# Large-Scale Formal Proof for the Working Mathematician—Lessons Learnt from the ALEXANDRIA Project

Lawrence C. Paulson(✉)

Computer Laboratory, University of Cambridge, Cambridge, UK
lp15@cam.ac.uk
https://www.cl.cam.ac.uk/~lp15/

**Abstract.** ALEXANDRIA is an ERC-funded project that started in 2017, with the aim of bringing formal verification to mathematics. The past six years have seen great strides in the formalisation of mathematics and also in some relevant technologies, above all machine learning. Six years of intensive formalisation activity seem to show that even the most advanced results, drawing on multiple fields of mathematics, can be formalised using the tools available today.

**Keywords:** Isabelle · formalisation of mathematics · ALEXANDRIA project

## 1 Introduction

In the summer of 2017, the Newton Institute at Cambridge held a programme entitled *Big Proof* (BPR) "directed at the challenges of bringing proof technology into mainstream mathematical practice". It was held in recognition of the formalisations that had already been done (which were indeed big). The programme webpage[1] specifically lists the proofs of the Kepler conjecture [19], the odd order theorem [17] and the four colour theorem [16]. That summer also saw the start of my ERC project, ALEXANDRIA. *Big Proof* represented an acknowledgement that the formalisation of mathematics could no longer be ignored, but also an assertion that big problems remain to be solved. These included "novel pragmatic foundations" and large-scale "formal mathematical libraries" and "inference engines", and also the "curation" of formalised mathematical knowledge.

ALEXANDRIA was conceived in part to try to identify those big problems. By hiring professional mathematicians and asking them to formalise advanced mathematics, we would get a direct idea of the obstacles they faced. We would also try to refine our tools, extend our libraries and investigate other technologies. We would have only five years (extended to six due to COVID-19).

The need for formalisation had been stressed by Vladimir Voevodsky, a Fields medallist, who pointedly asked "And who would ensure that I did not forget something and did not make a mistake, if even the mistakes in much more simple

[1] https://www.newton.ac.uk/event/bpr/.

C. Dubois and M. Kerber (Eds.): CICM 2023, LNAI 14101, pp. 3–15, 2023.
https://doi.org/10.1007/978-3-031-42753-4_1

arguments take years to uncover?" [38]. He advocated a new sort of formalism, homotopy type theory, which was the subject of much excitement. However, the most impressive formalisations by that time had been done in Coq (four colour theorem, odd order theorem), HOL Light (Kepler conjecture and much else) or Isabelle/HOL (part of the Kepler proof, and more). Lean, a newcomer, was attracting a user community. Perhaps our project would shed light on the respective values of the available formalisms: calculus of constructions (Coq, Lean), higher-order logic or homotopy type theory. Voevodsky would never find out, due to his untimely death in September 2017.

Since that date, research into the formalisation of mathematics has plunged ahead. Kevin Buzzard, a number theorist at Imperial College London, followed some of the *Big Proof* talks online. This resulted in his adoption of Lean for his Xena Project, with the aim of attracting students to formalisation.[2] Xena has had a huge impact, but here I'd like to focus on the work done within ALEXANDRIA.

## 2 A Brief Prehistory of the Formalisation of Mathematics

Mathematics is a work of the imagination, and the struggle between intuition and rigour has gone on since classical times. Euclid's great contribution to Greek geometry was the unification of many separate schools through his system of axioms and postulates. Newton and Leibniz revolutionised mathematics, but the introduction of infinitesimals was problematical. During the 19th centuries, the "arithmetisation of analysis" carried out by Cauchy and Weierstrass replaced infinitesimals by rigorous $\epsilon$–$\delta$ arguments. (We would not get a consistent theory of infinitesimals until the 1960 s,s, under the banner of non-standard analysis.) Dedekind and Cantor promulgated a radical new understanding of sets and functions that turned out to be inconsistent until Zermelo came up with his axioms. It is notable that Zermelo set theory (which includes the axiom of choice but lacks Fraenkel's replacement axiom) is approximately equal in logical strength to higher-order logic.

