

## Tutoria Inteligente Completa para os Conceitos Formais da Lógica Proposicional: Experimentos e Resultados

João Carlos Gluz<sup>1</sup>, Rafael Bueno<sup>1</sup>, Rafael Koch Peres<sup>2</sup>, Fabiane Penteadó Galafassi<sup>2</sup>

<sup>1</sup>Pós-Graduação em Computação Aplicada (PPGCA) – Universidade do Vale do Rio dos Sinos (UNISINOS) – Av. Unisinos 950 – 93022-000 – São Leopoldo – RS – Brazil

<sup>2</sup>Instituto de informática, UFRGS, CEP: 91501-970, Porto Alegre, RS – Brasil

jcgluz@unisinos.br, rafaelbueno@outlook.com, rafaelkperes@gmail.com, fabiane.penteadó@gmail.com

**Abstract.** *Despite the importance of Logic discipline for the curriculum of scientific and technological areas, there are several problems with the teaching of this discipline in undergraduate courses. Computational tools, especially intelligent tutoring systems, can have a positive impact on the learning of Logic. The present work presents an intelligent environment for teaching Propositional Logic, which stands out from its predecessors in two fundamental aspects: an adherence to Dialectic and Socio-Historical principles in its tutoring processes, and a learning support for main concepts of Propositional Logic. The paper introduces the user interface and tutoring features of the environment. Empirical experiments conducted with the environment shown, with a high level of statistical significance, that the environment had a positive impact on learning of Logic and was well accepted by students.*

**Resumo.** *Apesar da importância da disciplina de Lógica para o currículo de áreas científicas e tecnológicas, existem vários problemas com o ensino desta disciplina em cursos de graduação. As ferramentas computacionais, especialmente os sistemas tutores inteligentes, podem ter um impacto positivo na aprendizagem da Lógica. O presente trabalho apresenta um ambiente inteligente para o ensino da Lógica Proposicional, que se destaca de seus predecessores em dois aspectos fundamentais: a adesão aos princípios dialécticos e sócio-históricos em seus processos de tutoria e um suporte de aprendizagem para os principais conceitos da Lógica Proposicional. O artigo apresenta a interface do usuário e os recursos de tutoria do ambiente. Experimentos empíricos realizados com o ambiente mostram, com um alto nível de significância estatística, que o ambiente teve um impacto positivo na aprendizagem da Lógica e foi bem aceito pelos alunos*

### 1. Introdução

A disciplina de Lógica Formal é fundamental para o currículo das áreas científicas e tecnológicas (genericamente agrupadas sob a sigla inglesa STEM – *Science, Technology, Engineering, and Mathematics*). Estudos empíricos indicam sérias dificuldades encontradas por alunos em assimilar os conceitos dessa disciplina, particularmente quando ela é ministrada nos semestre iniciais dos cursos de Computação e Informática (Penteadó e Gluz, 2011).

Ferramentas computacionais, principalmente aquelas como suporte a tutoria inteligente, podem trazer um impacto positivo no ensino de Lógica Formal, permitindo que alunos observem e reflitam sobre seus erros de raciocínio lógico e possam avançar

no processo de ensino-aprendizagem dessa disciplina. Tais ferramentas têm uma longa história de desenvolvimento e aplicação na Computação. O presente trabalho apresenta o ambiente inteligente *Heráclito* de ensino de Lógica proposicional, que se destaca de seus predecessores em três aspectos fundamentais: uma completa aderência aos princípios Dialéticos e Sócio-Históricos na implementação dos processos de mediação e tutoria, um modelo de aluno adaptativo e flexível e um suporte completo ao ensino dos conceitos formais da Lógica Proposicional.

A primeira versão do ambiente foi originalmente escrita em uma ampla variedade de linguagens de programação e com componentes desenvolvidos especificamente para as plataformas web e tablet (Gluz et al., 2014; Penteado e Gluz 2011). Para reduzir problemas de manutenção, suportar interface responsiva e permitir escalabilidade o ambiente foi completamente reescrito de 2014 a 2015 em Prolog, HTML5 e Java. Durante o ano de 2016 foram conduzidos experimentos empíricos com o ambiente com duas turmas da disciplina de Lógica, uma em cada semestre. Um total de 71 alunos distintos participaram dos experimentos que visaram avaliar o impacto pedagógico e a usabilidade do ambiente.

O impacto pedagógico do ambiente foi avaliado através de experimentos empíricos conduzidos de acordo com a metodologia de pré e pós-teste. O resultado destes experimentos mostrou, com um bom índice de significância estatística, que o impacto positivo do ambiente não pode ser descartado. A usabilidade foi avaliada pela metodologia TAM2, resultando em uma boa avaliação em termos da utilidade e facilidade de uso do ambiente do ponto de vista de seus usuários. Após estes resultados, o ambiente entrou em uso experimental e pode ser acessado no endereço <http://obaa.unisinos.br/heraclito/>.

