Some Notes About Automated Reasoning and Conjecturing

Josef Urban

Czech Technical University in Prague

November 27, 2025, Gothenburg

Reasoning, Intuition, Discovery, Conjectures

“C’est par la logique qu’on démontre, c’est par l’intuition qu’on invente.”
(It is by logic that we prove, but by intuition that we discover.)
Henri Poincaré, Mathematical Definitions and Education.

“Hypothesen sind Netze; nur der fängt, wer auswirft.”
(Hypotheses are nets: only he who casts will catch.)
Novalis, quoted by Popper – The Logic of Scientific Discovery

Leibniz's/Hilbert's/Russell's Dream: Let Us Calculate!

Solve all (math, physics, law, economics, society, ...) problems by
reduction to logic/computation

[Adapted from: *Logicomix: An Epic Search for Truth* by A. Doxiadis]

Induction/Learning vs Reasoning – Henri Poincaré

- Science and Method: Ideas about the interplay between correct deduction and induction/intuition
- *"And in demonstration itself logic is not all. The true mathematical reasoning is a real induction [...]"*
- I believe he was right: strong general reasoning engines have to combine deduction and induction (learning patterns from data, making conjectures, etc.)

Automated Theorem Proving (ATP)

- One of the oldest fields of AI (Simon and Newell, Davis)
- Try to infer conjecture C from axioms Ax: $Ax \vdash C$
- Systems: Vampire, E, SPASS, Prover9, Z3, CVC4, Satallax, iProver, ...
- Various search calculi (resolution, superposition, tableaux, inst-gen)
- more limited logics: SAT, QBF, SMT, UEQ, ... (DPLL, CDCL, ...)
- TP-motivated PLs: Prolog (*logic programming* - Hayes, Kowalski)
- Human-designed heuristics for pruning of the search space
- strongest methods: resolution (generalized modus ponens) on clauses:
- $\neg man(X) \vee mortal(X), man(socrates) \vdash mortal(socrates)$
- resolution/superposition (equational) provers generate inferences, looking for the contradiction (empty clause)
- tableaux, connection calculi
- instantiation-based/SMT – systematically add ground instances and use SAT solvers to check satisfiability
- combined approaches - SAT run often inside the ATP (generalized splitting, AVATAR)
- ML guiding methods - both symbolic (hints/templates) and statistical

The CADE ATP System Competition (CASC)

Higher-order Theorems	Zipperpir 2.0	*Satallax 3.4*	Satallax 3.5	Vampire 4.5	Leo-III 1.5	CVC4 1.8	LEO-II 1.7.0
Solved/500	424/500	323/500	319/500	299/500	287/500	194/500	112/500
Solutions	424 84%	323 64%	319 63%	299 59%	287 57%	194 38%	111 22%

Typed First-order Theorems +*-/	Vampire 4.5	*Vampire 4.4*	CVC4 1.8
Solved/250	191/250	190/250	187/250
Solutions	191 76%	190 76%	187 74%

First-order Theorems	Vampire 4.5	*Vampire 4.4*	Enigma 0.5.1	E 2.5	CSE_E 1.2	iProver 3.3	GKC 0.5.1	CVC4 1.8	Zipperpir 2.0	Etableau 0.2	Prover9 1109a	CSE 1.3	leanCo 2.2
Solved/500	429/500	416/500	401/500	351/500	316/500	312/500	289/500	275/500	237/500	162/500	146/500	124/500	111/5
Solutions	429 85%	416 83%	401 80%	351 70%	316 63%	312 62%	289 57%	275 55%	237 47%	162 32%	146 29%	124 24%	111 2

First-order Non-theorems	Vampire SAT-4.5	*Vampire SAT-4.4*	iProver SAT-3.3	CVC4 SAT-1.8	E FNT-2.5	PyRes 1.3
Solved/250	238/250	226/250	182/250	98/250	63/250	13/250
Solutions	238 95%	226 90%	182 72%	98 39%	63 25%	13 5%