Only axiomatic mathematics can be formalised. The first attempt was by Frege, whose work (contrary to common belief) was not significantly impacted by Russell's paradox [1]. Russell and Whitehead in their *Principia Mathematica* [40] wrote out the proofs of thousands of mathematical propositions in a detailed axiomatic form. The work of Bourbaki can also be seen as a kind of formalised mathematics. The philosopher Hao Wang wrote on the topic and also coded the first automatic theorem prover [39] for first-order logic, based on what we would now recognise as a tableau calculus.

This takes us to NG de Bruijn, who in 1968 created AUTOMATH [5], and to his student's formalisation [24] of Landau's *Foundations of Analysis* in 1977. This takes us to the birth of Mizar [18], in which a truly impressive amount of mathematics was formalised in a remarkably readable notation. More recent

[2] https://www.ma.imperial.ac.uk/~buzzard/xena/.

history—analysis in HOL Light, the four colour theorem in Coq, etc.—is presumably familiar to readers. But it is appropriate to close this section with a prescient remark by de Bruijn back in 1968:

> As to the question what part of mathematics can be written in AUTOMATH, it should first be remarked that we do not possess a workable definition of the word "mathematics". Quite often a mathematician jumps from his mathematical language into a kind of metalanguage, obtains results there, and uses these results in his original context. It seems to be very hard to create a single language in which such things can be done without any restriction [[4], p. 3].

And so we have two great scientific questions:

- **What sort of mathematics can be formalised?**
- **What sort of proofs can be formalised?**

We would investigate these questions—mostly in the context of Isabelle/HOL—by formalising as much mathematics as we could, covering as many different topics as possible. I expected to run into obstacles here and there, which would have to be recorded if they could not be overcome.

## 3 ALEXANDRIA: Warmup Formalisation Exercises

The ERC proposal called for hiring research mathematicians, who would bring their knowledge of mathematics as it was practised, along with their *inexperience* of Isabelle/HOL. Their role would be to formalise increasingly advanced mathematical material with the twin objectives of developing formalisation methodologies and identifying deficiencies that might be remedied by extending Isabelle/HOL somehow. The project started in September 2017. We hired Anthony Bordg and Angeliki Koutsoukou-Argyraki. A third postdoc was required to undertake any necessary Isabelle engineering, and Wenda Li was hired.

One of the tasks for the first year was simply to reorganise and consolidate the Isabelle/HOL analysis library, which had mostly been translated from HOL Light. But we were also supposed to conduct pilot studies. The team set to work enthusiastically, and already in the first year they created a number of impressive developments:

- *Irrational rapidly convergent series*, formalising a 2002 proof by J. Hančl [20]
- *Projective geometry*, including Hessenberg's theorem and Desargues's theorem
- The theory of *quantum computing* (which identified a significant error in one of the main early papers)
- *Quaternions*, *octonions* and several other small exercises
- Effectively counting *real and complex roots of polynomials*, and the Budan-Fourier theorem [30,31]
- The first formal proof that *every field contains an algebraically closed extension* [37]

Koutsoukou-Argyraki wrote up her reactions to Isabelle/HOL from the perspective of a mathematician in her paper "Formalising Mathematics —in Praxis" [25].

# 4 Advanced Formalisations

As noted above, Kevin Buzzard had taken an interest in formalisation through participation in *Big Proof*, and by 2019 had marshalled large numbers of enthusiastic students to formalise mathematics using Lean. He had also made trenchant criticisms of even the most impressive prior achievements: that most of it concerned simple objects such as finite groups, or was just 19th-century mathematics. Nobody seemed to be working with sophisticated objects. He expressed astonishment that Grothendieck schemes—fundamental objects in algebraic geometry and number theory—had not been formalised in any tool. His criticisms helped focus our attention on the need to tackle difficult, recent and deep mathematics. Team members proposed their own tasks, but we also contributed to one another's tasks, sometimes with the help of interns or students. We completed three notable projects during this middle period:

- *Irrationality and transcendence criteria for infinite series* [27], extending the Hančl work mentioned above with material from two more papers: Erdős–Straus [13] and Hančl–Rucki [21].
- *Ordinal partition theory* [9]: infinite forms of Ramsey's theorem, but for order types rather than cardinals. We formalised relatively papers by Erdős–Milner [14] and Larson [29], and as a preliminary, the Nash-Williams partition theorem [36]. These were deep results in the context of Zermelo–Fraenkel set theory, involving highly intricate inductive constructions. One of the papers contained so many errors as to necessitate publishing a second paper [15] with a substantially different proof. This material was difficult even for Erdős!
- *Grothendieck Schemes* [3]. Buzzard had formalised schemes in Lean [6] (three times), and even claimed that Isabelle was not up to the job due to its simple type system. We took the challenge and it was straightforward, following a new approach based on locales to manage the deep hierarchies of definitions.