## 2. A Lógica Proposicional

### 2.1. Cálculo Proposicional

O cálculo proposicional é um sistema formal capaz de representar conhecimento lógico por fórmulas que combinam proposições através de operadores lógicos. O alfabeto deste cálculo é constituído por: símbolos de valor-verdade: V ou F; variáveis lógicas: A, B, C, ..., A<sub>1</sub>, A<sub>2</sub>, ...; operadores lógicos:  $\neg$  (negação),  $\wedge$  (conjunção),  $\vee$  (disjunção),  $\rightarrow$  (condicional) e  $\leftrightarrow$  (bicondicional); parênteses. Fórmulas Bem Formada (FBFs) são definidas indutivamente como: qualquer variável lógica é uma FBF; se P é uma FBF, então  $\neg P$  também é; se P e Q são FBFs, então  $(P \wedge Q)$ ,  $(P \vee Q)$ ,  $(P \rightarrow Q)$  e  $(P \leftrightarrow Q)$  também são; nada mais é FBF. A ordem de precedência usual desse cálculo é: 1) parênteses ( e ), 2)  $\neg$ , 3)  $\wedge$ , 4)  $\vee$ , 5)  $\rightarrow$  e 6)  $\leftrightarrow$ . Quando todos os valores-verdade das variáveis lógicas de uma fórmula estão definidos, então é possível calcular o valor-verdade da fórmula, através da avaliação de suas expressões seguindo a ordem de precedência. O valor das expressões formadas por um operador lógico cujos operandos são variáveis lógicas é calculada seguindo a tabela-verdade de cada operador lógico.

### 2.2. Dedução Natural na Lógica Proposicional

A afirmação que uma proposição lógica é verdadeira em consequência de outras proposições lógicas serem verdadeiras expressa um *argumento lógico*. Um argumento é válido quando a proposição que forma sua *conclusão* Q for verdadeira sempre que as demais proposições P<sub>1</sub>, P<sub>2</sub>, ..., P<sub>n</sub>, que são suas *premissas* ou *hipóteses*, forem verdadeiras. Isso é simbolizado através da expressão:

$$P_1, P_2, \dots, P_n \vdash Q$$

Porém nem todos os argumentos que podem ser expressos são válidos. A validade de um argumento pode ser provada por uma demonstração ou prova formal construída por um método de dedução. Os métodos de dedução começam com as premissas  $P_1, P_2, \dots, P_n$ , e através da aplicação sucessiva de regras de dedução (ou de inferência) obtém fórmulas intermediárias até chegar na conclusão  $Q$ . O *Heráclito* usa o método de Dedução Natural, por ser considerado um método que se aproxima da forma como deduções lógicas são feitas naturalmente (este método foi desenvolvido por Gentzen para provas da Teoria dos Números (Nolt et al., 1998)). A Tabela 2 mostra as 10 regras de dedução do método de dedução natural para a Lógica Proposicional. Tais regras foram baseadas em (Nolt et al., 1998; Baronett, 2008). O método de dedução natural é completo e correto, ou seja, ele permite provar todos os argumentos que são válidos e todas as inferências feitas com suas regras preservam a validade.

**Tabela 2:** Regras de Inferência da Dedução Natural

Introdução de Operadores	Eliminação de Operadores
Redução ao absurdo (RAA) - $\neg I$ $\begin{array}{l} P \\ \dots \\ Q \wedge \neg Q \\ \hline \neg P \end{array}$	Dupla negação (DN) - $\neg E$ $\frac{\neg \neg P}{P}$
Prova condicional (PC) - $\rightarrow I$ $\begin{array}{l} P \\ \dots \\ Q \\ \hline P \rightarrow Q \end{array}$	Modus Ponens (MP) - $\rightarrow E$ $\frac{P \quad P \rightarrow Q}{Q}$
Conjunção (Cj) - $\wedge I$ $\frac{P \quad Q}{P \wedge Q}$	Simplificação (Sp) - $\wedge E$ $\frac{P \wedge Q}{P} \quad \frac{P \wedge Q}{Q}$
Adição (Ad) - $\vee I$ $\frac{P}{P \vee Q} \quad \frac{P}{Q \vee P}$	Eliminação da disjunção (-Dj) - $\vee E$ $\frac{P \vee Q \quad P \rightarrow R \quad Q \rightarrow R}{R}$
Introd. da Equivalência (+Eq) - $\leftrightarrow I$ $\frac{P \rightarrow Q \quad Q \rightarrow P}{P \leftrightarrow Q}$	Eliminação da Equivalência (-Eq) - $\leftrightarrow E$ $\frac{P \leftrightarrow Q}{P \rightarrow Q} \quad \frac{P \leftrightarrow Q}{Q \rightarrow P}$

