VINTE E UMA FALHAS EM VINTE E DOIS DIAS: UM ESTUDO DE CASO INSTRUMENTADO DE PESQUISA EM ENGENHARIA ELÉTRICA ASSISTIDA POR IA

Williams Fernandes Fontinele[1]

[1]*Universidade Federal do Acre, Rio Branco, Brasil (williams.fontinele@ufac.br)*

Resumo: Vinte e uma falhas registradas em 22 dias de produção de dois artigos de engenharia elétrica assistida por IA foram classificadas pelo mecanismo que as deteve. Nenhuma foi descoberta pela suíte de 629 testes automatizados: a detecção veio de revisão adversarial (8), confronto com fonte primária (5), execução sobre dado real (4), portão determinístico (2) e inspeção humana (2). Discutem-se os pontos em que a verificação automática é estruturalmente cega.

Palavras-chave: Pesquisa assistida por IA; Verificação; Modos de falha; Rastreabilidade; Engenharia elétrica

INTRODUÇÃO

A adoção de modelos de linguagem na produção científica deixou de ser hipótese e passou a ser mensurável em escala. Zhao et al. (2026b) auditaram 111 milhões de referências em 2,5 milhões de artigos depositados em arXiv, bioRxiv, SSRN e PubMed Central e estimam, de forma conservadora, 146.932 citações a trabalhos inexistentes apenas em 2025, concentradas nos campos de maior adoção e em manuscritos com assinatura linguística de escrita assistida. Zhao et al. (2026a) analisaram 17.443 citações geradas por quatro modelos sob cinco regimes de instrução e não encontraram nenhum cuja taxa de existência de citação superasse 0,475.

O problema não se limita ao texto. Rao e Callison-Burch (2026) construíram um conjunto de referência com 931 artigos em quatro domínios e mediram acurácia agregada de 83,6 % na geração de entradas bibliográficas, mas apenas 50,9 % de entradas completamente corretas, com erros simultâneos em vários campos; a corretude sobe para 78,3 % quando a arquitetura separa a busca inicial de uma revisão contra registro autoritativo. Em código, Shukla et al. (2025) submeteram 400 amostras a 40 rodadas de refinamento e observaram aumento de 37,6 % em vulnerabilidades críticas após apenas cinco iterações de melhoria - o refinamento degradou aquilo que deveria corrigir. Perry et al. (2023) já haviam mostrado, em estudo com usuários, que desenvolvedores assistidos por IA escrevem código menos seguro enquanto avaliam as próprias soluções como seguras.

Esses trabalhos medem o produto: a citação que não existe, o código vulnerável, a percepção equivocada. Sabe-se menos sobre o que ocorre dentro de um fluxo de trabalho instrumentado, isto é, quais falhas atravessam as verificações já em operação e quais são efetivamente detidas por elas. Gridach et al. (2025) situam confiabilidade e calibração entre os desafios abertos da IA agêntica na descoberta científica e apontam a colaboração humano-IA como direção, mas o repertório de modos de falha observados em ambiente real de produção permanece escasso.

Este trabalho relata 21 falhas registradas em 22 dias de operação de um motor de pesquisa em engenharia elétrica assistido por IA, durante a produção de dois artigos completos, e classifica cada uma pelo mecanismo que efetivamente a deteve. O objetivo é responder a uma pergunta operacional: entre as camadas de verificação disponíveis, quais capturam defeito de fato, e onde a verificação automática é estruturalmente cega.

MATERIAL E MÉTODOS

O ambiente estudado é um motor de pesquisa em engenharia elétrica operado por um agente de IA sob supervisão humana. Reúne 109 módulos em linguagem Python distribuídos por 17 domínios - entre eles circuitos, sistemas de potência, proteção, energia, máquinas elétricas e otimização -, cobertos por 629 testes automatizados. A produção de um artigo percorre oito fases: briefing, investigação, modelagem, execução, validação, redação, revisão

editorial e registro. Três portões de aprovação humana são obrigatórios: após o plano de pesquisa, após os resultados brutos e antes da versão final do manuscrito. Subagentes especializados atuam em papéis distintos - levantamento bibliográfico, revisão técnica, revisão editorial e redação - e a revisão é deliberadamente adversarial, com o revisor instruído a refutar, não a aprovar.

A instrumentação decorre de uma regra de projeto: toda falha de execução, hipótese incorreta ou retrabalho é registrada em formato fixo - data, contexto, causa-raiz, correção e prevenção - em arquivo versionado junto ao código. O registro é condição de encerramento de sessão, não item opcional, e os padrões que reincidem são promovidos a um documento de correções transversal aos projetos. Essa obrigatoriedade permite tratar o conjunto como corpus completo do período, e não como amostra selecionada por conveniência ou por memória.