We were aiming for a special issue devoted to formalisation in the journal *Experimental Mathematics*, and were delighted to see these projects take up three of the six papers ultimately accepted.

# 5 Seriously Deep Formalisation Projects

Inspired by the success of the previous projects—conducted under the difficult circumstances of COVID-19 lockdown—team members continued to propose theorems to formalise, and we continued to collaborate in small groups. By now we had the confidence to take on almost anything. There are too many projects to describe in full, so let's look at some of the highlights.

## 5.1 Szemerédi's Regularity Lemma and Roth's Theorem on Arithmetic Progressions

*Szemerédi's regularity lemma* is a fundamental result in extremal graph theory. It concerns a property called the *edge density* of two given sets of vertices $X$,

$Y \subseteq V(G)$, and a further property of $(X, Y)$ being an $\epsilon$-regular pair for any given $\epsilon > 0$. The lemma itself states that for a given $\epsilon > 0$ there exists some $M$ such that every graph has an $\epsilon$-regular partition of its vertex set into at most $M$ parts. Intuitively, $(X, Y)$ is an $\epsilon$-regular pair if the density of edges between various subsets $A \subseteq X$ and $B \subseteq Y$ is more or less the same for all possible $A$ and $B$; an $\epsilon$-regular partition enjoys that property for all but an insignificant number of pairs $(X, Y)$ of vertex sets taken from the partition. Intuitively then, the vertices of any graph can be partitioned into most $M$ parts such that the edges between the various parts are uniform in this sense.

We used Szemerédi's regularity lemma to prove *Roth's theorem on arithmetic progressions*, which states that every "sufficiently dense" set of natural numbers includes three elements of the form $k$, $k + d$, $k + 2d$.

We used a variety of source materials and discovered a good many significant infelicities in the definitions and proofs. These included confusion between $\subset$ and $\subseteq$ (which are often synonymous in combinatorics) and between a number of variants of the lemma statement. One minor claim was flatly incorrect. To make matters worse, the significance of these issues only became clear in the application of the regularity lemma to Roth's theorem. Much time was wasted, and yet the entire formalisation project [10] took under six months.[3] By a remarkable coincidence, a group based in the mathematics department at Cambridge formalised a slightly different version of Szemerédi's regularity lemma, using Lean, around the same time [8].

### 5.2 Additive Combinatorics

Let $A$ and $B$ be finite subsets of a given abelian group $(G, +)$, and define their *sumset* as

$$A + B = \{a + b : a \in A, b \in B\}.$$

Write $nA$ for the $n$-fold iterated sumset $A + \cdots + A$. *Additive combinatorics* concerns itself with such matters as the relationship between the cardinality of $A+B$ and other properties of $A$ and $B$. Angeliki proposed this field as the natural successor to the formalisation of Szemerédi's regularity lemma because it's fairly recent (many results are less than 50 years old) and significant (providing a route to Szemerédi's theorem, a much stronger version of the Roth result mentioned above).

Here's an overview of the results formalised, all within the 7-month period from April to November 2022:

- The *Plünnecke–Ruzsa inequality*: yields an upper bound on the *difference* set $mB - nB$
- *Khovanskii's theorem*: for any finite $A \subseteq G$, the cardinality of $nA$ grows like a polynomial for all sufficiently large $n$.

[3] An email from Angeliki proposing to prove Szemerédi's regularity lemma is dated 8 July 2021. The formalisation was done by 5 November; Roth, 28 December.