Unit Equality CNF	E 2.5	Twee 2.2.1	*E 2.4*	Vampire 4.5	Etableau 0.2	GKC 0.5.1	iProver 3.3	lazyCoP 0.1
Solved/250	202/250	197/250	185/250	162/250	148/250	128/250	124/250	20/250
Solutions	202 80%	197 78%	185 74%	162 64%	148 59%	128 51%	124 49%	0 0%

Large Theory Batch Problems	MaLARea 0.9	E LTB-2.5	iProver LTB-3.3	Zipperpir LTB-2.0	Leo-III LTB-1.5	ATPBoost 1.0	GKC LTB-0.5.1	*Leo-III LTB-1.4*
Solved/10000	7054/10000	3393/10000	3164/10000	1699/10000	1413/10000	1237/10000	493/10000	134/10000
Solutions	7054 70%	3393 33%	3163 31%	1699 16%	1413 14%	1237 12%	493 4%	134 1%

Using First/Higher Order Automated Theorem Proving

- 1996: Bill McCune proof of Robbins conjecture (Robbins algebras are Boolean algebras)
- Robbins conjecture unsolved for 50 years by mathematicians like Tarski
- 2021: M. Kinyon, R. Veroff, Prover9: Weak AIM conjecture
- If Q is an Abelian Innner Mapping loop, then Q is nilpotent of class $\leqslant 3$.
- ATP has currently only limited use for proving new conjectures
- mainly in very specialized algebraic domains
- however ATP has become very useful in Interactive Theorem Proving
- a recent (2020) performance jump in higher-order ATP:
- Zipperposition, HO-Vampire, E-HO (J. Blanchette, A Bentkamp, P. Vukmirovic)

What is Formal Mathematics?

- Developed thanks to the Leibniz/Russell/Frege/Hilbert/... program
- Mathematics put on formal logic foundations (*symbolic computation*)
- ... which btw. led also to the rise of computers (Turing/Church, 1930s)
- Formal math (1950/60s): combine formal foundations and computers
- Proof assistants/Interactive theorem provers and their large libraries:
- De Bruijn, Milner, Trybulec, Boyer and Moore, Gordon, Huet, Paulson, ...
- Automath (1967), LCF, Mizar, NQTHM, HOL, Coq, Isabelle, ACL2, Lean
- Conceptually very simple:
- Write all your axioms and theorems so that computer understands them
- Write all your inference rules so that computer understands them
- Use the computer to check that your proofs follow the rules
- But in practice, it turns out not to be so simple
- Many approaches, still not mainstream, but big breakthroughs recently

The QED Manifesto – 1994

- *QED is the very tentative title of a project to build a computer system that effectively represents all important mathematical knowledge and techniques.*
- *The QED system will conform to the highest standards of mathematical rigor, including the use of strict formality in the internal representation of knowledge and the use of mechanical methods to check proofs of the correctness of all entries in the system.*
- *The QED project will be a major scientific undertaking requiring the cooperation and effort of hundreds of deep mathematical minds, considerable ingenuity by many computer scientists, and broad support and leadership from research agencies.*
- ....
- Never happened, but inspired a lot of development – *"QED Singularity"*

Bird's Eye View of ITP Systems by T. Hales

HOL Light

HOL Light has an exquisite minimal design. It has the smallest kernel of any system. John Harrison is the sole

Mizar

Once the clear front-runner, it now shows signs of age. Do not expect
to understand the inner workings of this system unless you have been

Coq

Coq is built of modular components on a foundation of dependent type theory. This system has grown one PhD thesis at a time.

Isabelle

Designed for use with multiple foundational architectures, Isabelle's early
development featured classical constructions in set theory. However,

Metamath

Does this really work? Defying expectations, Metamath seems to function
shockingly well for those who are happy to live without plumbing.

Lean

Lean is ambitious, and it will be massive. Do not be fooled by the name.
"*Construction area keep out*" signs are prominently posted on the perimeter fencing.