Foram analisadas as 21 falhas registradas entre 3 e 24 de julho de 2026: 10 no próprio motor, 7 durante a produção de um artigo sobre dimensionamento de sistemas fotovoltaicos isolados para comunidades ribeirinhas e 4 durante um artigo sobre coordenação de proteção de sobrecorrente sob geração distribuída. Delas foram destilados sete padrões recorrentes.

Cada registro foi lido e atribuído à camada que primeiro expôs o defeito, conforme descrito no próprio campo de correção. Seis camadas foram distinguidas: suíte de testes automatizados, executada a cada alteração; portão determinístico, verificação programada que bloqueia o avanço de fase; execução sobre dado real, isto é, rodar o procedimento sobre material de produção em vez de entrada sintética, incluindo o experimento barato que testa a premissa antes de construir a solução; revisão adversarial, conduzida por subagente instruído a refutar; confronto com fonte primária, entendida como a norma técnica, o registro bibliográfico autoritativo, o dado oficial ou o próprio código-fonte; e inspeção humana direta. Em um caso o registro não nomeia o mecanismo, e a atribuição foi inferida do contexto.

RESULTADOS E DISCUSSÃO

A distribuição das 21 falhas pela camada que as deteve consta da Tabela 1.

Tabela 1. Falhas registradas por camada de detecção (3 a 24 de julho de 2026).

Camada que deteve a falha	Falhas
Revisão adversarial	8
Confronto com fonte primária	5
Execução sobre dado real	4
Portão determinístico	2
Inspeção humana direta	2
Suíte de testes automatizados	0
Total	21

Nenhuma das 21 falhas foi descoberta pelos 629 testes automatizados. O resultado não indica que a suíte seja dispensável: todas as 21 receberam teste de regressão após a correção, e é essa a função observada, impedir o retorno de um defeito já conhecido. O que a suíte não fez, em nenhum dos casos, foi revelar um defeito ainda desconhecido. A distinção importa porque a métrica usual de qualidade - a suíte verde - costuma ser lida como evidência de correção, quando é evidência de ausência de regressão conhecida.

Verificação sintática aprovada não implica conformidade semântica. Três casos ilustram o mecanismo. Ao formatar as citações de um manuscrito, aplicou-se a norma de referências onde vale a norma de citações e, além disso, a edição revogada desta última; o resultado foram doze citações na grafia errada. O portão determinístico aprovou, porque verifica a correlação entre a citação e a base bibliográfica, não a grafia; a revisão por subagente aprovou pelo mesmo motivo. O defeito foi detido pela leitura humana. Em outro caso, um teste verificava se o número de aberturas de ambiente no código gerado era o esperado, e a contagem passava mesmo havendo um ambiente órfão; o compilador, operando em modo de recuperação, ainda produzia o documento final, e o erro só apareceu ao inspecionar o registro de compilação. No terceiro, equações convertidas para o formato nativo do editor de texto produziram marcação válida e testes aprovados, mas exibiam retângulos de espaço reservado no lugar dos símbolos: foi preciso renderizar a página e olhar.

A correção pode ser pior do que o defeito. Um dos registros documenta uma cadeia em que a correção de um defeito de severidade alta introduziu um segundo defeito, e a primeira tentativa de corrigir este introduziu um terceiro, mais grave: onde antes havia interrupção visível, passou a haver remoção silenciosa de conteúdo dentro de uma região que o sistema garantia preservar. Trocar uma falha ruidosa

por uma silenciosa é regressão, ainda que o sintoma original desapareça. O padrão converge com a medição de Shukla et al. (2025): pedir a um modelo que melhore o próprio produto não é operação monotônica.

Alegação sobre o estado do mundo exige confronto com fonte primária. Cinco falhas foram detidas por essa camada, e a natureza delas é instrutiva. Em dois casos, subagentes afirmaram com confiança o estado de módulos do próprio repositório - um deles descrito como pronto para uso era estruturalmente incompatível com a tarefa; outro, dado como inexistente, existia - e ambas as afirmações sustentavam estimativas de esforço. Bastou abrir o arquivo. Em outro caso, um número de contexto obtido de fonte secundária atravessou cinco fases rotulado como pendente de confirmação primária; quando enfim confrontado com o dado oficial, estava uma ordem de grandeza acima. O rótulo de pendência não impediu a propagação: o que a impediu foi um portão que bloqueia texto contendo marcador não resolvido. Em um quarto caso, metadados bibliográficos só se revelaram incorretos ao serem confrontados com o registro autoritativo externo, que corrigiu autoria e paginação. Este último converge com o achado de Rao e Callison-Burch (2026): é a arquitetura que separa a geração da revisão contra registro autoritativo que eleva a corretude.

