

LFAweb como Ambiente Unificado para o Ensino de Linguagens Formais e Autômatos

Willian Souza, Ricardo Terra

Departamento de Ciência da Computação – Universidade Federal de Lavras (UFLA)
Lavras – MG – Brasil

`willian.souza2@estudante.ufla.br, terra@ufla.br`

Abstract. *The teaching and learning of Formal Languages and Automata (FLA) pose challenges because of the theoretical, abstract, and mathematical nature of the subject. This article presents the evolution of LFAweb as a web platform for supporting FLA education, with emphasis on the expansion of its simulators and the incorporation of an exercise module with automatic validation based on predefined test cases. The platform provides an integrated workflow for constructing and simulating models, solving exercises, receiving feedback, and reviewing errors. Input strings associated with failed test cases serve as witness strings and can be simulated step by step in the same environment. The study adopts an analytical and comparative approach involving LFAweb, JFLAP, and Automata Tutor across functional, pedagogical, and cognitive dimensions. The analysis suggests that the integration provided by LFAweb can support a more continuous workflow for teaching and learning FLA.*

Keywords: *Formal Languages and Automata; Educational Tools; Automatic Validation; Formative Feedback.*

Resumo. *A disciplina de Linguagens Formais e Autômatos (LFA) apresenta desafios de aprendizagem associados ao caráter teórico, abstrato e matemático de seus conteúdos. Este artigo apresenta a evolução do LFAweb como plataforma web de apoio ao ensino de LFA, com ênfase na ampliação dos simuladores e na incorporação de um módulo de exercícios com validação automática baseada em casos de teste previamente cadastrados. A plataforma integra construção de modelos, simulação, resolução de exercícios, feedback e revisão de erros, permitindo que as cadeias associadas aos casos de teste falhos sejam utilizadas como testemunhas e simuladas passo a passo no mesmo ambiente. O estudo adota uma abordagem analítica e comparativa envolvendo LFAweb, JFLAP e Automata Tutor, considerando as dimensões funcional, pedagógica e cognitiva. A análise sugere que a integração promovida pelo LFAweb pode apoiar um fluxo mais contínuo no ensino de LFA.*

Palavras-chave: *Linguagens Formais e Autômatos; Ferramentas Educacionais; Validação Automática; Feedback Formativo.*

1. Introdução

A disciplina de Linguagens Formais e Autômatos (LFA) possui papel fundamental nos cursos de Computação, pois aborda conceitos teóricos que sustentam áreas como Compiladores, Teoria da Computação e Inteligência Artificial [Chesñear et al. 2004,

Dognini and Raabe 2003]. Apesar de sua relevância, o ensino e a aprendizagem de LFA apresentam desafios recorrentes, principalmente devido ao caráter abstrato, matemático e formal de seus conteúdos [Vijayalaskhmi and Karibasappa 2012]. Nesse contexto, ferramentas computacionais têm sido utilizadas como recursos de apoio ao ensino, pois permitem representar visualmente modelos formais, simular seu comportamento e favorecer a compreensão de conceitos pouco intuitivos em abordagens exclusivamente expositivas.

Entre as ferramentas utilizadas no ensino de LFA, destacam-se JFLAP, Automata Tutor e LFAweb [Rodger and Finley 2006, D’Antoni et al. 2020, LFAweb 2026]. Estudos indicam que essas ferramentas apresentam diferenças de foco e abrangência, justificando seu uso complementar no apoio à disciplina [Paiva et al. 2023]. Entretanto, essa complementaridade também evidencia uma lacuna: a necessidade de alternar entre diferentes ambientes para construir modelos, simular palavras, resolver exercícios, validar soluções e revisar erros.

Essa fragmentação pode representar uma dificuldade para o discente, especialmente em uma disciplina que já exige elevado esforço cognitivo para compreender formalismos, símbolos, estados, transições e critérios de aceitação. Sob a perspectiva da Teoria da Carga Cognitiva, atividades que exigem esforço mental associado a elementos externos ao conteúdo principal podem aumentar a carga cognitiva extrínseca e dificultar a aprendizagem [Sweller 1988, Paas et al. 2003]. Em ambientes digitais e online, essa carga também pode ser influenciada por aspectos de interface, navegação, interatividade e organização da informação [Skulmowski and Xu 2022]. Assim, a integração de recursos em um mesmo ambiente pode favorecer a continuidade no processo de aprendizagem.

Diante desse cenário, este artigo apresenta a evolução do LFAweb como um ambiente unificado para o apoio ao ensino de LFA. O trabalho parte da lacuna identificada por Paiva et al. [Paiva et al. 2023], segundo a qual JFLAP, Automata Tutor e LFAweb são utilizados de forma complementar para atender às diferentes demandas da disciplina.

Diferentemente daquele estudo, cujo foco esteve na análise comparativa dessas ferramentas, este trabalho apresenta a evolução do LFAweb a partir de funcionalidades desenvolvidas pelo primeiro autor, voltadas à ampliação do suporte à simulação, à resolução de exercícios, à validação automática e à revisão de erros.