### 3. Estado da Arte em Sistemas Tutores de Lógica

Atualmente existem vários exemplos de sistema de tutoria inteligente para Lógica. O sistema DIALOG (Benzmuller et al, 2007) usa processamento de linguagem natural e um provador de teoremas de alta ordem para ensinar prova avançada de teoremas na Lógica Matemática. Na categoria de dedução simbólica, o trabalho que mais se destaca é o ambiente Logic-ITA (Yacef, 2010), que é um sistema de tutoria inteligente com uma arquitetura bastante tradicional para o ensino de dedução na Lógica Proposicional. Os sistemas KRRT (Alonso et al., 2007), P-Logic Tutor (Lukins et al., 2002) e Deep Thought's Hint Factory (Stamper, 2012) também são exemplos importantes de sistemas tutores de Lógica. Trabalhos recentes nesta área começaram a explorar o uso de modelos probabilísticos para inferir propriedades do modelo de aluno (Barnes e Stamper et al., 2010). O trabalho de Costa et al. (2012) está em fase preliminar, mas oferece uma aplicação interessante de sistemas multiagentes capaz de suportar múltiplas representações ou pontos de vista para ensinar Lógica Proposicional. O sistema web Elementar (Andrade e Canese, 2013) se concentra no ensino de dedução na Lógica por meio de um assistente de provas que usa técnicas de gamificação.

O ambiente *Heráclito* apresentado neste trabalho se diferencia desses sistemas através de várias contribuições importantes: a) é o único ambiente a oferecer suporte completo ao ensino de todos os conteúdos da Lógica Proposicional, incluindo construção de fórmulas, tabelas-verdade e provas de argumentos; b) é o único dos sistemas tutores estudados a ter um modelo de aluno adaptativo e flexível que permite uma completa liberdade para escolher qual fórmula, argumento ou teorema será usado

como exercício, sem se restringir ao suporte de apenas um conjunto de casos pré-definidos; c) é o único dos sistemas tutores de Lógica a aplicar a teoria sócio-histórica nos processos de mediação e tutoria. O *Heráclito* suporta a resolução e oferece tutoria em exercícios que cobrem os conteúdos da Lógica Proposicional, permitindo aos alunos avançar passo a passo na aprendizagem dos conceitos dessa disciplina.

Uma característica comum para em todos os sistemas de tutoria de Lógica estudados que são voltados ao ensino de provas de argumentos, incluindo o ambiente *Heráclito*, é a estrutura do modelo de aluno, que é derivada da estrutura de provas formais. No entanto, para o *Heráclito* foi desenvolvido um provador automático de argumentos lógicos, que é capaz de imitar o raciocínio utilizado pelos alunos, permitindo manter um bom sincronismo entre o modelo de aluno e o serviço de tutoria fornecido pelo sistema (Gluz et al., 2014). Diferente do Logic-ITA (Yacef, 2005), o *Heráclito* pode analisar a validade lógica do passo atual da prova e identificar a utilidade que este passo assume na estratégia geral de prova geral. De maneira similar, o *Heráclito* também resolve de forma automática qualquer exercícios de construção de fórmula ou de tabela-verdade submetido pelo aluno, e usa esta solução como base do modelo de aluno. São estas capacidades de resolução prévia e, no caso das provas formais, adaptativa dos problemas trabalhados pelos alunos, que permitem que o modelo de aluno do *Heráclito* seja flexível e adaptativo, o que o diferencia de todos os outros trabalhos, pelo menos no contexto do ensino de Lógica Proposicional.

Em termos da aplicação pedagógica da Teoria Sócio-Histórica o *Heráclito* não força os alunos a se conformarem a uma única solução por um problema. Mesmo quando a solução apresentada pelo aluno diverge da solução do tutor, principalmente no caso das provas formais, inicialmente são aplicadas estratégias pedagógicas que encorajam o aluno a refletir sobre problemas ou erros diagnosticados pelo sistema. É importante que o aluno possa refletir um pouco sobre um passo problemático, sem que o sistema ofereça uma solução imediata. Tal solução e explicação serão eventualmente fornecidas, mas apenas com a evolução do processo dialético de mediação.

#### 4. Resolução de Exercícios de Lógica Proposicional

O ambiente *Heráclito* é capaz de suportar os principais tipos de exercícios relacionados aos conceitos formais da Lógica Proposicional. São suportados exercícios que permitem aos alunos avançar passo a passo na aprendizagem dos conceitos de proposições, operações lógicas, fórmulas, cálculo do valor-verdade, construção de tabelas-verdade e resolução (prova formal) de argumentos e teoremas da Lógica Proposicional.

