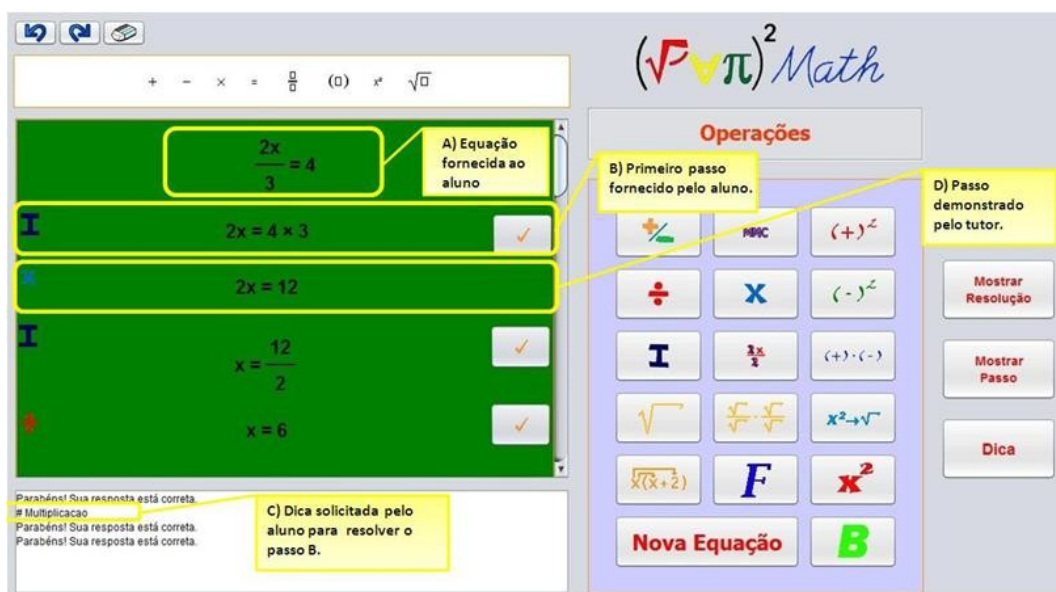


Figura 4: Interface do PATEquation

Para oferecer estes recursos de ajuda, o MC utiliza os recursos do módulo resolvidor. De fato, o recurso “Mostrar Resolução” opera da mesma forma que o PATSolver em modo *stand-alone*. Esses recursos de ajuda ao aluno compreendem o modo de demonstração, e os exercícios propostos, bem como a correção em tempo real, compreendem o modo de tutoramento. Maiores detalhes deste módulo estão descritos em [Seffrin et al., 2010].

O MC é utilizado pelo PATEquation: um módulo do STI PAT2Math para resolução de equações pelo aluno. Conforme mostrado na Figura 4, nesse módulo é fornecida uma equação para o aluno resolver. Cada passo da resolução da equação fornecida pelo aluno será corrigida pelo MC, que irá verificar se (1) a resposta é correta; e (2) se a operação escolhida pelo aluno (na interface de operações possíveis) é realmente a operação que foi aplicada. O primeiro caso procura saber se o aluno sabe “como” resolver a equação. O segundo caso tenta verificar se o aluno sabe identificar “qual” operação deve ser aplicada. Por exemplo, na Figura 4, como 1º passo da resolução, o aluno forneceu como resposta “ $2x=4 \times 3$ ” e como operação foi escolhida “Operação Inversa” (conforme símbolo mostrado a esquerda da resposta do aluno).

### 4.3. Falsas Concepções

Uma vez que o módulo cognitivo não consiga encontrar um caminho de resolução que leve à solução apresentada pelo aluno, ele assume que o aluno cometeu algum erro. Este erro provavelmente é devido a um obstáculo epistemológico que o aluno apresenta, também chamado de falsas concepções ou *misconceptions* (ver Seção 3). Partindo deste pressuposto, o módulo cognitivo deve detectar qual foi esta concepção, utilizando outro conjunto de regras que representam cada uma das concepções incorretas conhecidas

pelo sistema. Tais regras visam reproduzir, na equação do exercício, cada uma das *misconceptions* enunciadas, ou seja, o sistema deve cometer o mesmo erro que o aluno, de modo a identificá-lo.

A equação, geralmente, utilizada como fato no SE é a equação do passo anterior ao que o aluno errou, para que o MC possa reproduzir o erro cometido pelo aluno. Uma vez que tenha sido detectada ao menos uma *misconception*, esta é retornada ao MC. Se a *misconception* não for detectada, utilizando a primeira equação, utiliza-se as equações encontradas durante a busca pelo caminho de solução. Por exemplo, no exercício há a equação  $2x+3+4x+4=0$ , o aluno pode responder com  $2x+4x=7$ . Ao buscar pelo erro do aluno, o MC utilizará a primeira equação do exemplo para tentar reproduzir o erro do aluno. Sem sucesso, ele utiliza o passo intermediário  $2x+4x+7=0$ , neste caso a falsa concepção de Operação Inversa é detectada, pois o correto é  $2x+4x=-7$ . No caso de não ter sido encontrada uma *misconception* correspondente, é retornada uma mensagem indicando que nada foi detectado.