Com base nesse objetivo, este trabalho busca responder à seguinte pergunta de pesquisa: *Quais benefícios funcionais, pedagógicos e cognitivos podem ser identificados na integração, em uma única plataforma, das funcionalidades de construção, simulação, resolução de exercícios, validação automática e revisão de erros oferecidas pelo LFAweb, em comparação com JFLAP e Automata Tutor?*

Este artigo está organizado da seguinte forma. A Seção 2 contextualiza as ferramentas de apoio ao ensino de LFA; a Seção 3 apresenta os conteúdos e requisitos pedagógicos utilizados para fundamentar a análise; a Seção 4 descreve a evolução do LFAweb e os recursos implementados; a Seção 5 apresenta a discussão dos resultados da análise; a Seção 6 apresenta as ameaças à validade; e a Seção 7 conclui o artigo.

2. Ferramentas de Apoio ao Ensino de LFA

Ferramentas educacionais voltadas ao ensino de LFA buscam aproximar definições matemáticas do comportamento operacional dos modelos formais. Para isso, possibilitam

a construção, simulação, transformação e validação de artefatos computacionais, favorecendo a compreensão de conceitos abstratos por meio de experiências visuais, interativas e exploratórias. Nesse contexto, destacam-se JFLAP, Automata Tutor e LFAweb, cuja utilização complementar foi proposta por Paiva et al. [Paiva et al. 2023].

JFLAP:¹ O JFLAP é uma das ferramentas mais consolidadas no ensino de LFA, oferecendo suporte à construção, visualização e simulação de autômatos finitos, autômatos com pilha, máquinas de Turing, gramáticas e transformações entre modelos [Rodger and Finley 2006]. Sua relevância pedagógica decorre da experimentação visual e interativa, que permite relacionar a especificação formal de um modelo ao seu comportamento durante a simulação [Rodger et al. 2009, Cavalcante et al. 2004]. Além disso, estudos sobre seu uso indicam contribuições para o engajamento dos estudantes e para a compreensão de conceitos da disciplina [Rodger et al. 2009]. No estudo de Paiva et al., o JFLAP é destacado principalmente por sua ampla cobertura de conteúdos e por seus recursos de simulação [Paiva et al. 2023].

Apesar dessa relevância, o JFLAP não se caracteriza, em seu fluxo principal, como uma plataforma unificada para gerenciamento sistemático de exercícios autocorrigidos, acompanhamento contínuo de desempenho e *feedback* formativo. Embora possa apoiar atividades avaliativas por meio de testes, verificação de equivalência de máquinas e materiais externos com sugestões de exercícios, a aplicação, a correção e a revisão sistemática de soluções não constituem o foco central da ferramenta. Essa limitação é reforçada por trabalhos que propõem mecanismos adicionais ao JFLAP para apoiar atividades de correção automática, como a verificação de submissões e a apresentação de contraexemplos quando a solução do estudante não satisfaz o comportamento esperado [Bezáková et al. 2022]. Ou seja, tais propostas indicam que, embora o JFLAP seja robusto para construção e simulação, funcionalidades voltadas à avaliação sistemática e ao *feedback* formativo demandam extensões ou recursos complementares ao fluxo tradicional da ferramenta.

Automata Tutor:² O Automata Tutor é uma plataforma direcionada à prática de exercícios com correção automática e geração de *feedback*. Seu foco está na submissão de soluções para problemas formais, permitindo que estudantes recebam retorno sobre suas respostas. A terceira versão ampliou sua cobertura, incluindo expressões regulares, gramáticas livres de contexto, autômatos com pilha, máquinas de Turing e conversões entre modelos formais [D’Antoni et al. 2020]. Essa característica torna a plataforma relevante em turmas numerosas, pois reduz o esforço de correção manual e permite oferecer retorno individualizado em larga escala. Além disso, estudos sobre *feedback* automático em autômatos destacam a importância de mecanismos capazes de orientar o estudante na construção de soluções formais, especialmente por meio de contraexemplos e indicações sobre erros cometidos [D’Antoni et al. 2015].

Apesar dessas contribuições, a proposta do Automata Tutor permanece organizada principalmente em torno de exercícios previamente definidos. Assim, embora a ferramenta ofereça suporte importante à prática e à correção automática, ela não se caracteriza prioritariamente como um ambiente de exploração livre e contínua, no qual construção,

¹<https://www.jflap.org/>

²<https://automata-tutor.live-lab.fi.muni.cz/home>

simulação, transformação, validação e revisão de modelos formais estejam integradas em um mesmo fluxo de aprendizagem.

LFAweb:³ O LFAweb é uma plataforma web voltada ao apoio ao ensino de LFA. Em Paiva et al., a ferramenta é associada à simulação de gramáticas e máquinas e a algoritmos de transformação, sendo recomendada de forma complementar ao JFLAP e ao Automata Tutor [Paiva et al. 2023]. A evolução apresentada neste trabalho incorpora construção, simulação, exercícios, validação automática e revisão em um mesmo ambiente, permitindo analisar o LFAweb como uma plataforma unificada.

Ao reunir recursos antes distribuídos entre diferentes ferramentas, o LFAweb responde à lacuna identificada no estudo anterior e favorece a continuidade entre construção, execução, validação e revisão de modelos formais. Seu diferencial está menos nas funcionalidades isoladas e mais em sua articulação em um único fluxo de interação e aprendizagem.