- The *Balog-Szemerédi-Gowers theorem* is a deep result bearing on Szemerédi's theorem. The formalisation combines additive combinatorics with extremal graph theory and probability [26].
- *Kneser's theorem* and the *Cauchy-Davenport theorem* yield lower bounds for the size of $A + B$.

These are highly significant results by leading mathematicians. They can all be found in Isabelle's *Archive of Formal Proofs* (AFP).[4]

### 5.3 Other Formalisation Projects

The members chose a variety of large and small projects with a variety of specific objectives:

- *Combinatorial structures.* This is the PhD project of Chelsea Edmonds, who has used Isabelle's locale system to formalise dozens of varieties of block designs, hypergraphs, graphs and the relationships among them [11]. Results proved include Fisher's inequality [12].
- *Number theory.* We have formalised several chapters of *Modular Functions and Dirichlet Series in Number Theory*, a graduate textbook by Tom M. Apostol.
- *Wetzel's problem* is a fascinating small example, due to Erdős, where the answer to a question concerning complex analysis depends on the truth or falsity of the continuum hypothesis. The formal proof illustrates analysis and axiomatic set theory smoothly combined into a single argument [33].
- *Turán's graph theorem* states a maximality property of Turán graphs. This was a Master's student project.

This is a partial list, especially as regards contributions from interns, students and other visitors.

### 5.4 On Legibility of Formal Proofs

A proof is an argument, based on logical reasoning from agreed assumptions, that convinces mathematicians that a claim is true. How then do we understand a computer proof? To follow the analogy strictly, a computer proof convinces computers that a claim is true. But computers, even in this age of clever chatbots, are not sentient. We need to convince mathematicians.

Of the early efforts at the formalisation of mathematics, only Mizar aimed for legibility. Even pre-computer formal proofs such as *Principia Mathematica* are unreadable. Isabelle's proof language (Isar) follows the Mizar tradition, as in the following example:

```
lemma deriv_sum_int:
  "deriv (λx. ∑i=0..n. real_of_int (c i) * x^i) x
       = (if n=0 then 0 else (∑i=0..n-1. of_int((i+1) * c(Suc i)) *
x^i))"
```

[4] https://www.isa-afp.org.

```
  (is "deriv ?f x = (if n=0 then 0 else ?g)")
proof -
  have "(?f has_real_derivative ?g) (at x)" if "n > 0"
  proof -
    have "(∑i = 0..n. i * x ^ (i - Suc 0) * (c i))
        = (∑i = 1..n. (real (i-1) + 1) * of_int (c i) * x ^ (i-1))"
      using that by (auto simp: sum.atLeast_Suc_atMost intro!: sum.cong)
    also have "... = sum ((λi. (real i + 1) * c (Suc i) * x^i) ∘ (λn.
n-1))
                         {1..Suc (n-1)}"
      using that by simp
    also have "... = ?g"
      by (simp flip: sum.atLeast_atMost_pred_shift [where m=0])
    finally have §: "(∑a = 0..n. a * x ^ (a - Suc 0) * (c a)) = ?g" .
    show ?thesis
      by (rule derivative_eq_intros § | simp)+
  qed
  then show ?thesis
    by (force intro: DERIV_imp_deriv)
qed
```

Only a little training is required to make some sense of this. The lemma claims that the derivative of a certain summation equals a certain other summation. The proof refers of the variables ?f and ?g, which are defined by the pattern provided in the lemma statement: ?f denotes the original summation, and we prove that ?g is its derivative. Within that proof we can see summations being manipulated through changes of variable. Since we can see these details of the reasoning, we have reasons to believe that the proof is indeed correct: we do not simply have to trust the computer.

Not all Isabelle proofs can be written in a structured style. Page-long formulas often arise when trying to verify program code, and sometimes just from expanding mathematical definitions. Then we must use the traditional tactic style: long sequences of proof commands. However, most mathematical proofs that humans can write go into the structured style with ease. We have aimed for maximum legibility in all our work.

## 6 Library Search and Machine Learning Experiments

The focus of this paper is achievements in the formalisation of mathematics, but the ALEXANDRIA proposal also called for investigating supporting technologies. The name of the project refers to the library of Alexandria, and Isabelle's AFP already has nearly 4 million lines of proof text and well over 700 separate entries. How can we take advantage of all this material when developing new proofs?

