Formalism and Computations

Peter Koepke, University of Bonn, Germany

Mathematical Institute

15th Congress of Logic, Methodology and Philosophy of Science

Session of History and Philosophy of Computing

Helsinki, 7 August 2015

Semi-formal proofs: natural language and symbolic formulas

Proofs from the Book:

Euclid's Proof. For any finite set $\{p_1, ..., p_r\}$ of primes, consider the number $n = p_1\, p_2 \cdots p_r + 1$. This n has a prime divisor p. But p is not one of the p_i: otherwise p would be a divisor of n and of the product $p_1\, p_2 \cdots p_r$, and thus also of the difference $n - p_1\, p_2 \cdots p_r = 1$, which is impossible. So a finite set $\{p_1, ..., p_r\}$ cannot be the collection of *all* prime numbers. <box>

Formalism: Aristotelian syllogisms

All men are mortal

All Greeks are men

All Greeks are mortal

All B are A

All C are B

All C are A

Formalism: David Hilbert's Axiomatic Method

I 2. *Irgend zwei voneinander verschiedene Punkte einer Geraden bestimmen diese Gerade.*

I 3. *Auf einer Geraden gibt es stets wenigstens zwei Punkte, in einer Ebene gibt es stets wenigstens drei nicht auf einer Geraden gelegene Punkte.*

I 4. *Drei nicht auf ein und derselben Geraden liegende Punkte A, B, C bestimmen stets eine Ebene α.*

Wir gebrauchen auch die Wendungen: *A, B, C* „liegen in" *α*; *A, B, C* „sind Punkte von" *α* u. s. w.

I 5. *Irgend drei Punkte einer Ebene, die nicht auf ein und derselben Geraden liegen, bestimmen die Ebene α.*

I 6. *Wenn zwei Punkte A, B einer Geraden a in einer Ebene α liegen, so liegt jeder Punkt von a in der Ebene α.*

In diesem Falle sagen wir: die Gerade *a* liegt in der Ebene *α* u. s. w.

Felix Hausdorff on Formalism in Mathematics

([...] *formalism* [...], which dominates modern mathematics in an ever growing determinateness and consciousness; [...] formalism which originally only intends to clearly bring out the premises of each deduction and to avoid nebulous appeals to the seemingly self-evident, and which in its further development has lead to the complete emancipation from “intuition” and other specific, extra-logical sources of knowledge, and which in its present form also defines a duty and a right of pure mathematics: the duty, never to recurr onto the accidental actual meaning of the objects and their relation, but to define and deduce from precisely given undefined notions and undeduced judgments (axioms) everything else, and the right to give such a logically built system of things and relations every logically admissable interpretation.)

Symbolic Formalism: Whitehead, Russell: Principia Mathematica

***54·43.** $\vdash :. \alpha, \beta \in 1 . \supset : \alpha \cap \beta = \Lambda . \equiv . \alpha \cup \beta \in 2$

Dem.

$\vdash . *54{\cdot}26 . \supset \vdash :. \alpha = \iota`x . \beta = \iota`y . \supset : \alpha \cup \beta \in 2 . \equiv . x \neq y .$

[*51·231] $\equiv . \iota`x \cap \iota`y = \Lambda .$

[*13·12] $\equiv . \alpha \cap \beta = \Lambda$ (1)

$\vdash . (1) . *11{\cdot}11{\cdot}35 . \supset$

$\vdash :. (\exists x, y) . \alpha = \iota`x . \beta = \iota`y . \supset : \alpha \cup \beta \in 2 . \equiv . \alpha \cap \beta = \Lambda$ (2)

$\vdash . (2) . *11{\cdot}54 . *52{\cdot}1 . \supset \vdash .$ Prop

From this proposition it will follow, when arithmetical addition has been defined, that $1 + 1 = 2$.

The Gödel completeness theorem

Über die Vollständigkeit des Logikkalküls
(1929)

1. Einleitung

Der Hauptgegenstand der folgenden Untersuchungen ist der Beweis der Vollständigkeit des in Russell, *Principia mathematica*, P. I, Nr. 1 und Nr. 10, und ähnlich in Hilbert–Ackermann, *Grundzüge der theoretischen Logik* (zitiert als H. A.), III, § 5, angegebenen Axiomensystems des sogenannten engeren Funktionenkalküls. Dabei soll "Vollständigkeit" bedeuten, daß jede im engeren Funktionenkalkül ausdrückbare allgemein giltige Formel (allgemein giltige Zählaussage nach Löwenheim) sich durch eine endliche Reihe formaler Schlüsse aus den Axiomen deduzieren lässt. Diese Behauptung lässt sich leicht als äquivalent erkennen mit der folgenden: Jedes widerspruchslose nur aus Zählaussagen bestehende Axiomensystem[1] hat eine Realisierung. (Widerspruchslos heißt dabei, daß durch endlich viele formale Schlüsse kein Widerspruch hergeleitet werden kann.)

(Doctoral Dissertation, Vienna 1929)

The Gödel completeness theorem

Every logically true mathematical statement has a formal derivation.

Every true mathematical statement has a formal derivation within some (foundational) axiom system.

Every mathematical proof can be replaced by a formal derivation.

Mathematics can be in principle be carried out completely formal (*Formal mathematics*).