3. Conteúdos e Requisitos Pedagógicos de LFA

Esta seção delimita os conteúdos e requisitos pedagógicos de LFA considerados na análise do LFAweb. O objetivo não é avaliar a cobertura curricular completa da plataforma, mas fundamentar os critérios usados para discutir os recursos de simulação, exercícios, validação automática e revisão de erros.

O trabalho de Paiva et al. [Paiva et al. 2023] é adotado como referência por relacionar tópicos recorrentes da disciplina a ferramentas de apoio ao ensino de LFA. A partir desse estudo, este artigo concentra-se nos conteúdos diretamente mobilizados pelas funcionalidades analisadas: construção e simulação de modelos formais, resolução de exercícios envolvendo autômatos, gramáticas e expressões regulares, validação de respostas por casos de teste e revisão do comportamento produzido pelas soluções submetidas.

Esses conteúdos exigem que o estudante relacione definições formais, representações gráficas ou simbólicas e comportamento operacional dos modelos computacionais [Chesñevar et al. 2004, Ramos 2009]. Por isso, ferramentas educacionais nesse domínio devem apoiar a experimentação, a execução de palavras, a observação de estados e transições, a identificação de divergências entre a solução construída e o comportamento esperado e a revisão do raciocínio utilizado.

Nesse contexto, o *feedback* formativo é especialmente relevante, pois ajuda o estudante a compreender a distância entre sua resposta e o resultado esperado, revisar estratégias e corrigir equívocos conceituais [Black and Wiliam 1998, Hattie and Timperley 2007, Shute 2008]. Assim, no recorte deste artigo, os requisitos pedagógicos considerados incluem: continuidade entre construção e simulação; possibilidade de testar palavras; retorno específico sobre casos de teste que falharam; e uso dessas evidências para revisar a solução no mesmo ambiente.

4. LFAweb como Ambiente Unificado de Apoio ao Ensino de LFA

Esta seção apresenta a evolução do LFAweb a partir de dois recursos centrais: a simulação de modelos formais e o módulo de exercícios com validação automática. Após a descrição

³<https://lfaweb.dcc.ufla.br>

da metodologia, discute-se como esses recursos articulam construção, execução de palavras, identificação de erros e revisão das soluções no mesmo ambiente.

4.1. Metodologia

Para responder à pergunta de pesquisa, este trabalho adota uma abordagem aplicada, descritiva e analítica, voltada à identificação dos benefícios funcionais, pedagógicos e cognitivos associados à integração das funcionalidades do LFAweb, tomando JFLAP e Automata Tutor como referências comparativas. A avaliação realizada não é experimental com estudantes; portanto, seus resultados devem ser interpretados como uma análise qualitativa do potencial da plataforma, e não como medição empírica de aprendizagem, desempenho, engajamento ou usabilidade.

Os critérios de análise foram definidos a partir do estudo de Paiva et al. [Paiva et al. 2023], que caracteriza JFLAP, Automata Tutor e LFAweb como ferramentas com focos distintos no apoio ao ensino de LFA. Essa referência foi complementada pela literatura sobre *feedback* formativo e carga cognitiva, utilizada para interpretar implicações pedagógicas e cognitivas da integração de funcionalidades [Black and Wiliam 1998, Hattie and Timperley 2007, Shute 2008, Sweller 1988, Paas et al. 2003, Skulmowski and Xu 2022].

As ferramentas comparadas foram JFLAP, Automata Tutor e LFAweb, por serem aquelas analisadas no estudo que fundamenta a lacuna discutida neste artigo. A observação das funcionalidades disponíveis nas versões acessadas das ferramentas foi realizada em maio de 2026, considerando seus principais fluxos de uso. Em cada plataforma, verificou-se a presença ou ausência de recursos relacionados à construção e à simulação de modelos formais, à resolução de exercícios, à validação automática, à apresentação de *feedback* e à revisão das respostas, bem como a forma pela qual esses recursos se articulam no fluxo principal de uso. No LFAweb, também foram descritos o cadastro de exercícios, a validação por casos de teste e o uso das cadeias associadas aos testes falhos como apoio à revisão.

A comparação foi organizada nas dimensões funcional, pedagógica e cognitiva, cujos critérios e indicadores são apresentados na Tabela 1. Com base nesses critérios, a análise examina como as evoluções incorporadas ao LFAweb articulam simulação, exercícios, validação automática e revisão de erros. Os resultados da comparação são sintetizados e discutidos na Seção 5.

Tabela 1. Critérios utilizados na análise comparativa

Dimensão	Critério observado	Indicador observado na plataforma
Funcional	Integração entre construção, simulação, validação e revisão.	Fluxo de uso que permite construir, executar, validar e revisar no mesmo ambiente.
Pedagógica	<i>Feedback</i> acionável ao estudante.	Casos de teste falhos que podem ser analisados e simulados passo a passo.
Cognitiva	Continuidade da interação.	Redução da alternância entre ferramentas e manutenção do contexto da tarefa.

4.2. Simulação Integrada ao Fluxo de Aprendizagem

A simulação é um recurso central no ensino de LFA, pois permite aproximar definições formais do comportamento operacional dos modelos estudados. Ao executar uma palavra em um autômato ou máquina, o estudante pode observar estados, transições, configurações intermediárias e critérios de aceitação ou rejeição, tornando mais visível o funcionamento de estruturas normalmente apresentadas por meio de definições matemáticas.