In May 2019, the team acquired a new postdoc: Yiannos Stathopoulos. He came with the perfect background to tackle these objectives. After much labour, he and Angeliki produced the SErAPIS search engine,[5] which searches both the pre-installed Isabelle libraries and the AFP, offering a great many search strategies based on anything from simple keywords to abstract mathematical concepts [35]. It is not easy to determine the relevance or significance of a formal text to an abstract concept, but a variety of query types can be combined to explore the libraries.

Also mentioned in the proposal was the aim of Intelligent User Support. I had imagined that common patterns of proofs could be identified in the existing libraries and offered up to users, but with no idea how. To generate structured proofs automatically would require the ability to generate intermediate mathematical assertions. Six years of dramatic advances in machine learning have transformed our prospects. Language models can generate plausible texts given a corpus of existing texts. And as the texts we want would be inserted into Isabelle proofs, we can immediately check their correctness.

An enormous amount of work is underway, particularly by a student in our group, Albert Qiaochu Jiang, working alongside Wenda Li and others. It is now clear that language models can generate formal Isabelle proof skeletons [32] and can also be useful for identifying relevant lemmas [22]. We can even envisage *automatic formalisation* [23,41]: translating informal proofs into formal languages, by machine. Autoformalisation is easier with a legible proof language like ours, because the formal proof can have the same overall structure as the given natural language proof; a project currently underway is to develop the Isabelle Parallel Corpus, pairing natural language and Isabelle texts.[6] The next few years should see solid gains through machine learning.

## 7 Evaluation

At the start of this paper, I listed two scientific questions: what sort of mathematics, and what sort of proofs, can be formalised? And the answer so far is, everything we attempted, and we attempted a great variety of mathematical topics: number theory, combinatorics, analysis, set theory. The main difficulties have been errors and omissions in proofs. A vignette illustrates this point. Chelsea was formalising a probabilistic argument where the authors wrote "these probabilities are clearly independent, and therefore the joint probability is obtained by multiplying them." The problem is that this multiplication law is the mathematical definition of independent probabilities, which the authors had somehow confused with the real-world concept of unconnected random events. Frequently we have found proofs that are almost right: they need a bit of adjustment, but getting everything to fit takes effort.

Effort remains the main obstacle to the use of verification tools by mathematicians. Obvious claims are often tiresome to prove, which is both discouraging and a waste of an expert's time. But we might already advocate an approach of formalising the definitions and the proofs, stating the obvious claims without

[5] https://behemoth.cl.cam.ac.uk/search/.

[6] https://behemoth.cl.cam.ac.uk/ipc/.

proofs (using the keyword **sorry**). Even for this idea to be feasible, much more library material is needed, covering at least all the definitions a mathematician might expect to have available.

Another key scientific question is the role of dependent types. People in the type theory world seem to share the conviction that dependent types are necessary to formalise nontrivial mathematics. But in reality it seems to be Lean users who repeatedly fall foul of *intensional equality*: that $i = j$ does not guarantee that $T(i)$ is the same type as $T(j)$. Falling foul of this can be fatal: the first definition of schemes had to be discarded for this reason. Intensional equality is adopted by almost all dependent type theories, including Coq and Agda: without it, type checking becomes undecidable. But with it, type dependence does not respect equality.

The main limitation of simple type theory is that axiomatic type classes are less powerful than they otherwise would be. Isabelle/HOL has type classes for groups, rings, topological spaces among much else, but they are not useful for defining the theories of groups, rings or topological spaces. Rather they allow us, for example, to define the quaternions, prove a dozen or so laws and immediately inherit entire libraries of algebraic and topological properties. Abstract groups, rings, etc., need to be declared with an explicit carrier set (logically, the same thing as a predicate) rather than using the corresponding type class. It's a small price to pay for a working equality relation.

Having said this, one must acknowledge the enormous progress made by the Lean community over roughly the same period, 2017–now. Lean users, inspired by Buzzard, have taken on hugely ambitious tasks. The most striking is probably the Liquid Tensor Experiment [7]: brand-new mathematics, by a Fields medallist (Peter Scholze) who was concerned about its correctness, formalised over about a year and a half. This one accomplishment, more than anything else, demonstrates that formalisation can already offer real value to professional mathematicians.