A camada mais produtiva também é de IA. Oito das 21 detecções vieram de revisão adversarial e, em sete delas, o revisor era um subagente. A leitura correta não é, portanto, que a IA erra e o humano corrige. O que distingue essa camada não é a natureza do revisor, mas o seu papel: independente de quem produziu e instruído a refutar. Em duas ocasiões foi necessário repetir a revisão sobre as próprias correções, e a segunda rodada encontrou defeitos que a primeira havia introduzido.

O papel humano é de baixa frequência e alta unicidade. A inspeção humana direta responde por 2 das 21 detecções, o menor número entre as camadas ativas. A relevância, contudo, não está no volume: nos dois casos, todas as demais camadas haviam aprovado. As citações erradas passaram por portão determinístico e por parecer de subagente; os símbolos ausentes nas equações passaram por validação de marcação e por testes. Ambos os defeitos eram visíveis a quem lesse o texto ou olhasse a página, e invisíveis a qualquer verificação que não fizesse exatamente isso. O humano não foi redundante com as máquinas: cobriu o que elas não alcançavam.

O padrão tem implicações para a formação. Sugere que a competência decisiva de quem pesquisa com assistência de IA não é redigir instruções melhores, mas projetar e operar camadas de verificação heterogêneas, sabendo a que cada uma é cega. Três disposições aparecem com frequência nos registros: exigir evidência de execução antes de aceitar uma afirmação sobre estado; confrontar toda alegação sobre o mundo com fonte primária; e preservar um ponto de inspeção humana sobre o artefato final, tal como ele será lido.

O estudo tem limitações. O corpus provém de um único ambiente, um operador e 22 dias, de modo que as frequências não devem ser lidas como taxas gerais. A classificação por camada foi feita pelo mesmo operador que registrou as falhas, o que introduz risco de viés, mitigado apenas pelo formato fixo do registro e por sua obrigatoriedade no momento do erro. Um caso teve atribuição inferida. Por fim, falhas não percebidas não podem, por definição, integrar o corpus: o número relatado é um piso, não um total.

CONCLUSÃO

Vinte e uma falhas registradas em 22 dias de pesquisa em engenharia elétrica assistida por IA foram classificadas pelo mecanismo que as deteve, e nenhuma delas foi descoberta pela suíte de 629 testes automatizados. A detecção coube à revisão adversarial (8), ao confronto com fonte primária (5), à execução sobre dado real (4), a portões determinísticos (2) e à inspeção humana direta (2). A suíte automatizada atuou como retenção de regressão, não como descoberta.

Três conclusões operacionais decorrem do exame. A primeira é que verificação aprovada não é conformidade: os defeitos mais persistentes atravessaram testes, portões e pareceres porque cada um verificava algo adjacente ao que estava errado. A segunda é que o refinamento iterativo não é monotônico, e que correção de severidade alta exige nova revisão sobre a própria correção. A terceira é que a verificação automática é estruturalmente cega ao que só se revela na leitura do texto e na inspeção do artefato renderizado - e é nesse ponto, de baixa frequência e alta unicidade, que a supervisão humana não é substituível.

A resiliência de um processo de pesquisa assistida por IA não decorre, portanto, da qualidade do modelo empregado, mas da heterogeneidade das camadas de verificação e do registro disciplinado das falhas que cada uma deixa passar.

REFERÊNCIAS

GRIDACH, M. et al. Agentic AI for scientific discovery: a survey of progress, challenges, and future directions. arXiv, 2025. DOI: 10.48550/arXiv.2503.08979.

PERRY, N. et al. Do users write more insecure code with AI assistants? In: ACM SIGSAC CONFERENCE ON COMPUTER AND COMMUNICATIONS SECURITY, 2023, Copenhague. Proceedings. New York: ACM, 2023. p. 2785-2799. DOI: 10.1145/3576915.3623157.

RAO, D.; CALLISON-BURCH, C. BibTeX citation hallucinations in scientific publishing agents: evaluation and mitigation. arXiv, 2026. DOI: 10.48550/arXiv.2604.03159.

SHUKLA, S.; JOSHI, H.; SYED, R. Security degradation in iterative AI code generation: a systematic analysis of the paradox. arXiv, 2025. DOI: 10.48550/arXiv.2506.11022.

ZHAO, C.; TANG, Y.; QIAN, Y. Do deployment constraints make LLMs hallucinate citations? An empirical study across four models and five prompting regimes. arXiv, 2026a. DOI: 10.48550/arXiv.2603.07287.

ZHAO, Z. et al. LLM hallucinations in the wild: large-scale evidence from non-existent citations. arXiv, 2026b. DOI: 10.48550/arXiv.2605.07723.