A Figura 1 ilustra o uso da simulação no LFAweb como recurso de apoio à compreensão do comportamento de um modelo formal. A visualização da execução permite que o estudante relacione o artefato construído à sequência de transições percorrida durante o processamento da palavra.

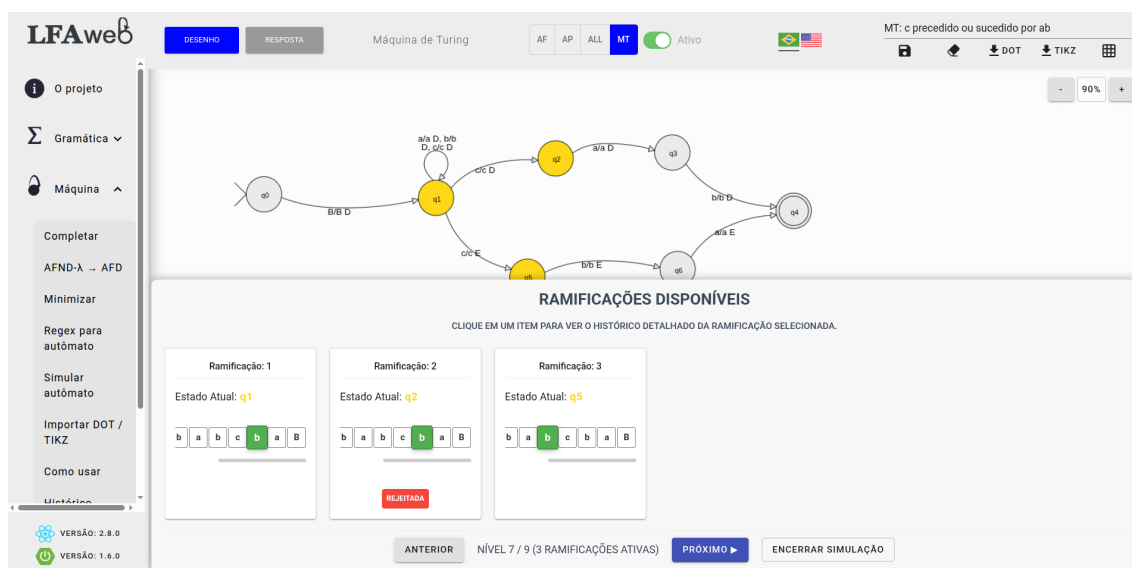


Figura 1. Simulação integrada no LFAweb, permitindo que o estudante acompanhe o comportamento operacional do modelo, visualize diferentes ramificações de execução e revise o processamento da palavra passo a passo no mesmo ambiente de construção.

Implementação no LFAweb: Em versões anteriores, o LFAweb já oferecia suporte à simulação de palavras em autômatos finitos, incluindo autômatos determinísticos, não determinísticos e com transições λ . A evolução da plataforma ampliou esse suporte, permitindo simulações em diferentes modelos formais, como autômatos com pilha, autômatos linearmente limitados e máquinas de Turing. Além disso, a plataforma passou a tratar computações não determinísticas por meio da visualização de diferentes caminhos de execução, quando existentes, aspecto relevante para a compreensão de que a aceitação de uma palavra depende da existência de pelo menos uma computação válida.

Durante a simulação, o LFAweb adota convenções específicas de implementação para cada modelo. Nos autômatos finitos, a aceitação ocorre pelo alcance de um estado final. Nos autômatos com pilha, a implementação exige o alcance de um estado final com a pilha vazia. Já nos autômatos linearmente limitados e nas máquinas de Turing, utiliza-se exclusivamente a aceitação por estado final, ainda não contemplando outras variações de aceitação descritas na literatura.

No LFAweb, a simulação está articulada à construção do modelo formal. O estudante pode desenhar o autômato ou máquina, informar uma palavra de entrada, acompanhar seu processamento e ajustar a construção no próprio ambiente. Desse modo, a plataforma não apenas informa se uma palavra é aceita ou rejeitada, mas permite observar como esse resultado é produzido, evidenciando estados, transições e configurações percorridas durante a execução. Além disso, mensagens de orientação durante a criação de transições auxiliam na redução de erros operacionais e direcionam a atenção do estudante ao conceito formal envolvido.

Comparação com ferramentas existentes: No JFLAP, a simulação ocupa papel central, permitindo construir modelos formais e executar entradas com visualização passo a passo das transições realizadas [Rodger and Finley 2006, Cavalcante et al. 2004, Rodger et al. 2009]. Essa característica contribui para sua ampla adoção no ensino de LFA, pois permite que o estudante relacione a representação gráfica do modelo ao comportamento produzido durante o processamento de uma palavra. Nesse aspecto, JFLAP e LFAweb apresentam papéis semelhantes, uma vez que ambos favorecem a experimentação visual e a observação do funcionamento operacional de autômatos e máquinas formais.