We have from time to time looked at type issues directly. De Vilhena [37] describes an interesting technique for defining the $n$-ary direct product of a finite list of groups, iterating the binary direct product; his trick to avoid type issues involves creating an isomorphism to a suitable type. However, here one could avoid type issues (and handle the infinite case) by defining the direct product of a family in its own right as opposed to piggybacking off of the binary product. Anthony Bordg has done a lot of work on the right way to express mathematics without dependent types [2,3]. Ongoing work, still unpublished, is exploring the potential of the *types-to-sets framework* [28] to allow a smooth transition between type-based and carrier-set based formalisations.

One can also compare formalisms in terms of their logical strength. Higher-order logic is somewhat weaker than Zermelo set theory, which is much weaker than ZFC, which in turn is much weaker than Tarski-Grothendieck set theory:

$$\mathrm{HOL} < \mathrm{Z} \ll \mathrm{ZF} \ll \mathrm{TG}$$

The Calculus of Inductive Constructions, which is the formalism of Lean and Coq, is roughly equivalent to TG. The advantage of a weaker formalism is better automation. The power of ZF set theory, when it is required, can be obtained simply by loading the corresponding library from the AFP [33]. It's highly likely that a similar library could be created for Tarski-Grothendieck. And yet, remarkably, everything we have tried to formalise, unless it refers explicitly to ZF, sits comfortably within HOL alone. Since HOL is essentially the formalism of *Principia Mathematica* [40], we can conclude that Whitehead and Russell were right all along.

The AFP entries contributed by the project authors are too many to list, but they can be consulted via the on-line author indices:

– Anthony Bordg
  https://www.isa-afp.org/authors/bordg/
– Chelsea Edmonds
  https://www.isa-afp.org/authors/edmonds/
– Angeliki Koutsoukou-Argyraki
  https://www.isa-afp.org/authors/argyraki/
– Wenda Li
  https://www.isa-afp.org/authors/li/
– Lawrence C. Paulson
  https://www.isa-afp.org/authors/paulson/

## 8 Conclusions

We set out to tackle serious mathematics with a combination of hope and trepidation. We were able to formalise everything we set out to formalise and were never forced to discard a development part way through. As Angeliki has pointed out, "we have formalised results by two Fields medalists (Roth and Gowers), an Abel prize winner (Szemerédi) and of course Erdős too!"

We've also seen impressive advances in search and language models to assist users in proof development. Although the effort required to formalise mathematical articles remains high, we can confidently predict that formalisation will be playing a significant role in mathematical research in the next few years.

**Acknowledgements.** This work was supported by the ERC Advanced Grant ALEXANDRIA (Project GA 742178). Chelsea Edmonds, Angeliki Koutsoukou-Argyraki and Wenda Li provided numerous helpful comments and suggestions.

For the purpose of open access, the author has applied a Creative Commons Attribution (CC BY) licence to any Author Accepted Manuscript version arising from this submission.

## References

1. Boolos, G.S.: Saving Frege from contradiction. In: Logic, Logic, and Logic, pp. 171–182. Harvard University Press (1998)
2. Bordg, A., Doña Mateo, A.: Encoding dependently-typed constructions into simple type theory. In: Proceedings of the 12th ACM SIGPLAN International Conference on Certified Programs and Proofs, CPP 2023, pp. 78–89. Association for Computing Machinery (2023). https://doi.org/10.1145/3573105.3575679