F. Wiedijk: Irrationality of $\sqrt{2}$ (informal text)

tiny proof from Hardy & Wright, texts collected by F. Wiedijk:

Theorem 43 (Pythagoras' theorem). $\sqrt{2}$ is irrational.
The traditional proof ascribed to Pythagoras runs as follows. If $\sqrt{2}$ is rational, then the equation

$$a^2 = 2b^2 \tag{4.3.1}$$

is soluble in integers a, b with $(a, b) = 1$. Hence a^2 is even, and therefore a is even. If $a = 2c$, then $4c^2 = 2b^2$, $2c^2 = b^2$, and b is also even, contrary to the hypothesis that $(a, b) = 1$. □

Irrationality of $\sqrt{2}$ in Isabelle/HOL

```
theorem sqrt2_not_rational:
  "sqrt (real 2) ∉ ℚ"
proof
  assume "sqrt (real 2) ∈ ℚ"
  then obtain m n :: nat where
    n_nonzero: "n ≠ 0" and sqrt_rat: "¦sqrt (real 2)¦ = real m / real n"
    and lowest_terms: "gcd m n = 1" ..
  from n_nonzero and sqrt_rat have "real m = ¦sqrt (real 2)¦ * real n" by simp
  then have "real (m²) = (sqrt (real 2))² * real (n²)"
    by (auto simp add: power2_eq_square)
  also have "(sqrt (real 2))² = real 2" by simp
  also have "... * real (m²) = real (2 * n²)" by simp
  finally have eq: "m² = 2 * n²" ..
  hence "2 dvd m²" ..
  with two_is_prime have dvd_m: "2 dvd m" by (rule prime_dvd_power_two)
  then obtain k where "m = 2 * k" ..
  with eq have "2 * n² = 2² * k²" by (auto simp add: power2_eq_square mult_ac)
  hence "n² = 2 * k²" by simp
  hence "2 dvd n²" ..
  with two_is_prime have "2 dvd n" by (rule prime_dvd_power_two)
  with dvd_m have "2 dvd gcd m n" by (rule gcd_greatest)
  with lowest_terms have "2 dvd 1" by simp
  thus False by arith
qed
```

Irrationality of $\sqrt{2}$ in Coq

```
Theorem irrational_sqrt_2: irrational (sqrt 2%nat).
intros p q H H0; case H.
apply (main_thm (Zabs_nat p)).
replace (Div2.double (q * q)) with (2 * (q * q));
 [idtac | unfold Div2.double; ring].
case (eq_nat_dec (Zabs_nat p * Zabs_nat p) (2 * (q * q))); auto; intros H1.
case (not_nm_INR _ _ H1); (repeat rewrite mult_INR).
rewrite <- (sqrt_def (INR 2)); auto with real.
rewrite H0; auto with real.
assert (q <> 0%R :> R); auto with real.
field; auto with real; case p; simpl; intros; ring.
Qed.
```

What Has Been Formalized? (2022)

top 100 of interesting theorems/proofs (Paul & Jack Abad, 1999),
tracked by Freek Wiedijk - `https://www.cs.ru.nl/~freek/100/`

1. $\sqrt{2} \notin \mathbb{Q}$
2. fundamental theorem of algebra
3. $|\mathbb{Q}| = \aleph_0$
4.  $\Rightarrow a^2 + b^2 = c^2$
5. $\pi(x) \sim \frac{x}{\ln x}$
6. Gödel's incompleteness theorem
7. $\left(\frac{p}{q}\right)\left(\frac{q}{p}\right) = (-1)^{\frac{p-1}{2}\frac{q-1}{2}}$
8. impossibility of trisecting the angle and doubling the cube

⋮

32. four color theorem
33. Fermat's last theorem

⋮

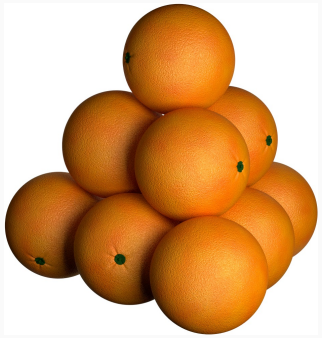
99. Buffon needle problem
100. Descartes rule of signs

all together	98%
HOL Light	86%
Mizar	69%
Isabelle	86%
Coq	78%
ProofPower	43%
Metamath	74%
Lean	70%
ACL2	18%
PVS	22%

Big Example: The Flyspeck project

- Kepler conjecture (1611): The most compact way of stacking balls of the same size in space is a pyramid.

$$V = \frac{\pi}{\sqrt{18}} \approx 74\%$$



- Proved by Hales & Ferguson in 1998, 300-page proof + computations
- Big: Annals of Mathematics gave up reviewing after 4 years
- Formal proof finished in 2014
- 20000 lemmas in geometry, analysis, graph theory
- All of it at `https://code.google.com/p/flyspeck/`
- All of it computer-understandable and verified in HOL Light:
- `polyhedron s /\ c face_of s ==> polyhedron c`
- However, this took 20 – 30 person-years!