A diferença principal está no fluxo de interação oferecido ao estudante. No JFLAP, a construção do modelo normalmente exige alternância entre modos ou ferramentas específicas para criar estados, inserir transições, mover elementos e editar componentes. Essa organização é funcional e consolidada, mas pode acrescentar passos operacionais ao processo de construção e simulação. No LFAweb, por outro lado, a interação busca ser mais direta: o estudante constrói o modelo, informa a palavra de entrada, acompanha a execução e realiza ajustes no mesmo espaço de trabalho. Assim, a contribuição do LFAweb não está em substituir a simulação oferecida pelo JFLAP, mas em integrá-la a um fluxo mais contínuo de construção, teste e revisão no ambiente web.

O Automata Tutor, por sua vez, não tem como foco principal a exploração livre e contínua de modelos formais por meio de simulação, mas a resolução de problemas formais em um ambiente estruturado [D’Antoni et al. 2020]. Assim, no recorte específico desta subseção, sua contribuição é menos diretamente relacionada à simulação exploratória. Dessa forma, a comparação sobre simulação concentra-se principalmente entre JFLAP e LFAweb, uma vez que ambos permitem construir modelos formais e observar seu comportamento durante a execução de palavras.

4.3. Exercícios com Validação e Revisão de Erros

O módulo de exercícios representa outro componente essencial para consolidar o LFAweb como ambiente unificado. Enquanto a simulação favorece a observação do comportamento dos modelos formais, os exercícios permitem que o estudante pratique a construção de soluções, teste hipóteses e receba *feedback* sobre sua resposta. Em LFA, essa prática é indispensável, pois a aprendizagem envolve a aplicação de conceitos formais em diferentes representações, como autômatos, gramáticas e expressões regulares. A Figura 2 exemplifica o fluxo de resolução de exercícios no LFAweb. O estudante constrói sua solução no próprio ambiente, submete a resposta para validação automática e recebe informações que auxiliam a revisão do modelo formal produzido.

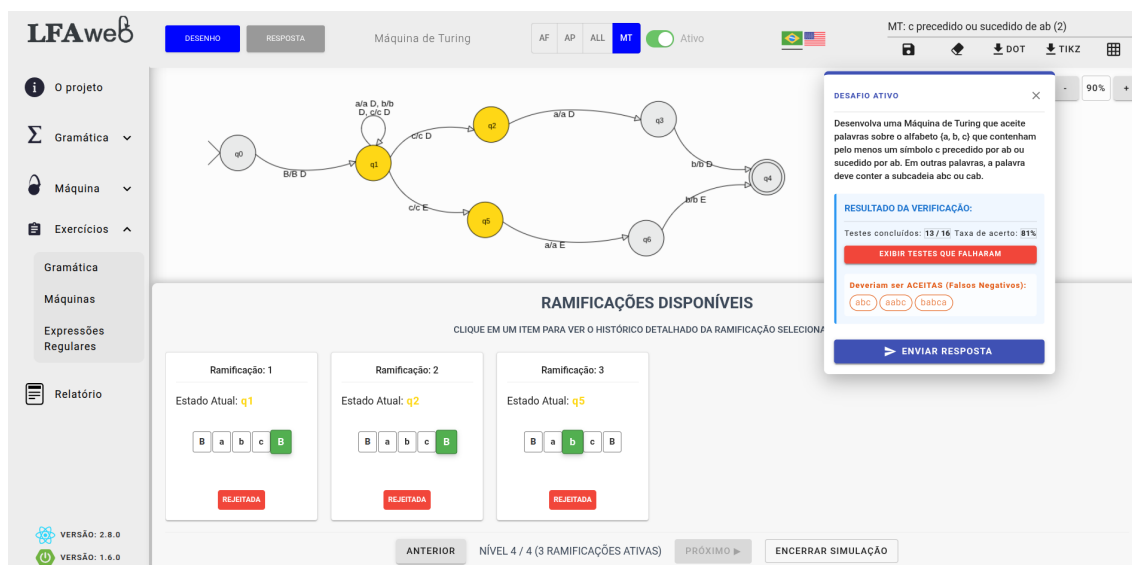


Figura 2. Fluxo de exercício no LFAweb, integrando construção da solução, validação automática, apresentação de cadeias testemunhas e simulação passo a passo para apoiar a revisão dos erros.

Implementação no LFAweb: O módulo de exercícios do LFAweb permite que estudantes pratiquem a construção de autômatos, gramáticas e expressões regulares diretamente na plataforma. O fluxo de utilização é dividido entre o cadastro dos exercícios e sua posterior resolução pelos estudantes.

O cadastro é realizado por usuários com perfil *Admin*, como professores e monitores autorizados. Atualmente, o sistema contempla o cadastro individual de exercícios, não dispondo de um mecanismo para criação e compartilhamento de listas de exercícios. Durante esse processo, o responsável define o enunciado, o formalismo abordado e um conjunto de palavras, cada uma associada ao resultado esperado de aceitação ou rejeição. Essas palavras constituem os casos de teste utilizados posteriormente na validação automática das respostas submetidas pelos estudantes.

Os exercícios são organizados de acordo com o formalismo tratado pela plataforma. Para máquinas, os filtros contemplam autômatos finitos, autômatos com pilha, autômatos linearmente limitados e máquinas de Turing. Para gramáticas, os exercícios são classificados segundo os quatro níveis da Hierarquia de Chomsky: gramáticas regulares, livres de contexto, sensíveis ao contexto e irrestritas.

