

Plectis

Research with AI, mathematics and software

WILL COOK · SEPTEMBER 2026

I built Plectis to pursue research I found interesting with AI, and to make a back-and-forth with human experts possible. I choose a direction, work at it and prepare the results for someone who can interpret them. Their feedback changes the next direction; I work on that and return, [recording their contribution and where later work builds on it](#)¹. An expert should be able to shape the research without having to operate the AI.

I have time and prompting experience, but limited mathematical knowledge. I chose mathematics because formal proofs can be checked, while interpretation and significance still require judgement. That lets me explore how far I can get, where I need help, and how those boundaries change as models improve. The name draws on *plectere*², to weave or twist: bringing different capabilities into the same work.

The [Navier–Stokes announcement](#)³ makes the question more immediate. If comparable capabilities reach inexpensive open models, output could exceed what experts can examine. [Tao and his co-signatories](#)⁴ distinguish solving problems from developing understanding. I think mathematics needs a floor for AI-assisted submissions: intelligible arguments, appropriate checks, clear limitations and attribution. Meeting it should make engagement easier, not create an entitlement to review. Experts can favour well-prepared work, giving producers an incentive to reduce the burden they impose.

The [mathematics repository](#)⁵ prototypes that floor around eight deliberately difficult Erdős problems. All remain open. Its [problem pages](#) join partial results, computations, Lean proofs and Comparator checks of selected statements. Short papers introduce the mathematics; longer records retain the reasoning for researchers and agents. For [#251](#), constructed sequences imitate features of prime gaps but give rational sums, exposing what a proof would need beyond those features. Preparing Palomar submissions also produced a [merged renderer fix](#)⁶. I have concentrated on exposition and interesting intermediate results rather than formalising everything.

Nineteenth-century mariners [contributed observations and received improved sailing charts](#)⁷. Each voyage benefited from what other crews had learnt. Plectis is designed around that cumulative approach. Failed routes remain alongside successful lemmas, with their conditions and limitations. Someone else, or a future model, may connect those observations differently, find another route or formulate a better question.

The [website](#) is the human wrapper. Its [mathematics map](#) connects questions, claims, papers and formal source; the [glossary](#) explains terminology. A downloadable [source brief](#) lets readers explore with their own assistant. Agents help prepare these views from the underlying work. I want increasing intelligence applied to making complexity manageable, not just producing more of it. The system is deliberately broad, which makes signal to noise the central problem; the front end is where I work on how presentation adapts to that breadth.

The [software repository](#)⁸ applies this beyond mathematics. Its [component pages](#) expose code, examples and checks; one [audit example](#) checks completion reports against actual commits and test runs in a sample repository. The [front-end recordings](#) show navigation from system maps into files and agent activity. The principle is that as AI raises what a back end can do, the front end has to rise with it, so that understanding and auditability scale with capability. Uncertainty about releasing the whole system led me to [publish testable parts instead](#).

The [safety doctrine](#) sets out axioms, principles and anti-principles; checks and permissions enforce particular rules. My [June questions](#) ask how model capability, operating environments and independent scrutiny interact. The [systems](#) and [open-source](#) papers explain how people can contribute direction, review, exposition, software or compute. The back-and-forth still needs testing with independent collaborators. My aim is for improving AI to multiply what an expert can direct and understand, rather than multiply the output waiting in their inbox.

SOURCES

- 1 Plectis, credit policy, docs/research-commons/CREDIT_POLICY.md in the mathematics repository.
https://github.com/wcook04/plectis-erdos/blob/main/docs/research-commons/CREDIT_POLICY.md
- 2 Corpus of Latin Dictionaries (BBAW), lemma *plecto*. <https://cld.bbaw.de/lemma/lat/plecto>
- 3 OpenAI, *On the Navier–Stokes Millennium Prize Problem*. <https://openai.com/index/navier-stokes-solution/>
- 4 T. Tao and co-signatories, *A severe misalignment of AI in mathematics*, 11 September 2026.
<https://terrytao.wordpress.com/2026/09/11/a-severe-misalignment-of-ai-in-mathematics/>
- 5 Plectis mathematics repository. <https://github.com/wcook04/plectis-erdos>
- 6 PalomarSubmission, pull request 137. <https://github.com/PalomarRegistry/PalomarSubmission/pull/137>
- 7 *The Origin and Work of the Division of Marine Meteorology*, Hydrographic Office, U.S. Naval Institute Proceedings 19/3, July 1893.
<https://www.usni.org/magazines/proceedings/1893/july/origin-and-work-division-marine-meteorology-hydrographic-office>
- 8 Plectis software repository. <https://github.com/wcook04/plectis>