Conjecturing in Mathematics

- Targeted: generate intermediate lemmas (cuts) for a harder conjecture
- Unrestricted (theory exploration):
- Creation of interesting conjectures based on the previous theory
- One of the most interesting activities mathematicians do (how?)
- Higher-level AI/reasoning task - can we learn it?
- If so, we have solved math:
- ... just (recursively) divide Fermat into many subtasks ...
- ... and conquer (I mean: hammer) them away

A Brief History of Automated Conjecturing

- 1970s–80s
  - Lenat's AM: rediscovered concepts (e.g., primes, UFD); "interestingness"
  - Langley's Bacon: automated scientific discovery (Kepler-like laws)
  - EURISKO: adaptive heuristics that modify their own search strategies
- 1980s
  - Fajtlowicz's Graffiti: graph theory conjecturing
  - many results later proven and influential
- 1990s–2000s
  - Colton's HR: concept invention + conjecturing; discoveries in integer sequences & algebraic relations
  - Integration with ATPs for filtering trivial/false conjectures
- 2010s - More Symbolic Connjecturing
  - Hipster: Isabelle-integrated exploration; lemmas for proof automation
  - QuickSpec ecosystem: scalable term generation, pruning & testing for specifications and lemma mining
- 2013+ Learning-Based/Neural Conjecturing
  - Learning from large formal libraries: analogy-based lemma prediction
  - Neural→formal conjecture pipelines: generation, disambiguation, pruning
  - Reinforcement feedback loops connecting conjecturing & proving
  - Informalization/Autoformalization systems creating conjectures

Karpathy'15 - RNN generating fake Math over Stacks

http://karpathy.github.io/2015/05/21/rnn-effectiveness/

For $\bigoplus_{n=1,\ldots,m}$ where $\mathcal{L}_{m_\bullet} = 0$, hence we can find a closed subset $\mathcal{H}$ in $\mathcal{H}$ and any sets $\mathcal{F}$ on X, U is a closed immersion of S, then $U \to T$ is a separated algebraic space.

Proof. Proof of (1). It also start we get

$$S = \mathrm{Spec}(R) = U \times_X U \times_X U$$

and the compariciy in the fibre product covering we have to prove the lemma generated by $\coprod Z \times_U U \to V$. Consider the maps M along the set of points Sch_{fppf} and $U \to U$ is the fibre category of S in U in Section, **??** and the fact that any U affine, see Morphisms, Lemma **??**. Hence we obtain a scheme S and any open subset $W \subset U$ in $Sh(G)$ such that $\mathrm{Spec}(R') \to S$ is smooth or an

$$U = \bigcup U_i \times_{S_i} U_i$$

which has a nonzero morphism we may assume that f_i is of finite presentation over S. We claim that $\mathcal{O}_{X,x}$ is a scheme where $x, x', s'' \in S'$ such that $\mathcal{O}_{X,x'} \to \mathcal{O}'_{X',x'}$ is separated. By Algebra, Lemma **??** we can define a map of complexes $\mathrm{GL}_{S'}(x'/S'')$ and we win. □

To prove study we see that $\mathcal{F}|_U$ is a covering of $\mathcal{X}'$, and $\mathcal{T}_i$ is an object of $\mathcal{F}_{X/S}$ for $i > 0$ and $\mathcal{F}_p$ exists and let $\mathcal{F}_i$ be a presheaf of $\mathcal{O}_X$-modules on $\mathcal{C}$ as a $\mathcal{F}$-module. In particular $\mathcal{F} = U/\mathcal{F}$ we have to show that

$$\widetilde{M}^\bullet = \mathcal{I}^\bullet \otimes_{\mathrm{Spec}(k)} \mathcal{O}_{S,s} - i_X^{-1}\mathcal{F})$$

is a unique morphism of algebraic stacks. Note that

$$\mathrm{Arrows} = (Sch/S)^{opp}_{fppf}, (Sch/S)_{fppf}$$

and

$$V = \Gamma(S, \mathcal{O}) \longmapsto (U, \mathrm{Spec}(A))$$

is an open subset of X. Thus U is affine. This is a continuous map of X is the inverse, the groupoid scheme S.