## 5. Trabalhos Relacionados

CTAT (*Cognitive Tutor Authoring Tool*) [Koedinger et al., 2004] é um sistema para a concepção de sistemas tutores inteligentes. Ele dispõe de diversas ferramentas para que qualquer pessoa, que não tenha familiaridade com programação, consiga desenvolver o seu próprio sistema tutor. Segundo [Koedinger et al., 2004], o desenvolvimento de um sistema tutor utilizando esta ferramenta garante um desenvolvimento ágil em comparação aos STI tradicionais, programados, pois a programação dos conteúdos é baseada em demonstrações.

Aleven (2009) utilizou o CTAT para desenvolver um sistema tutor de matemática, utilizando o modelo de programação por demonstração. Na interface deste tutor é demonstrado o exercício a ser realizado, bem como os possíveis acertos e erros que podem ser cometidos, que são organizados no grafo de comportamento (*behavior graph*). Para cada nodo do grafo, etapas possíveis de solução ou erro, são associadas a habilidades e ajudas possíveis (de resolução ou *feedback* de erro). Em outras palavras, todos os possíveis caminhos que o aluno poderá seguir estão definidos explicitamente no grafo de comportamento.

PAT2Math diferencia-se, pois como foi implementado em módulos, permite que estes módulos sejam utilizados isoladamente como OA. Por exemplo, o PATSolver e o PATEquation, ambos componentes do MC, foram também disponibilizados como OAs, com respectivas interfaces.

Ainda, sendo implementado como um SE, o MC do PAT2Math permite maior liberdade na resolução de equações, ao invés de se limitar o número de exemplos àqueles inseridos previamente pelo professor. Isto permite com que cada aluno escolha um caminho diferente de resolução, e mesmo assim possa atingir a resposta correta. Ele também facilita o trabalho do professor, que não precisa fornecer todos os caminhos possíveis de solução pelo aluno em um grafo de comportamento. O uso de um SE no modelo cognitivo também permite, em futuras implementações, com que o professor insira os próprios exercícios (equações a resolver) no tutor, uma vez que ele pode resolver qualquer equação de 1º e 2º grau. Outro diferencial está no fato que toda a implementação segue os PCNs e a forma como o conteúdo da álgebra é ensinado no Brasil.

## 6. Avaliação

O MC foi avaliado segundo duas abordagens: *direta* e *indireta*. Na abordagem *direta*



foram realizados diversos testes unitários com o PATSolver e com o MC. Testes Unitários, também conhecidos como testes unidade, são um processo que inclui o desempenho do planejamento do teste, a aquisição de um conjunto de teste e a medida de uma unidade de teste em relação aos requisitos [IEEE SB, 1986]. Esses testes foram realizados durante a fase de concepção do sistema. Para cada nova operação algébrica implementada no MC, esta foi testada com diversos exemplos de equações que exigem a utilização desta operação para sua resolução. Além disso, cada vez que uma nova operação era implementada, todas as operações implementadas eram novamente testadas. Para realização destes testes unitários, foi elaborado um roteiro, onde eram testadas tanto as funcionalidades algébricas (por exemplo, se o aluno inserir o resultado correto de uma soma no sistema, o mesmo reconhece como correto o resultado), quanto as funcionalidades de interface gráfica do PATEquation (botões, menus e eventos de clique).

Na abordagem *indireta*, foram realizadas avaliações através das interfaces do PATEquation e do PATSolver. Nesta abordagem, dois principais métodos de avaliação foram realizados: testes unitários e avaliações quantitativas e qualitativas com usuários. Os testes unitários reproduziram a resolução de diversas equações, conforme explicado anteriormente.

As avaliações com usuários foram aplicadas em diversas etapas do desenvolvimento, sendo do tipo somativas (buscam aprimorar o sistema) e formativa (buscavam avaliar uma versão final do sistema) [Woolf, 2009]. Assim, avaliações detalhadas realizadas anteriormente podem ser encontradas em [Seffrin et al., 2009; Seffrin et al., 2010]. Para a avaliação do MC, descrito neste artigo, foi realizada uma avaliação qualitativa somativa, inicialmente, buscando verificar com nove professores a maneira como é dado o *feedback* aos alunos, bem como as maneiras como os professores auxiliam seus alunos no processo de ensino de equações algébricas. Ainda foram realizadas observações sobre o sistema verificando a interação deste com os alunos, além de sugestões de melhorias. Posteriormente, foi realizada uma avaliação formativa, também de cunho qualitativo, com seis alunos (três de 6ª série e três de 8ª série) para verificar a interação dos alunos com o sistema observando se o auxílio através da correção simultânea, dicas, entre outros auxiliam o aluno em seu aprendizado. Assim, observou-se que esses recursos contribuem para a tomada de decisão do aluno no momento da resolução de equações. Ainda, os recursos de dica, correção simultânea e mostrar passo auxiliam o aluno não desestimulando-o na interação com o sistema e na resolução de equações. Salienta-se que, se faz necessário, novas avaliações de caráter quantitativo e com um número maior de usuários.