Para tanto, o *Heráclito* implementa duas ferramentas de apoio a resolução de exercícios: uma para exercícios de prova de argumentos da Lógica Proposicional e outra para exercícios de fórmulas proposicionais. Ambas ferramentas possuem uma interface web responsiva, capaz de se adaptar automaticamente as plataformas *tablet*, *smartphone* e *computador*. A ferramenta de exercícios de provas de argumentos suporta apenas exercícios envolvendo a construção passo a passo da demonstração formal de um argumento ou teorema da Lógica Proposicional. A ferramenta de exercícios de fórmulas suporta três tipos principais de exercícios: a) exercícios para decompor e identificar corretamente as sub-fórmulas, proposições simples (variáveis) e operadores lógicos de uma fórmula, b) exercícios para calcular o valor-verdade de uma fórmula devidamente decomposta, permitindo observar detalhadamente como esse valor é construído e c) exercícios para a montagem coluna a coluna e linha a linha de uma tabela-verdade.

A ferramenta de resolução de exercícios de provas, permite iniciar um exercício tanto entrando com um argumento personalizado, quanto selecionando um argumento dentre um conjunto de opções de provas básicas, intermediárias ou avançadas. A resolução de um exercício de prova se dá em duas etapas: a adição de hipóteses do

argumento, caso estas existam, e, posteriormente, a construção passo a passo da prova formal pela aplicação de regras de inferência. Uma prova é considerada concluída quando o último passo gerado, corresponde a conclusão do argumento. A Figura 1 mostra um exemplo de resolução de exercício de prova, em duas versões: a tela maior corresponde a interface em um computador e a tela menor corresponde à interface em um *tablet*. Os botões à esquerda permite escolher a regra de inferência a ser aplicada, a tabela numerada ao lado corresponde à prova sendo construída, onde cada linha numerada corresponde a um passo da prova, a primeira coluna mostra a fórmula obtida no passo e a coluna mais à direita mostra a justificativa para a obtenção dessa fórmula, composta da identificação da regra de inferência usada e de referência às linhas anteriores da prova usadas na aplicação da regra. O campo em verde no final da tela é a área de interação do serviços de tutoria. Mensagens de supervisão indicando, por exemplo, erros na operação da regras de inferência são apresentadas por telas *popup*.

Na ferramenta de exercícios de proposicionais, também é possível entrar com uma fórmula de livre escolha do aluno ou selecionar uma fórmulas dentre uma lista de fórmulas pré-cadastrados. Os tipos de exercícios que podem ser executados sobre fórmulas da Lógica Proposicional são: decomposição da fórmula em árvore, cálculo do valor-verdade da fórmula ou construção da tabela-verdade da fórmula.

De acordo com Gersting (2012, p. 302) “expressões algébricas envolvendo operações binárias podem ser representadas por árvores binária...”. Assim como as expressões algébricas, as fórmulas proposicionais são avaliadas quanto ao seu valor-verdade aplicando-se um operador sobre dois operandos. Um operando pode ser uma variável proposicional ou uma subfórmula derivada da fórmula principal.

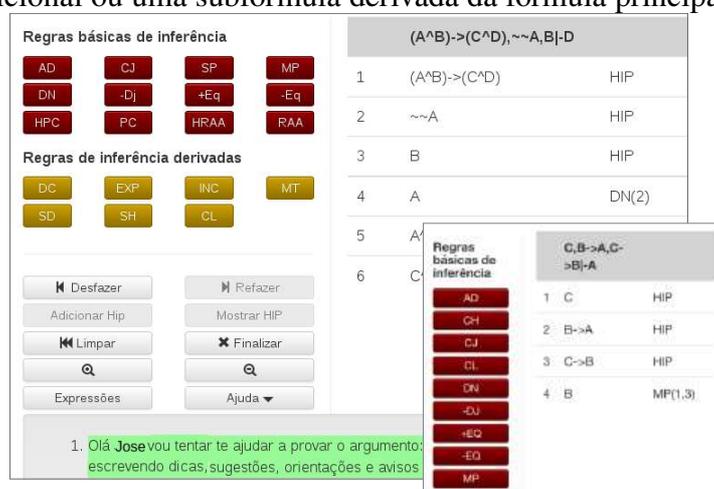


Figura 1. Interface de Exercícios de Prova de Argumentos

A funcionalidade de decomposição em árvore permite que seja realizado o desmembramento das subfórmulas de uma fórmula através de uma representação em árvore binária. A decomposição ocorre de cima para baixo, ou seja, na raiz da árvore encontra-se a fórmula original e nas folhas as variáveis proposicionais, que são as unidades mínimas de uma fórmula proposicional.