Após selecionar um exercício, o estudante desenvolve sua solução no próprio ambiente e a submete para validação automática. O resultado da correção apresenta uma nota e informa o comportamento observado para cada caso de teste, indicando palavras que deveriam ser aceitas e foram rejeitadas, bem como palavras que deveriam ser rejeitadas e foram aceitas.

O módulo utiliza casos de teste previamente cadastrados que funcionam como cadeias testemunhas da divergência observada, permitindo que o estudante simule, no mesmo ambiente, as entradas em que sua solução não produziu o comportamento esperado. Essas cadeias vêm exclusivamente do conjunto de testes definido no cadastro do exercício; portanto, não provam equivalência entre modelos, não substituem verificação

formal e não devem ser interpretadas como garantia de correção completa da linguagem reconhecida ou gerada. O diferencial defendido neste artigo está na integração dessas cadeias à simulação passo a passo e à revisão da solução no próprio LFAweb.

Sob a perspectiva do docente, esse fluxo permite preparar previamente os exercícios, definir os casos de teste utilizados na validação automática e disponibilizar atividades aos estudantes de forma organizada. Dessa maneira, professores e monitores podem reduzir parte do esforço associado à correção inicial das atividades e concentrar o acompanhamento das turmas na discussão conceitual das dificuldades observadas.

Comparação com ferramentas existentes: Em relação aos exercícios, o JFLAP disponibiliza módulos e materiais online com exemplos e propostas de prática para diferentes tópicos de LFA [JFLAP 2016]. Contudo, esses materiais funcionam como recursos de apoio externos ao fluxo principal da aplicação. Assim, para realizar atividades de prática, validação e revisão, o estudante normalmente precisa articular enunciados, arquivos, simulações e formas de correção fora de um ambiente unificado.

Propostas como a de Bezaková et al. reforçam essa limitação ao estender o JFLAP com mecanismos de correção automática e apresentação de cadeias testemunhas para soluções incorretas [Bezaková et al. 2022]. Esse tipo de extensão evidencia a relevância pedagógica do *feedback* automático, mas também indica que tais recursos não constituem o foco original do fluxo tradicional da ferramenta.

O Automata Tutor, por outro lado, caracteriza-se pelo apoio à prática de exercícios com correção automática. A plataforma permite que estudantes submetam respostas e recebam *feedback*, sendo especialmente útil em contextos com grande número de alunos [D’Antoni et al. 2020]. Estudos sobre *feedback* automático em autômatos mostram que contraexemplos e orientações sobre erros podem auxiliar estudantes na construção de soluções formais [D’Antoni et al. 2015].

O diferencial do LFAweb está na articulação entre prática, validação automática, simulação e revisão da solução. Enquanto o Automata Tutor se organiza principalmente em torno da submissão e correção de exercícios, o LFAweb conecta a resposta incorreta a cadeias testemunhas que podem ser simuladas no próprio ambiente. Dessa forma, o estudante não apenas recebe uma indicação de erro, mas observa o comportamento da solução para as cadeias associadas aos casos de teste falhos e utiliza essas evidências para revisar sua construção.

5. Discussão

A Tabela 2 sintetiza os resultados da análise comparativa realizada a partir dos critérios apresentados na Tabela 1. A comparação considera o fluxo principal das ferramentas e concentra-se nos recursos relacionados à construção, simulação, resolução de exercícios, validação automática, *feedback* e revisão de erros. Extensões, materiais externos e funcionalidades não diretamente relacionadas ao recorte deste estudo não foram considerados.

A partir dessa síntese, os benefícios associados à integração promovida pelo LFAweb são discutidos nas dimensões funcional, pedagógica e cognitiva.

Dimensão funcional: O primeiro aspecto diz respeito à integração entre atividades que pertencem ao mesmo percurso de aprendizagem. O JFLAP se destaca pela construção

Tabela 2. Síntese dos resultados da análise comparativa entre as ferramentas

Aspecto	JFLAP	Automata Tutor	LFAweb
Foco principal	Construção e simulação visual de modelos formais.	Resolução de exercícios com correção automática e <i>feedback</i> .	Integração entre construção, simulação, exercícios, validação e revisão.
Construção e exploração livre	Oferece construção livre com ampla cobertura de modelos formais.	Não é o foco principal, pois o uso é orientado pela resolução de exercícios.	Oferece construção livre no mesmo ambiente utilizado para simulação e exercícios.
Simulação passo a passo	Disponível para vários modelos formais, incluindo visualização em computações não determinísticas.	Não constitui o foco principal da ferramenta.	Disponível, incluindo visualização em computações não determinísticas.
Exercícios integrados	Os materiais de exercícios não integram o fluxo principal da aplicação.	Oferece exercícios integrados à plataforma.	Oferece cadastro pelos docentes e resolução de exercícios diretamente na plataforma.
Mecanismo de validação	Não oferece correção sistemática de exercícios no fluxo principal.	Realiza correção automática conforme o tipo de exercício.	Executa casos de teste previamente cadastrados.
<i>Feedback</i> ao estudante	O resultado da simulação apoia a análise, mas o <i>feedback</i> formativo depende de recursos complementares.	Fornecer <i>feedback</i> automático sobre a solução submetida.	Apresenta nota e identifica os casos de teste em que o comportamento produzido diverge do esperado.
Revisão do erro	Exige a articulação entre a simulação e materiais externos de exercício ou correção.	Ocorre no contexto da correção do exercício, sem ter a simulação exploratória como foco principal.	As cadeias associadas aos casos de teste falhos podem ser simuladas passo a passo no mesmo ambiente.
Continuidade do fluxo	Integra construção e simulação, mas exercícios, correção e revisão não fazem parte do mesmo fluxo principal.	Integra exercícios e correção, mas a construção livre e a simulação exploratória não constituem seu foco principal.	Articula construção, simulação, exercícios, validação, <i>feedback</i> e revisão no recorte analisado.

