Experiences with Natural Language Proof Checking

by Peter Koepke

University of Bonn, Germany

CICM, NatFoM, Cambridge, September 6th 2023

$\mathbb{N}$aproche (Natural Proof Checking)

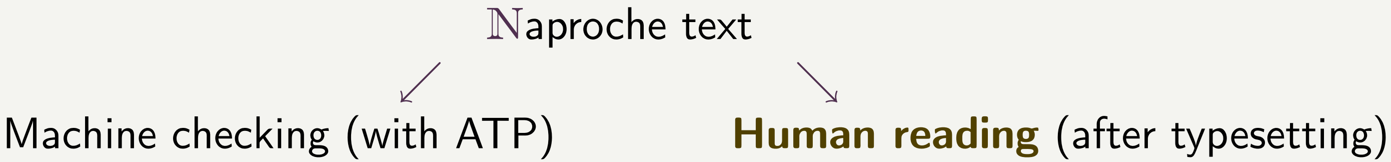
Since ∼2017, building on SAD (A. Paskevich) and previous Naproche (M. Cramer)

Controlled Natural Language for Mathematics as ITP input (LaTeX format)

Natural proof details and granularities

Natural ontology (First-order logic, set/function/number theory)

Convenient prover environment (Isabelle/jEdit)

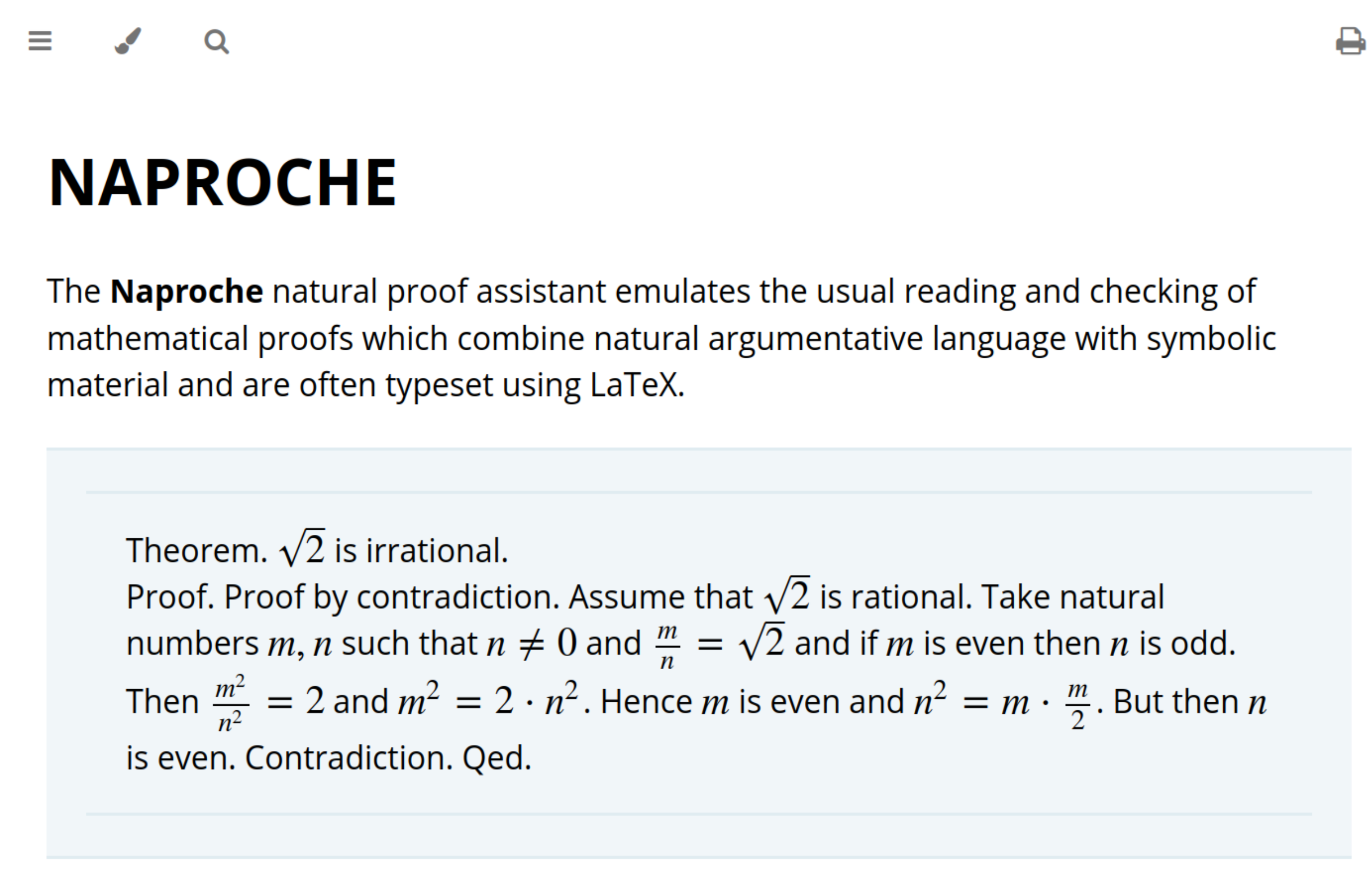


$\mathbb{N}$aproche text

Machine checking (with ATP)