Proof. See discussion of sheaves of sets. □

The result for prove any open covering follows from the less of Example **??**. It may replace S by $X_{spaces,étale}$ which gives an open subspace of X and T equal to S_{Zar}, see Descent, Lemma **??**. Namely, by Lemma **??** we see that R is geometrically regular over S.

Lemma 0.1. *Assume (3) and (3) by the construction in the description.*

Suppose $X = \lim |X|$ *(by the formal open covering* X *and a single map* $\underline{Proj}_X(\mathcal{A}) = \mathrm{Spec}(B)$ *over* U *compatible with the complex*

$$Set(\mathcal{A}) = \Gamma(X, \mathcal{O}_{X,\mathcal{O}_X}).$$

When in this case of to show that $\mathcal{Q} \to \mathcal{C}_{Z/X}$ *is stable under the following result in the second conditions of (1), and (3). This finishes the proof. By Definition* **??** *(without element is when the closed subschemes are catenary. If* T *is surjective we may assume that* T *is connected with residue fields of* S*. Moreover there exists a closed subspace* $Z \subset X$ *of* X *where* U *in* X' *is proper (some defining as a closed subset of the uniqueness it suffices to check the fact that the following theorem*

(1) f *is locally of finite type. Since* $S = \mathrm{Spec}(R)$ *and* $Y = \mathrm{Spec}(R)$.

Proof. This is form all sheaves of sheaves on X. But given a scheme U and a surjective étale morphism $U \to X$. Let $U \cap U = \coprod_{i=1,\ldots,n} U_i$ be the scheme X over S at the schemes $X_i \to X$ and $U = \lim_i X_i$. □

The following lemma surjective restrocomposes of this implies that $\mathcal{F}_{x_0} = \mathcal{F}_{x_0} = \mathcal{F}_{\mathcal{X},\ldots,0}$.

Lemma 0.2. *Let* X *be a locally Noetherian scheme over* S, $E = \mathcal{F}_{X/S}$. *Set* $\mathcal{I} = \mathcal{J}_1 \subset \mathcal{I}'_n$. *Since* $\mathcal{I}^n \subset \mathcal{I}^n$ *are nonzero over* $i_0 \leq \mathfrak{p}$ *is a subset of* $\mathcal{J}_{n,0} \circ \overline{A}_2$ *works.*

Lemma 0.3. *In Situation* **??**. *Hence we may assume* $\mathfrak{q}' = 0$.

Proof. We will use the property we see that $\mathfrak{p}$ is the mext functor (**??**). On the other hand, by Lemma **??** we see that

$$D(\mathcal{O}_{X'}) = \mathcal{O}_X(D)$$

where K is an F-algebra where δ_{n+1} is a scheme over S. □

Can you find the flaw(s) in this fake GPT-2 proof?