3. Bordg, A., Paulson, L., Li, W.: Simple type theory is not too simple: Grothendieck's schemes without dependent types. Exp. Math. **31**(2), 364–382 (2022). https://doi.org/10.1080/10586458.2022.2062073
4. de Bruijn, N.G.: AUTOMATH, a language for mathematics. Tech. Rep. 68-WSK-05, Technical University Eindhoven (Nov 1968)
5. de Bruijn, N.G.: The mathematical language AUTOMATH, its usage, and some of its extensions. In: Laudet, M. (ed.) Proceedings of the Symposium on Automatic Demonstration, pp. 29–61. Springer LNM 125, Versailles, France (Dec 1968)
6. Buzzard, K., Hughes, C., Lau, K., Livingston, A., Mir, R.F., Morrison, S.: Schemes in lean. Experim. Math. **31**(2), 355–363 (2022). https://doi.org/10.1080/10586458.2021.1983489
7. Castelvecchi, D.: Mathematicians welcome computer-assisted proof in 'grand unification' theory. Nature **595**, 18–19 (2021)
8. Dillies, Y., Mehta, B.: Formalizing Szemerédi's regularity lemma in Lean. In: Andronick, J., de Moura, L. (eds.) 13th International Conference on Interactive Theorem Proving, pp. 9:1–9:19. Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2022)
9. Džamonja, M., Koutsoukou-Argyraki, A., Paulson, L.C.: Formalising ordinal partition relations using Isabelle/HOL. Exp. Math. **31**(2), 383–400 (2022). https://doi.org/10.1080/10586458.2021.1980464
10. Edmonds, C., Koutsoukou-Argyraki, A., Paulson, L.C.: Formalising Szemerédi's regularity lemma and Roth's theorem on arithmetic progressions in Isabelle/HOL. J. Autom. Reasoning **67**(1) (2023), https://doi.org/10.1007/s10817-022-09650-2
11. Edmonds, C., Paulson, L.C.: A modular first formalisation of combinatorial design theory. In: Kamareddine, F., Sacerdoti Coen, C. (eds.) CICM 2021. LNCS (LNAI), vol. 12833, pp. 3–18. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-81097-9_1
12. Edmonds, C., Paulson, L.C.: Formalising Fisher's inequality: formal linear algebraic proof techniques in combinatorics. In: Andronick, J., de Moura, L. (eds.) 13th International Conference on Interactive Theorem Proving (ITP 2022), vol. 237, pp. 11:1–11:19. Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2022). https://doi.org/10.4230/LIPIcs.ITP.2022.11
13. Erdős, P., Straus, E.G.: On the irrationality of certain series. Pacific J. Math. 55(1), 85–92 (1974). https://doi.org/pjm/1102911140
14. Erdős, P., Milner, E.C.: A theorem in the partition calculus. Can. Math. Bull. **15**(4), 501–505 (1972). https://doi.org/10.4153/CMB-1972-088-1
15. Erdős, P., Milner, E.C.: A theorem in the partition calculus corrigendum. Can. Math. Bull. **17**(2), 305 (1974). https://doi.org/10.4153/CMB-1974-062-6
16. Gonthier, G.: The four colour theorem: engineering of a formal proof. In: Kapur, D. (ed.) ASCM 2007. LNCS (LNAI), vol. 5081, pp. 333–333. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-87827-8_28
17. Gonthier, G., et al.: A machine-checked proof of the odd order theorem. In: Blazy, S., Paulin-Mohring, C., Pichardie, D. (eds.) ITP 2013. LNCS, vol. 7998, pp. 163–179. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-39634-2_14
18. Grabowski, A., Korniłowicz, A., Naumowicz, A.: Four decades of Mizar. J. Autom. Reasoning **55**(3), 191–198 (Oct 2015). https://doi.org/10.1007/s10817-015-9345-1
19. Hales, T., et al.: A formal proof of the Kepler conjecture. Forum Math. Pi **5**, e2 (2017). https://doi.org/10.1017/fmp.2017.1
20. Hančl, J.: Irrational rapidly convergent series. Rendiconti del Seminario Matematico della Università di Padova **107**, 225–231 (2002). http://eudml.org/doc/108582