Human reading (after typesetting)

`https://naproche.github.io/`



NAPROCHE

The **Naproche** natural proof assistant emulates the usual reading and checking of mathematical proofs which combine natural argumentative language with symbolic material and are often typeset using LaTeX.

> Theorem. $\sqrt{2}$ is irrational.
> Proof. Proof by contradiction. Assume that $\sqrt{2}$ is rational. Take natural numbers m, n such that $n \neq 0$ and $\frac{m}{n} = \sqrt{2}$ and if m is even then n is odd. Then $\frac{m^2}{n^2} = 2$ and $m^2 = 2 \cdot n^2$. Hence m is even and $n^2 = m \cdot \frac{m}{2}$. But then n is even. Contradiction. Qed.

The $\mathbb{N}$aproche process

Controlled Natural mathematical language (ForTheL, Formula Theory Language, LaTeX)

$\Downarrow$ *Parser + Simplifications*

(Enriched) First-order statements, in an internal representation

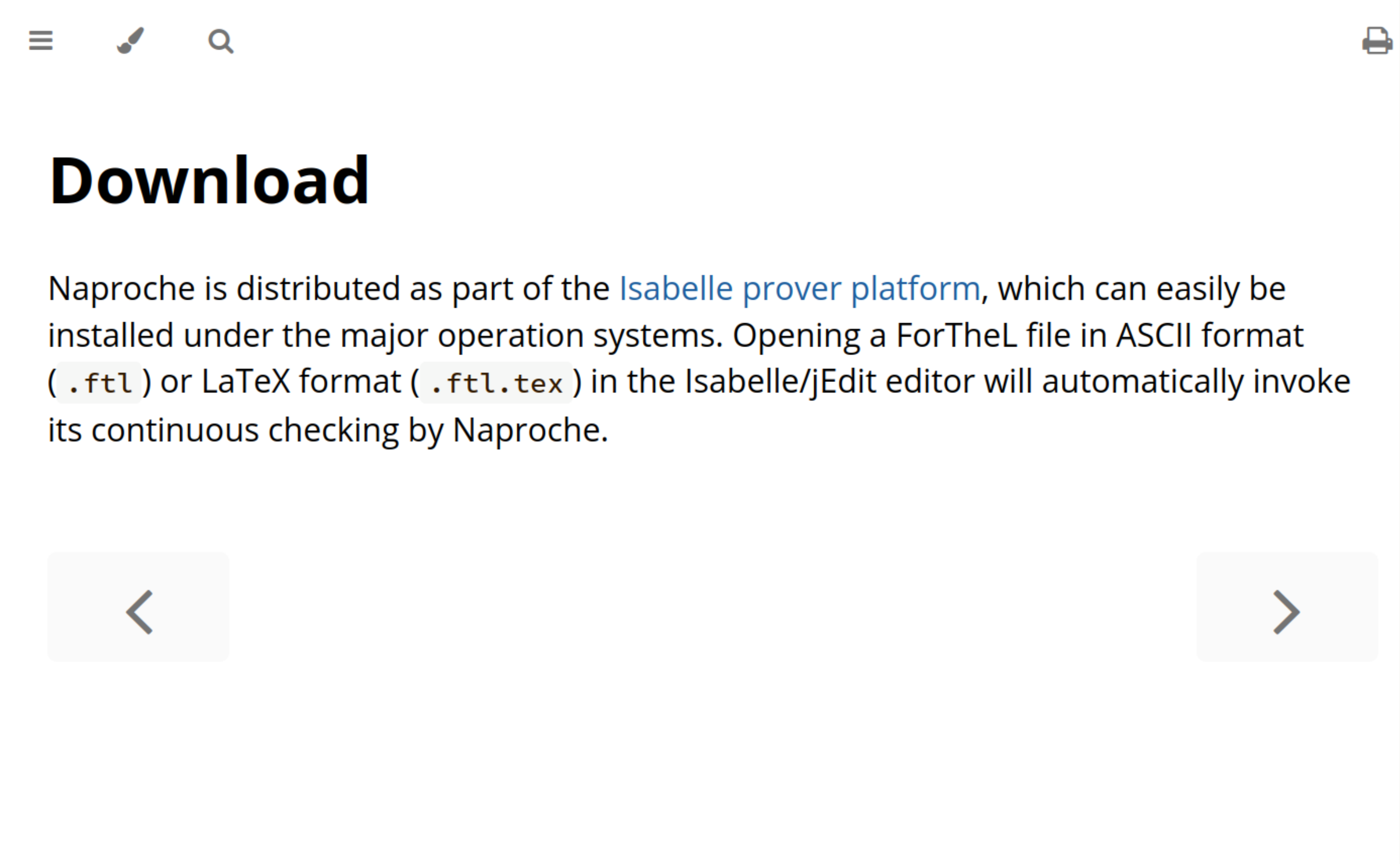
$\Downarrow$ *Reasoner*

Proof tasks (TPTP)

$\Downarrow$ *E prover* (*or Vampire*, ...)

```
verification successful / failed
```

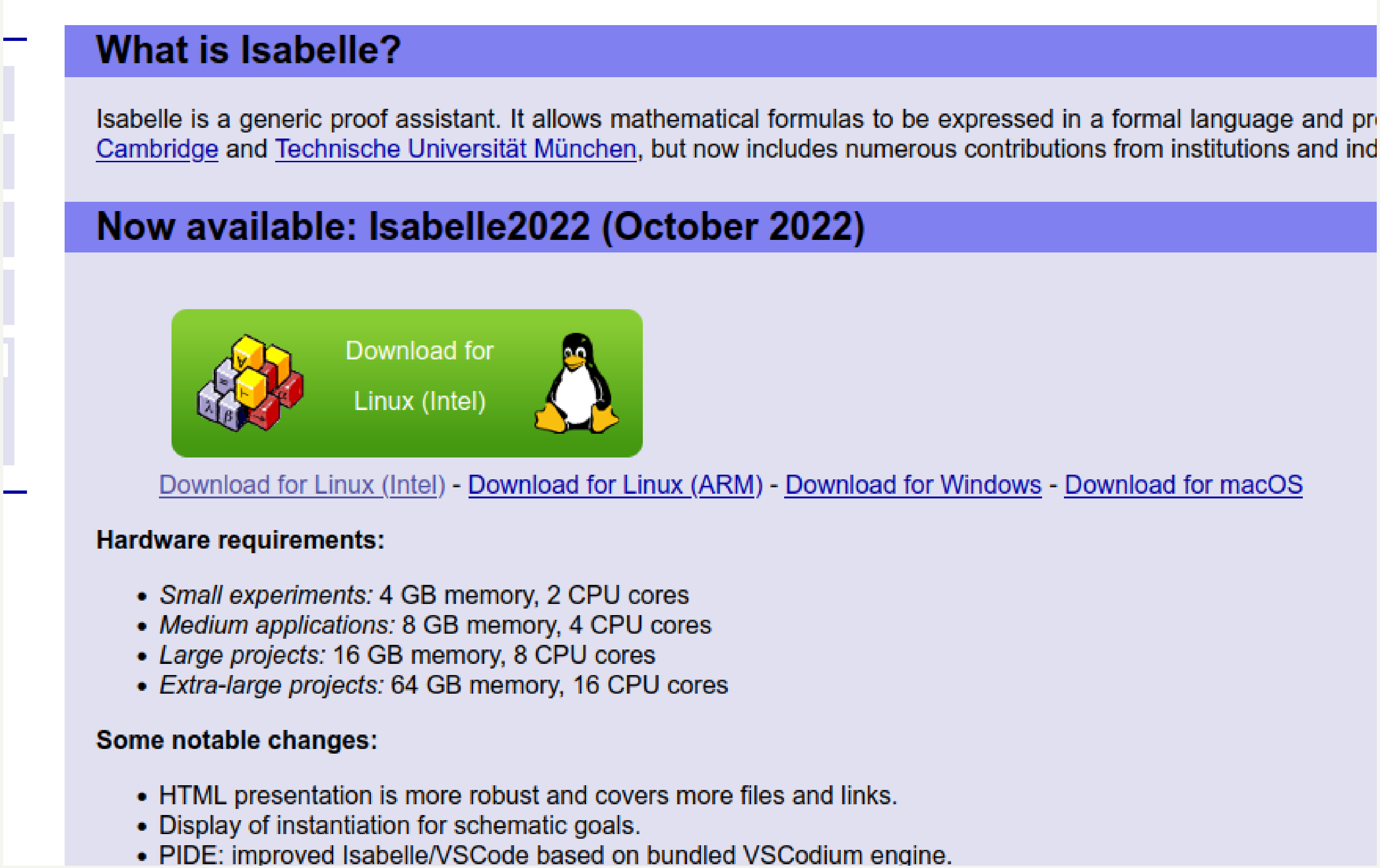
https://naproche.github.io/



Download

Naproche is distributed as part of the Isabelle prover platform, which can easily be installed under the major operation systems. Opening a ForTheL file in ASCII format (`.ftl`) or LaTeX format (`.ftl.tex`) in the Isabelle/jEdit editor will automatically invoke its continuous checking by Naproche.