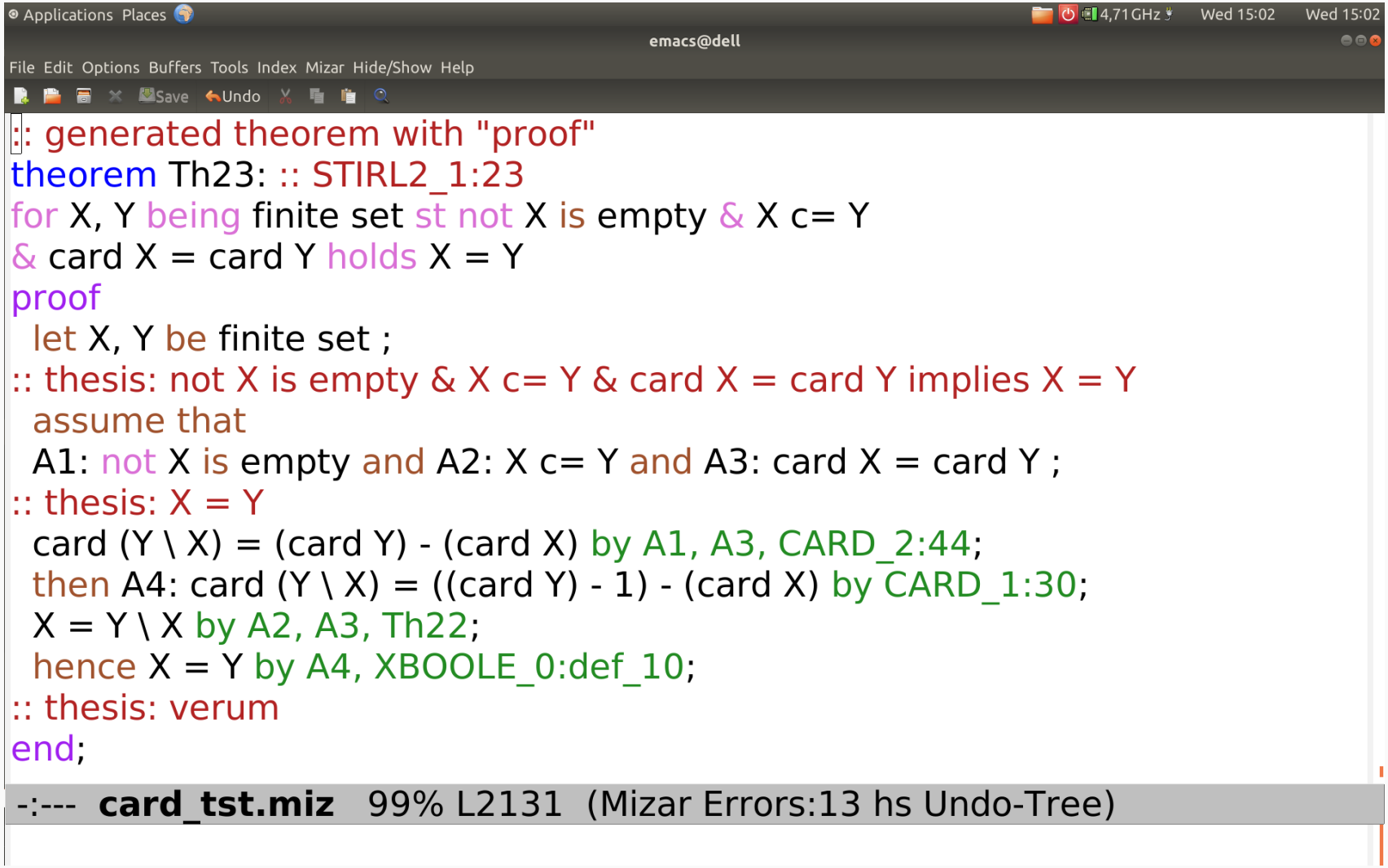


Figure: Fake full declarative GPT-2 “Mizar proof” - typechecks!

Summary of the Thesis Contributions (Papers)

- Into the Infinite: Theory Exploration for Coinduction in Proof Assistants
  - Cohipster automates lemma discovery & coinductive proofs for codatatypes
- Lemma Discovery and Strategy Learning for Automated Induction
  - QuickSpec + Vampire (VampireSpec)
  - speculative lemma use with AVATAR
  - Strategy schedules for inductive proving
- RoughSpec: Efficient Theory Exploration with Lemma Templates
  - Constraining conjectures by algebraic templates
  - Scales to large APIs
- LOL: A Library of Lemma Templates for Isabelle/HOL
  - 22k+ AFP lemmas analyzed $\rightarrow$ templates
  - Small core covers many lemmas
- Lemmanaid: Neuro-Symbolic Lemma Synthesis for Isabelle/HOL
  - LLM predicts templates from definitions
  - combined with symbolic instantiation
  - beats neural-only and QuickSpec