O exercício de cálculo do valor-verdade, consiste em calcular o valor lógico de uma fórmula proposicional. Este tipo de exercício pode ser começado logo após a decomposição de uma fórmula em árvore, ou então pode ser fornecida uma fórmula nova para a execução do exercício. Em ambos casos o ambiente realiza a decomposição da fórmula exibindo-a na forma de uma árvore binária, porém com campos adicionais nos nós da árvore as opções “V” (verdadeiro) e “F” (falso). A Figura 2 mostra a tela com o resultado do cálculo do valor-verdade fórmula após preenchimento dos campos e acionamento do botão avaliar exercício.

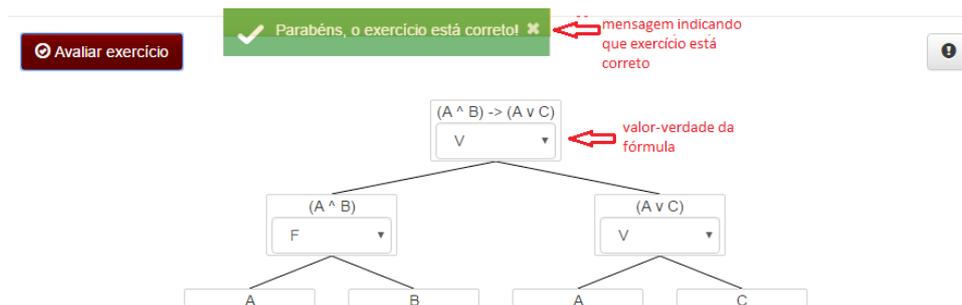


Figura 2. Exemplo de Resultado Cálculo de Valor-Verdade

Outra forma de realizar o cálculo de uma fórmula proposicional é através de uma tabela-verdade. Uma tabela-verdade é o conjunto de todas as possibilidades combinatórias que as variáveis proposicionais podem assumir. A tabela é construída da esquerda para a direita com colunas contendo subfórmulas da fórmula do exercício decompostas segundo a precedência dos operadores. O início da construção de uma tabela-verdade no *Heráclito* é automatizado, ou seja, a ferramenta cria a tabela e insere colunas contendo as variáveis proposicionais da fórmula e também linhas de acordo com a quantidade de variáveis. Para inclusão de novas colunas na tabela, o usuário clica com o botão direito sobre uma coluna existente (ou pressiona a coluna por alguns segundos caso esteja utilizando um dispositivo móvel) e seleciona a opção para inclusão de uma nova coluna à direita ou à esquerda. A Figura 3 mostra um exemplo de exercício de tabela-verdade resolvido.

A	B	$\sim A$	$(A \wedge B)$	FINAL
V	V	F	V	F
V	F	F	F	V

Figura 3. Exemplo de Exercício de Construção de Tabela-Verdade Resolvido

## 5. Tutoria Baseada na Teoria Sócio-Histórica

A aderência aos princípios Dialéticos e Sócio-Históricos pelo ambiente *Heráclito* parte da pressuposição que cada exercício de Lógica se constitui em uma Cena de Atenção Conjunta (JAS, do inglês *Joint Attention Scene*) (Tomasello, 2001), onde o processo dialético de mediação entre o ambiente e o aluno, permite a este último evoluir sua Zona de Desenvolvimento Proximal (ZDP) (Vygotsky, 1986). No caso do *Heráclito* há uma ampla liberdade de escolha do objeto a ser trabalhado na JAS: não há uma restrição de escolhas de exercícios particulares. Dentro das formas de exercícios suportadas pelo ambiente, alunos e professores podem escolher livremente quais exercícios serão trabalhados.

O presente trabalho assume que a mediação ocorre sob um processo de regulação proposto por Diaz et al. (1992) e adaptado por Passerino e Santarosa (2008) que começa a partir da categoria de *controle* e passa pela categoria de *autocontrole* com o objetivo de alcançar a categoria de *autorregulação*. A categoria de controle é identificada pela necessidade de ações mediadoras geradas pelo sujeito mais experiente e direcionadas ao sujeito menos experiente e tem duas dimensões: direta ou indireta. O controle direto é verificado através de pedidos, diretivas e questionamentos. O controle

indireto pode ser identificado através de questões perceptuais, conceituais ou processuais. Isso culmina com o estabelecimento de uma distância física (o sujeito mais experiente deixa o menos experiente sozinho, apenas observando o processo) que se enquadra na categoria de autocontrole. Diaz et al. (1992) definem o autocontrole como a realização pelo sujeito de uma ação obedecendo a expectativa de um tutor internalizado. A autorregulação não pode ser observada diretamente porque é interna ao sujeito. No entanto, considera-se que o sujeito está nesta categoria quando organiza, planeja e executa ações sem a intervenção do mediador externo. Assim, a autorregulação é alcançada quando o sujeito pode conceber um plano de ação e segui-lo como se fosse seu próprio tutor.