e simulação visual de modelos formais, enquanto o Automata Tutor é voltado principalmente à prática de exercícios com correção automática [Rodger and Finley 2006, Rodger et al. 2009, D’Antoni et al. 2020]. O LFAweb não substitui essas ferramentas em todos os seus usos, mas reúne etapas que antes apareciam distribuídas entre elas. Assim, o estudante pode construir um modelo, testá-lo, resolver exercícios, receber indicações de erro e revisar sua solução sem romper o fluxo da atividade.

Dimensão pedagógica: A simples indicação de acerto ou erro é insuficiente para apoiar a aprendizagem de LFA, pois uma solução visualmente plausível pode reconhecer uma linguagem diferente da esperada. Nesse sentido, a integração entre validação automática, casos de teste falhos e simulação passo a passo fortalece o caráter formativo da plataforma. As cadeias testemunhas tornam explícitas divergências observadas no conjunto

de testes previamente cadastrado e aproximam o retorno oferecido pela ferramenta dos princípios de avaliação formativa [Black and Wiliam 1998, Hattie and Timperley 2007, Shute 2008]. Ademais, ao permitir que essas cadeias sejam simuladas no próprio modelo, o LFAweb transforma o erro em evidência analisável e reutilizável no processo de revisão.

Dimensão cognitiva: Em LFA, o estudante precisa relacionar a estrutura sintática dos modelos formais às linguagens que eles reconhecem ou geram. Quando construção, simulação e validação ocorrem em espaços separados, parte da atenção pode ser desviada para operações externas ao conteúdo, como mudança de interface, reorganização de arquivos ou reconstrução do contexto da tarefa. Sob a perspectiva da carga cognitiva, esse esforço adicional pode aumentar a carga extrínseca e dificultar a concentração no comportamento formal do modelo [Sweller 1988, Paas et al. 2003, Skulmowski and Xu 2022]. Ao manter essas etapas articuladas, o LFAweb favorece um fluxo mais contínuo de observação, teste e reformulação.

Limitação da validação: Como discutido na Seção 4.3, a validação por casos de teste não comprova equivalência formal nem garante a correção completa da solução, oferecendo evidências para orientar sua revisão [D’Antoni et al. 2015, Bezáková et al. 2022].

Implicações operacionais: O módulo permite que professores e monitores cadastrem enunciados, formalismos e casos de teste, podendo reduzir parte do esforço de correção inicial e favorecer a organização da prática dos estudantes.

Os resultados indicam que a contribuição do LFAweb não decorre de funcionalidades isoladamente inéditas, mas da articulação de recursos que, no cenário analisado, aparecem distribuídos entre JFLAP e Automata Tutor. Ao integrar construção, simulação, exercícios, validação automática, *feedback* e revisão em um mesmo fluxo, o LFAweb busca reduzir a fragmentação da interação. Essas vantagens não representam superioridade absoluta sobre as demais ferramentas, mas caracterizam o diferencial funcional, pedagógico e cognitivo identificado na análise.

6. Ameaças à Validade

Quanto à *validade de construto*, as dimensões funcional, pedagógica e cognitiva foram operacionalizadas por indicadores qualitativos que podem não contemplar todos os aspectos relevantes das ferramentas. Para reduzir essa ameaça, os critérios de análise foram definidos a partir da literatura sobre ferramentas de apoio ao ensino de LFA, *feedback* formativo e carga cognitiva.

Quanto à *validade interna*, existe potencial viés de interpretação, uma vez que o LFAweb é desenvolvido pelos autores. Como forma de mitigação, os mesmos critérios foram aplicados às três plataformas e a comparação foi restrita às funcionalidades observáveis em seus fluxos principais. Quanto à *validade externa*, os resultados limitam-se às ferramentas, versões e funcionalidades analisadas, podendo não se generalizar para outras plataformas ou contextos de uso. Por fim, a *validade das conclusões* é limitada pela ausência de avaliação com estudantes ou docentes, não sendo possível inferir efeitos sobre aprendizagem, usabilidade, engajamento ou carga cognitiva.

7. Conclusão

Este artigo apresentou a evolução do LFAweb como um ambiente unificado de apoio ao ensino de LFA. A partir da lacuna identificada por Paiva et al. [Paiva et al. 2023], analisou-se como a incorporação de novos recursos permite articular construção, simulação, exercícios, validação automática e revisão de erros em um mesmo ambiente.