21. Hančl, J., Rucki, P.: The transcendence of certain infinite series. Rocky Mountain J. Math. **35**(2), 531–537 (2005). https://doi.org/10.1216/rmjm/1181069744
22. Jiang, A.Q., et al.: Thor: Wielding hammers to integrate language models and automated theorem provers. In: Neural Information Processing Systems (NeurIPS) (2022)
23. Jiang, A.Q., et al.: Draft, sketch, and prove: guiding formal theorem provers with informal proofs. In: Eleventh International Conference on Learning Representations (2023). https://openreview.net/forum?id=SMa9EAovKMC
24. Jutting, L.: Checking Landau's "Grundlagen" in the AUTOMATH System. Ph.D. thesis, Eindhoven University of Technology (1977). https://doi.org/10.6100/IR23183
25. Koutsoukou-Argyraki, A.: Formalising mathematics — in praxis; a mathematician's first experiences with Isabelle/HOL and the why and how of getting started. Jahresbericht der Deutschen Mathematiker-Vereinigung **123**(1), 3–26 (2021). https://doi.org/10.1365/s13291-020-00221-1
26. Koutsoukou-Argyraki, A., Bakšys, M., Edmonds, C.: A formalisation of the Balog-Szemerédi-Gowers theorem in Isabelle/HOL. In: 12th ACM SIGPLAN International Conference on Certified Programs and Proofs, CPP 2023, pp. 225–238. Association for Computing Machinery (2023). https://doi.org/10.1145/3573105.3575680
27. Koutsoukou-Argyraki, A., Li, W., Paulson, L.C.: Irrationality and transcendence criteria for infinite series in Isabelle/HOL. Exp. Math. **31**(2), 401–412 (2022)
28. Kunčar, O., Popescu, A.: From types to sets by local type definition in higher-order logic. J. Autom. Reasoning **62**(2), 237–260 (2019). https://doi.org/10.1007/s10817-018-9464-6
29. Larson, J.A.: A short proof of a partition theorem for the ordinal $\omega^\omega$. Annals Math. Logic **6**(2), 129–145 (1973). https://doi.org/10.1016/0003-4843(73)90006-5
30. Li, W., Paulson, L.C.: Counting polynomial roots in Isabelle/HOL: a formal proof of the Budan-Fourier theorem. In: 8th ACM SIGPLAN International Conference on Certified Programs and Proofs, CPP 2019, pp. 52–64. Association for Computing Machinery (2019). https://doi.org/10.1145/3293880.3294092
31. Li, W., Paulson, L.C.: Evaluating winding numbers and counting complex roots through Cauchy indices in Isabelle/HOL. J. Autom. Reasoning (Apr 2019). https://doi.org/10.1007/s10817-019-09521-3
32. Li, W., Yu, L., Wu, Y., Paulson, L.C.: Isarstep: a benchmark for high-level mathematical reasoning. In: 9th International Conference on Learning Representations, ICLR 2021. OpenReview.net (2021). https://openreview.net/forum?id=Pzj6fzU6wkj
33. Paulson, L.C.: Wetzel: formalisation of an undecidable problem linked to the continuum hypothesis. In: Intelligent Computer Mathematics: 15th International Conference, CICM 2022, pp. 92–106. Springer (2022). https://doi.org/10.1007/978-3-031-16681-5_6
34. Peltier, N., Sofronie-Stokkermans, V. (eds.): IJCAR 2020. LNCS (LNAI), vol. 12166. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-51074-9
35. Stathopoulos, Y., Koutsoukou-Argyraki, A., Paulson, L.: Developing a concept-oriented search engine for Isabelle based on natural language: technical challenges. In: 5th Conference on Artificial Intelligence and Theorem Proving (2020). http://aitp-conference.org/2020/abstract/paper_9.pdf
36. Todorčević, S.: Introduction to Ramsey Spaces. Princeton University Press (2010)

37. de Vilhena, P.E., Paulson, L.C.: Algebraically closed fields in Isabelle/HOL. In: Peltier and Sofronie-Stokkermans [34], pp. 204–220
38. Voevodsky, V.: The origins and motivations of univalent foundations. The Institute Letter, pp. 8–9 (Summer 2014). https://www.ias.edu/ideas/2014/voevodsky-origins
39. Wang, H.: Toward mechanical mathematics. IBM J. Res. Dev. **4**(1), 2–22 (1960)
40. Whitehead, A.N., Russell, B.: Principia Mathematica. Cambridge University Press (1962), paperback edition to *56, abridged from the 2nd edition (1927)
41. Wu, Y., et al.: Autoformalization with large language models. In: Neural Information Processing Systems (NeurIPS) (2022)