Aplicando este modelo de regulação ao processo de tutoria, o *Heráclito* assume que o aluno, como sujeito menos experiente, começa a partir de um estado onde há necessidade de intervenções pedagógicas correspondentes ao controle externo direto, passando a um estado onde as intervenções visando o controle indireto se tornam mais eficazes e gradualmente passando a ter autonomia para ser capaz de fazer ações autocontroladas, onde as ações pedagógicas se tornam mais reflexivas. O *Heráclito* suporta vários tipos de intervenção pedagógica, incluindo ações de supervisão, orientação, confirmação, suporte, contestação e reflexão. As intervenções de supervisão, orientação e confirmação, são classificadas como controle direto, enquanto confirmação, suporte e contestação são formas de controle indireto. As ações de reflexão visam ajudar na evolução cognitiva do aluno do autocontrole para a categoria de autorregulação.

Para cada exercício (objeto da JAS), o ambiente oferece um serviço de supervisão elementar de erros que detecta e avisa sobre erros básicos na resolução do exercício (por exemplo, erros na decomposição da fórmula, no preenchimento de um valor da tabela-verdade, ou na aplicação de uma regra de inferência). Em concomitante também é fornecido um serviço de tutoria inteligente capaz de diagnosticar e agir sobre situações críticas nas categorias de controle direto e indireto. Assim são detectadas situações que requerem intervenções de orientação (ajuda), confirmação (incentivo), suporte (dicas e indicações), contestação (avisos e alertas de possíveis problemas) e chegando a possibilidade do ambiente sugerir processos de reflexão, através de exemplos e referências de leitura, que ajudam o aluno a avançar para a autorregulação.

## 6. Experimentos e Resultados

A contribuição educacional do ambiente *Heráclito* foi avaliada por três experimentos em sala de aula com alunos da disciplina de Lógica do Curso de Ciência da Computação da Unisinos. Cada experimento foi realizado em um turno de aula, com a realização de um pré-teste sem o auxílio do ambiente, seguida da realização de pós-teste com o uso do ambiente. O conteúdo de Lógica Proposicional relativo ao experimento já havia sido visto em aula anteriormente e não foi apresentado ou revisto nenhum conteúdo da disciplina durante a realização do experimento. Os exercícios aplicados no pré-teste e no pós-teste foram elaborados em conjunto com o professor da disciplina. Nestes exercícios foram considerados aspectos como quantidade de variáveis proposicionais, tipos de operadores utilizados e complexidade das provas de argumentos, para que em ambos os testes o nível de dificuldade fosse equivalente. Cada teste foi realizado por um período de 40 minutos. Entre um teste e outro, houve um período de 30 minutos para explicação do funcionamento do ambiente. A entrega dos exercícios resolvidos foi manual, ou seja, após resolução com o ambiente, foi necessário realizar a transcrição dos exercícios para o papel que seria entregue.

O experimento 1 foi realizado no primeiro semestre de 2016 e contou com a participação de 35 alunos. Este experimento avaliou o impacto pedagógico da ferramenta de resolução de exercícios de prova de argumentos, por já estar disponível no

*Heráclito*. No segundo semestre de 2016 foram realizados dois experimentos com a mesma turma. O experimento 2 contou com a participação de 36 alunos e avaliou o impacto da ferramenta de resolução de exercícios de fórmulas, que passou a estar disponível no segundo semestre de 2016. O experimento 3 realizado com essa turma, para uma segunda avaliação da ferramenta de resolução de prova de argumentos, foi conduzido posteriormente e contou com a participação de 33 alunos que já haviam participado do experimento 2.

A Tabela 3 mostra a média dos percentuais de acertos dos pré e pós testes de cada experimento e o desvio padrão dessa média. Os resultados mostram que houve uma melhoria (aumento) perceptível nas médias do pré-teste para o pós-teste. Além disto, é possível perceber que em dois experimentos no pós-teste, a variabilidade da taxa de acertos (desvio padrão) foi menor que no pré-teste. No outro experimento, cuja média não sofreu um aumento muito grande, o desvio permaneceu da mesma ordem. Outro fato interessante é de que mais alunos no pós-teste atingiram taxa máxima de 100% de acertos, inclusive esta foi o percentual com maior ocorrência (moda) no pós-teste de todos os experimentos.

**Tabela 3:** Resultados dos experimentos educacionais

Experimento	Pré-teste			Pós-teste		
	Média	DP	Moda	Média	DP	Moda
1	69,1%	30,8%	80%	98,8%	4,7%	100%
2	66,4%	21,4%	60%	91,4%	14,2%	100%
3	63,6%	28,9%	60%	74,5%	30,5%	100%

Ainda que o resultado seja perceptivelmente melhor não é possível aceitar a hipótese de que o ambiente *Heráclito* traz um impacto positivo na aprendizagem sem antes aplicar um teste de amostras pareadas para avaliar se a diferença encontrada entre as médias é significativa. Com este objetivo, para cada experimento foi feito um teste não paramétrico pareado Wilcoxon usando a ferramenta R. Este teste foi escolhido por ser o teste recomendado quando as amostras não tem uma distribuição normal (Malhotra, 2016, p. 226).