## 7. Conclusão

A implementação do modelo cognitivo ocorreu em etapas, cada módulo foi implementado separadamente, iniciando-se pelo PATSolver. Este, conforme descrito, foi submetido a avaliações com professores, de acordo com os resultados destas avaliações foi-se adequando a terminologia do PATSolver à terminologia comumente utilizada em sala de aula. Como por exemplo, utiliza-se “Operação Inversa – Principio Aditivo” em vez de “Passar o termo para o outro lado da equação”. O segundo módulo implementado foi o módulo cognitivo, cujas regras herdaram a terminologia do PATSolver. Conforme este ia evoluindo, ele ia sendo integrado como componente do PATEquation. Este também, conforme visto, foi submetido a diversas avaliações. E por último concluiu-se o desenvolvimento do módulo de *misconceptions* que, ao contrário do PATSolver, foi integrado diretamente ao MC.

Nas avaliações qualitativas, os professores colocaram que consideram que o PAT2Math pode contribuir significativamente na aprendizagem dos alunos. Conforme sugestões colocadas pelos mesmos, ainda são necessárias algumas melhorias de forma a engajar mais os alunos no uso do sistema, visto que estes se desestimulam facilmente se o processo se tornar repetitivo. Considerar aspectos emocionais e motivacionais do aluno é uma das ambições futuras do projeto.

Um outro trabalho futuro, já em desenvolvimento, é um módulo de ajuda mais abrangente, ou seja, este dará dicas ao aluno mais complexas, e não apenas os próximos passos de solução. Estas dicas serão de diversos níveis, conforme dificuldades percebidas pelo tutor no aluno.

## Referências

- Aleven, V. et al. (2009) Scaling up programming by demonstration for intelligent tutoring systems development. *IEEE Trans. on Learning Technologies*, 2(2), 64-78.
- Anderson, J. R. et al. (1992) General principles for an intelligent tutoring architecture. *Cognitive approaches to automated instruction*, p. 81-106.
- Bloom, B. S. (1984) The 2 Sigma Problem: The search for methods of group instruction as effective as one-to-one tutoring. In: *Educational Researcher*. p. 4-16.
- Borba, M. C. (1999) Tecnologias informáticas na educação matemática e reorganização do pensamento. In: Bicudo, M. *Pesquisa em educação matemática*. SP: UNESP.
- Brasil (2001) Ministério da Educação. Secretaria de Ensino Fundamental. Parâmetros curriculares nacionais: matemática. Brasília v. 3.
- Brousseau, G. (1983) Les obstacles épistémologiques et les problèmes en Mathématiques. *Recherches em Didactique des Mathématiques*, 4(2), 165-198.
- Coxford, A. F.; Shulte, A. P. (1995) *As idéias da álgebra*. São Paulo: Atual.
- Gravina, M. A.; Santarosa, L. M. (1998) *A Aprendizagem da Matemática em Ambientes Informatizados*. IV Congresso RBIE, Brasília.
- IEEE Standards Board (1986) *IEEE Standard for Software Unit Testing: An American National Standard, ANSI/IEEE Std 1008-1987*. IEEE Computer Society.
- Koedinger, K. R.; Aleven, V.; Heffernan, T.; McLaren, B.; Hockenberry, M. (2004). *Opening the Door to Non-Programmers*. In: *Proceedings of 7th ITS*. Maceio, Brazil
- Lima, R. N. (2007) *Equações Algébricas no Ensino Médio*. Tese (Doutorado em Educação Matemática) – PPGEM - PUCSP, São Paulo.
- Psozka, J. et al (1988). *Intelligent Tutoring Systems: Lessons Learned*. Lawrence E..
- Seffrin, H.; Rubi, G.; Carlotto, T.; Mello, G.; Jaques, P. A. (2009). Um resolvidor de equações algébricas como ferramenta de apoio à sala de aula no ensino de equações algébricas. In: *Workshop de Informática na Escola - CSBC*. p. 1791-1800.
- Seffrin, H.; Rubi, G.; Cruz, B.; Damasceno, F.; Jaques, P. (2010) *PATequation: Um Objeto de Aprendizagem para Apoio a Prática de Resolução de Equações*. In: *V LACLO, 2010*.
- Stacey, K.; MacGregor, M. (1999) Learning the Algebraic Method of Solving Problems. *The Journal of Mathematical Behavior*, Norwood, NJ, v. 18, n. 2, 149-167
- VanLehn, K. (2006) The behavior of tutoring systems. In: *International Journal of Artificial Intelligence in Education*, v. 16, n. 3, p. 227-265, 2006.
- Wenger, E. (1987) *Artificial Intelligence and Tutoring Systems: Computational and Cognitive Approaches to the Communication of Knowledge*. Morgan Kaufmann.
- Wolf, B. P. (2009). *Building Intelligent Interactive Tutors*. M. Kaufman.