A principal contribuição do trabalho está em discutir o LFAweb como alternativa de uso integrado para cenários didáticos nos quais o estudante precisa construir, testar, validar e revisar soluções formais. A centralização desses recursos pode reduzir a fragmentação do fluxo de aprendizagem e diminuir demandas operacionais associadas à alternância entre ferramentas. Sob a perspectiva cognitiva, essa organização pode favorecer a manutenção do contexto da tarefa durante a construção e a revisão das soluções. Sob a perspectiva pedagógica, a integração entre validação automática, cadeias testemunhas e simulação fortalece o caráter formativo da plataforma, pois permite que o erro seja tratado como evidência para análise e revisão. Assim, a questão de pesquisa é respondida pela identificação de benefícios funcionais, pedagógicos e cognitivos associados à integração dessas funcionalidades em um mesmo ambiente. Tais benefícios, contudo, resultam de uma análise qualitativa e comparativa, e não de uma avaliação empírica.

Como principal trabalho futuro, pretende-se avaliar empiricamente o LFAweb com estudantes e docentes em contextos reais de ensino, investigando seus efeitos sobre a aprendizagem, o engajamento, a usabilidade, a revisão de erros e a percepção de carga cognitiva. Em seguida, pretende-se ampliar o módulo de exercícios com recursos para criação e compartilhamento de listas, permitindo organizar atividades por tema, turma ou objetivo pedagógico, bem como classificá-las por nível de dificuldade.

Declaração sobre Uso de Inteligência Artificial

Os autores declaram que utilizaram ferramentas de Inteligência Artificial Generativa para apoio à revisão textual. Todo o conteúdo foi revisado e validado pelos autores, que assumem integral responsabilidade pelo trabalho.

Agradecimentos

Esta pesquisa é apoiada pelo CNPq.

Referências

- Bezáková, I., Fluet, K., Hemaspaandra, E., Miller, H., and Narváez, D. E. (2022). Effective succinct feedback for intro CS theory: A JFLAP extension. In *53rd Technical Symposium on Computer Science Education (SIGCSE)*, pages 976–982.
- Black, P. and Wiliam, D. (1998). Assessment and classroom learning. *Assessment in Education: Principles, Policy & Practice*, 5(1):7–74.
- Cavalcante, R., Finley, T., and Rodger, S. H. (2004). A visual and interactive automata theory course with JFLAP 4.0. In *35th Technical Symposium on Computer Science Education (SIGCSE)*, pages 140–144.
- Chesñevar, C. I., González, M. P., and Maguitman, A. G. (2004). Didactic strategies for promoting significant learning in formal languages and automata theory. *ACM SIGCSE Bulletin*, 36(3):7–11.

- D'Antoni, L., Helfrich, M., Kretinsky, J., Ramneantu, E., and Weininger, M. (2020). Automata Tutor v3. In *32nd International Conference on Computer Aided Verification (CAV)*, pages 3–14.
- D'Antoni, L., Kini, D., Alur, R., Gulwani, S., Viswanathan, M., and Hartmann, B. (2015). How can automatic feedback help students construct automata? *ACM Transactions on Computer-Human Interaction*, 22(2):1–24.
- Dognini, M. J. and Raabe, A. L. A. (2003). EduLing: Software educacional para linguagens regulares. In *XIV Simpósio Brasileiro de Informática na Educação (SBIE)*, pages 218–228.
- Hattie, J. and Timperley, H. (2007). The power of feedback. *Review of Educational Research*, 77(1):81–112.
- JFLAP (2016). JFLAP modules and exercises. <https://www.jflap.org/modules/>. Acesso em: 08 maio 2026.
- LFAweb (2026). LFAweb. <https://lfaweb.dcc.ufla.br/>. Acesso em: 08 maio 2026.
- Paas, F., Renkl, A., and Sweller, J. (2003). Cognitive load theory and instructional design: Recent developments. *Educational Psychologist*, 38(1):1–4.
- Paiva, P., Souza, M., and Terra, R. (2023). Ferramentas de apoio para a disciplina de linguagens formais e autômatos: uma proposta de uso. In *XXXIV Simpósio Brasileiro de Informática na Educação (SBIE)*, pages 1698–1709.
- Ramos, M. V. M. (2009). Ensino de linguagens formais e autômatos em cursos superiores de computação. *Revista de Computação e Tecnologia*, 1(1):22–34.
- Rodger, S. H. and Finley, T. W. (2006). *JFLAP: An Interactive Formal Languages and Automata Package*. Jones & Bartlett Learning, Burlington.
- Rodger, S. H., Wiebe, E., Lee, K. M., Morgan, C., Omar, K., and Su, J. (2009). Increasing engagement in automata theory with JFLAP. In *40th Technical Symposium on Computer Science Education (SIGCSE)*, pages 403–407.
- Shute, V. J. (2008). Focus on formative feedback. *Review of Educational Research*, 78(1):153–189.
- Skulmowski, A. and Xu, K. M. (2022). Understanding cognitive load in digital and online learning: A new perspective on extraneous cognitive load. *Educational Psychology Review*, 34(1):171–196.
- Sweller, J. (1988). Cognitive load during problem solving: Effects on learning. *Cognitive Science*, 12(2):257–285.
- Vijayalaskhmi, M. and Karibasappa, K. G. (2012). Activity based teaching learning in formal languages and automata theory: An experience. In *2nd International Conference on Engineering Education: Innovative Practices and Future Trends (AICERA)*, pages 1–5.