O teste de Wilcoxon indica que a hipótese nula, que não houve aumento nas médias, pode ser rejeitada para um nível de significância menor do que 5%, ou seja, quando  $p\text{-value} < 0,05$ . O teste de Wilcoxon aplicado ao experimento 1 resultou em um  $p\text{-value}$  de 0,000004541. O mesmo teste resultou no  $p\text{-value}$  de 0,000003035 para o experimento 2 e no  $p\text{-value}$  de 0,01856 para o experimento 3.

Com base no resultado do teste de Wilcoxon, podemos afirmar que a hipótese nula pode ser rejeitada e a hipótese alternativa de que o ambiente *Heráclito* contribui para o aumento do percentual de acertos dos alunos pode ser aceita, ou seja, é uma boa evidência que o uso do *Heráclito* tem um impacto positivo na aprendizagem da Lógica Proposicional.

Para fins de avaliação da usabilidade do ambiente do ponto de vista dos alunos, também foi disponibilizado um questionário de avaliação da ferramenta ao final dos experimentos 1 e 2 para análise das percepções dos alunos e sugestões de melhoria. O questionário foi construído baseado na metodologia TAM2 (*Technology Acceptance Model* version 2) (Marangunić; Granić, 2015), considerando os indicadores de percepção de utilidade, no caso utilidade na aprendizagem de Lógica Proposicional, e facilidade de uso da interface do ambiente.

O questionário A, direcionado para a avaliação da ferramenta de resolução de provas de argumentos, teve a participação de 12 dos alunos que participaram previamente do experimento educacional 1, enquanto que o questionário B, elaborado

para avaliar a ferramenta de resolução de exercícios com fórmula, teve a participação de 19 dos alunos que participaram previamente do experimento educacional 2. O questionário A foi formado por 32 frases afirmativas, enquanto que o questionário B contou com 22 frases afirmativas. Em ambos questionários foi disponibilizado campos de texto de preenchimento livre para sugestões e críticas. As questões foram divididas em duas categorias para avaliação da utilidade no aprendizado e duas categorias para avaliar a facilidade de uso da interface de usuário. A medição do nível de concordância do aluno foi obtida com a escala Likert. A Tabela 4 mostra um sumário dos resultados dos questionários de usabilidade.

**Tabela 4:** Resultados dos questionários de usabilidade

Questionário	Aceitação	Utilidade no Aprendizado	Facilidade de Uso
A	<i>Concordo</i>	42,0%	50,0%
	<i>Concordo Parcialmente</i>	33,6%	21,6%
	<i>Indiferente</i>	11,9%	18,4%
	<i>Discordo Parcialmente</i>	6,3%	7,9%
	<i>Discordo</i>	6,3%	2,1%
B	<i>Concordo</i>	62,3%	56,6%
	<i>Concordo Parcialmente</i>	24,4%	20,6%
	<i>Indiferente</i>	8,3%	18,8%
	<i>Discordo Parcialmente</i>	2,3%	3,4%
	<i>Discordo</i>	2,7%	0,6%

No caso da utilidade das ferramentas disponíveis no *Heráclito*, o nível de concordância somado tanto do questionário A de 75,6% quanto do questionário B de 86,7%, ambos maiores que 75%, indicam uma boa aceitação dessas ferramentas no que tange a contribuição no aprendizado. O fato de tanto os resultados dos experimentos educacionais, quanto a avaliação da percepção do alunos da utilidade do *Heráclito* estarem de acordo é uma ótima indicação que este ambiente de ensino está na direção correta.

Por outro lado, o questionário A indicou possibilidade de melhora na interface de usuário da ferramenta de resolução de exercícios de provas de argumento. O nível de concordância somado de 71,6% é um bom valor de aceitação, mas tanto o nível de indiferença, quanto informações adicionais fornecidas nos campos de livre preenchimento mostraram possibilidades de melhoras nessa interface. A ferramenta de resolução de exercícios de fórmulas se saiu melhor nesta avaliação, com um nível de concordância somado de 77,2% para o questionário B. Porém, da mesma forma que no experimento A, foram detectadas possibilidades de melhora na interface da ferramenta, com os resultados do experimento.