Isabelle



What is Isabelle?

Isabelle is a generic proof assistant. It allows mathematical formulas to be expressed in a formal language and pr
Cambridge and Technische Universität München, but now includes numerous contributions from institutions and ind

Now available: Isabelle2022 (October 2022)

Download for Linux (Intel) - Download for Linux (ARM) - Download for Windows - Download for macOS

Hardware requirements:

- *Small experiments:* 4 GB memory, 2 CPU cores
- *Medium applications:* 8 GB memory, 4 CPU cores
- *Large projects:* 16 GB memory, 8 CPU cores
- *Extra-large projects:* 64 GB memory, 16 CPU cores

Some notable changes:

- HTML presentation is more robust and covers more files and links.
- Display of instantiation for schematic goals.
- PIDE: improved Isabelle/VSCode based on bundled VSCodium engine.

ℕaproche **Demo** in Isabelle/jEdit

– ℕaproche is (*in principle*) a complete proof assistant

– ℕaproche so far is prototypical, incomplete, unstable, …

- limited library of formalizations

– ℕaproche proofs can be readable

– Readability could be important for the acceptability of formal mathematics

Formalizing in ℕaproche requires dealing with:

– the mathematical problem at hand

– the underlying logic of ℕaproche (FOL)

– first-order proving with external ATPs

– the input language ForTheL

Formalizing in $\mathbb{N}$aproche requires dealing with:

– the mathematical problem at hand

– the underlying logic of $\mathbb{N}$aproche (FOL)

– first-order proving with external ATPs

– the input language ForTheL

plus naturalness requirements:

– natural mathematical language

– natural foundational assumptions (FOL and sets)

– natural proof granularities (heavy prover load)

- natural proof and text structures

– natural mathematical typesetting (LaTeX)

Naturalness requirements:

– natural mathematical language

– natural foundational assumptions (FOL and sets)

– natural proof granularities (heavy prover load)

- natural proof and text structures

– natural mathematical typesetting (LATEX)

The Burden of Naturalness:

– natural mathematical language: working in a rich controlled natural language (CNL, ForTheL) which approximates but is not equal to free natural language, possibilities of ambiguity and unexpected interpretations

– natural foundational assumptions (FOL and sets): interplay between natural language soft types and “first-order types”, type guards

– natural proof granularities (heavy prover load): declarative proofs, proof steps may take many seconds or may time out on E prover, resolution proving often “unnatural”
($\forall x \exists y \neg y < f(x) \Longleftrightarrow \forall x \exists y\, f(x) \leqslant y$),

- natural proof and text structures

– natural mathematical typesetting (LaTeX): document structuring, layout, LaTeX macros for natural symbolism, natural naming of entities, “aesthetic” requirements

– difficult user experience, $\mathbb{N}$aproche has so far only been used in the Bonn group

The Burden of Naturalness:

Possible Computer Assistance:

- natural mathematical language: improving ForTheL and ForThel parsing, language support in the editor, like suggestions of possible phrases

- natural foundational assumptions (FOL and sets): libraries of foundational files

- natural proof granularities (heavy prover load): improving the ℕaproche reasoner and the use of external ATPs, caching of prover outputs, (hoping for) stronger ATPs, term rewriting, SMT for certain theories

- natural mathematical typesetting (LaTeX): WYSIWYG mathematical editing, like TeXmacs

- . . .

Options for further developments

– Invest massively into the code of the $\mathbb{N}$aproche system

– Program a new natural system that circumvents weaknesses and complications of $\mathbb{N}$aproche; indeed, Adrian De Lon is working towards a novel set-theory based natural proof assistant

- Combine the natural language approach of $\mathbb{N}$aproche with established systems; $\mathbb{N}$aproche-type interfaces or input languages to Isabelle/HOL, Mizar, ...

- Focus on limited areas of mathematics for didactical or demonstration purposes, like introductory number systems or undergraduate calculus

– ..., ???

General take-home:

Formal mathematics can be done with a natural language input

Thank you!