## 7. Conclusões

O ambiente *Heráclito* apresentado neste trabalho visa ajudar os estudantes a aprender conceitos da Lógica Proposicional. Experimentos empíricos conduzidos com este ambiente mostram que a possibilidade de ajuda é real e que, além disso, o ambiente está sendo bem aceito pelos alunos. É claro que isso não sinaliza o fim da pesquisa, muito antes pelo contrário, abre todo um novo conjunto de possibilidades de pesquisa que se espera poder conduzir com este ambiente. Mas antes disso se espera poder contribuir com o avanço no ensino da Lógica pela disponibilização do ambiente *Heráclito* à comunidade. Assim, após os resultados dos experimentos, o ambiente entrou em uso experimental e pode ser acessado e usado por qualquer estudante no endereço <http://obaa.unisinos.br/heraclito/>. Em termos de novas pesquisas várias possibilidades estão sendo estudadas, sendo uma das mais importantes a evolução do modelo de aluno. Atualmente o modelo de aluno do *Heráclito* está centrado nas várias formas possíveis de

exercícios da Lógica Proposicional, faltando uma visão geral do processo de aprendizagem do aluno em relação aos conceitos por trás de cada tipo de exercícios e da posição destes conceitos na estrutura curricular da disciplina. Um modelo bayesiano capaz de integrar os a avaliação dos exercícios aos conceitos curriculares da disciplina está sendo pesquisado.

## References

- Andrade, J., Canese, M. (2013) Um Sistema Web Gamificado para a Aprendizagem de Logica Formal. *Anais do CBIE2013*, Campinas.
- Alonso, J.A., Aranda, G.A., Martin-Mateos, F. J. (2007) KRRT: Knowledge Representation and Reasoning Tutor System. *Procs of EUROCAST-2007*.
- Barnes, T., Stamper, J. (2010) Automatic Hint Generation for Logic Proof Tutoring Using Historical Data. In *Educational Technology & Society*, 13 (1), pg. 3–12.
- Baronett, S. (2008) *Logic*. Pearson
- Benzmuller, C., Horacek, H., Kruijff-Korbayov, I., Pinkal, M., Siekmann, J., Wolska, M. (2007) Natural language dialog with a tutor system for mathematical proofs. In Ullrich, C., Siekmann, J.H., Lu, R. (eds.) *Cognitive Systems*. LNCS (LNAI), v. 4429.
- Costa, E., Silva, P., Silva, M., Silva, E., Santos, A. (2012) A Multiagent-Based ITS Using Multiple Viewpoints for Propositional Logic. *Procs of ITS 2012*, Chania, Crete. LNCS. v. 7315. New York: Springer, pg. 640-641.
- Diaz, R., Neal, C., Amaya-Williams, M. (1992) The Social Origin of Self-regulation. In Moll, L.C (Comp.), *Vygotsky and Education: Instructional Implications and Applications of Sociocultural Psychology*. Cambridge University Press.
- Gersting, A. L. (2012) *Fundamentos Matemáticos para a Ciência da Computação: um tratamento moderno de matemática discreta*. 5. ed. Rio de Janeiro: LTC.
- Gluz, J. C., Penteado, F., Mossmann, M., Gomes, L., Vicari, R.M. (2014) A Student Model for Teaching Natural Deduction Based on a Prover that Mimics Student Reasoning. *Procs. of ITS 2014*, Haway. LNCS, v. 8474, p. 482-489
- Lukins, S., Levicki, A., Burg, J. (2002) A tutorial program for propositional logic with human/computer interactive learning. *SIGCSE Bull.* 34, 1, Feb. 2002, pg. 381-385.
- Malhotra, R. (2016) *Empirical Research In Software Engineering: Concepts, Analysis And Applications*. CRC Press. 1. ed. 486 p.
- Marangunić, N., Granić, A. (2015) Technology Acceptance Model: A Literature Review from 1986 to 2013. *J. of Univers. Access Inf. Soc.* v. 14, n. 1, p. 81-95.
- Nolt, J., Rohatyn, D., Varzi, A. (1998) *Schaum's Outline of Logic*. McGraw Hill.
- Passerino, L. M., Santarosa, L. (2008) Autism and Digital Learning Environments: processes of interaction and mediation. *Computers and Education*, v.51, pg. 385-402.
- Penteado, F., Gluz, J.C. (2011) Sistema Heráclito: Suporte a Objetos de Aprendizagem Interativos e Dialéticos Voltados ao Ensino de Dedução Natural na Lógica Proposicional. *Anais do SBIE 2011*, Aracaju. v. 1. p. 244-253.
- Stamper, J, Barnes, T., Croy, M. (2010) Enhancing the Automatic Generation of Hints with Expert Seeding. *Proceedings of ITS 2010*, Pittsburgh. LNCS. v. 6095. New York: Springer, pg. 31-40.
- Tomasello, M. (2001). *The Cultural Origins of Human Cognition*. Harvard Univ. Press.
- Vygotsky, L. S. (1986). *Thought and Language*. The M.I.T. Press, Cambridge, MA.
- Yacef, K. (2005) The Logic-ITA in the classroom: a medium scale experiment. In *International Journal of Artificial Intelligence in Education*, v.15(1).