1.	Φ_{Gr}	$\neg \circ v_0\, e \equiv v_0$		$\neg \exists v_0 \neg \circ v_0\, e \equiv v_0$	VR
2.	Φ_{Gr}	$\neg \circ v_0\, e \equiv v_0$		$\neg \circ v_0\, e \equiv v_0$	VR
3.	Φ_{Gr}	$\neg \circ v_0\, e \equiv v_0$		$\exists v_0 \neg \circ v_0\, e \equiv v_0$	$\exists$S auf 2
4.	Φ_{Gr}			$\circ v_0\, e \equiv v_0$	WR auf 1,3
5.				$(v_2 \equiv \circ v_0\, e)\frac{\circ v_0\, e}{v_2}$	($\equiv$)
6.		$\circ v_0\, e \equiv v_0$		$(v_2 \equiv \circ v_0\, e)\frac{v_0}{v_2}$	Sub auf 5
7.	Φ_{Gr}	$\circ v_0\, e \equiv v_0$		$v_0 \equiv \circ v_0\, e$	AR auf 6
8.	Φ_{Gr}			$v_0 \equiv \circ v_0\, e$	KS auf 4,7
9.	Φ_{Gr}	$v_0 \equiv e$		$v_0 \equiv e$	VR
10.	Φ_{Gr}	$v_0 \equiv e$		$(\neg \circ v_0\, e \equiv e \vee v_0 \equiv e)$	$\vee$S auf 9
11.	Φ_{Gr}	$\neg v_0 \equiv e$		$(\neg v_2 \equiv e)\frac{v_0}{v_2}$	VR
12.	Φ_{Gr}	$\neg v_0 \equiv e$	$v_0 \equiv \circ v_0\, e$	$(\neg v_2 \equiv e)\frac{\circ v_0\, e}{v_2}$	Sub auf 11
13.	Φ_{Gr}	$\neg v_0 \equiv e$	$v_0 \equiv \circ v_0\, e$	$\neg \circ v_0\, e \equiv e$	12
14.	Φ_{Gr}	$\neg v_0 \equiv e$		$v_0 \equiv \circ v_0\, e$	AR auf 8
15.	Φ_{Gr}	$\neg v_0 \equiv e$		$\neg \circ v_0\, e \equiv e$	KS auf 14
16.	Φ_{Gr}	$\neg v_0 \equiv e$		$(\neg \circ v_0\, e \equiv e \vee v_0 \equiv e)$	$\vee$S auf 15
17.	Φ_{Gr}			$(\neg \circ v_0\, e \equiv e \vee v_0 \equiv e)$	FU auf 10,16
18.	Φ_{Gr}	$\neg(\neg \circ v_0\, e \equiv e \vee v_0 \equiv e)$	$\neg\neg \exists v_0 \neg(\neg \circ v_0\, e \equiv e \vee v_0 \equiv e)$	$(\neg \circ v_0\, e \equiv e \vee v_0 \equiv e)$	AR auf 17
19.	Φ_{Gr}	$\neg(\neg \circ v_0\, e \equiv e \vee v_0 \equiv e)$	$\neg\neg \exists v_0 \neg(\neg \circ v_0\, e \equiv e \vee v_0 \equiv e)$	$\neg(\neg \circ v_0\, e \equiv e \vee v_0 \equiv e)$	VR
20.	Φ_{Gr}	$\neg(\neg \circ v_0\, e \equiv e \vee v_0 \equiv e)$		$\neg \exists v_0 \neg(\neg \circ v_0\, e \equiv e \vee v_0 \equiv e)$	WR auf 18,19
21.	Φ_{Gr}	$\exists v_0 \neg(\neg \circ v_0\, e \equiv e \vee v_0 \equiv e)$		$\neg \exists v_0 \neg(\neg \circ v_0\, e \equiv e \vee v_0 \equiv e)$	$\exists$A auf 20
22.	Φ_{Gr}	$\neg \exists v_0 \neg(\neg \circ v_0\, e \equiv e \vee v_0 \equiv e)$		$\neg \exists v_0 \neg(\neg \circ v_0\, e \equiv e \vee v_0 \equiv e)$	VR
23.	Φ_{Gr}			$\neg \exists v_0 \neg(\neg \circ v_0\, e \equiv e \vee v_0 \equiv e)$	FU auf 21,22

Formal proofs - derivations

N. Bourbaki:

If formalized mathematics were as simple as the game of chess, then once our chosen formalized language had been described there would remain only the task of writing out our proofs in this language, [...] But the matter is far from being as simple as that, and no great experience is necessary to perceive that such a project is absolutely unrealizable: the tiniest proof at the beginnings of the Theory of Sets would already require several hundreds of signs for its complete formalization. [...] formalized mathematics cannot in practice be written down in full, [...] We shall therefore very quickly abandon formalized mathematics, [...]

Computer-supported formal proofs

J. McCarthy:

Checking mathematical proofs is potentially one of the most interesting and useful applications of automatic computers. ... Proofs to be checked by computer may be briefer and easier to write than the informal proofs acceptable to mathematicians. This is because the computer can be asked to do much more work to check each step than a human is willing to do, and this permits longer and fewer steps.

McCarthy, J. "Computer Programs for Checking Mathematical Proofs," Proceedings of the Symposium in Pure Math, Recursive Function Theory,Volume V, pages 219-228, AMS, Providence, RI, 1962.

The development of formal mathematics systems

- Automath, de Bruijn, ~1967
- Mizar, Trybulec, ~1973
- **Isabelle/Isar**, Paulson, Nipkow, Wenzel, ~2002
- **Coq**
- **HOL Light**, Harrison
- many other systems

MIZAR example: Proof of Pythagoras

```
theorem for p1,p2,p3 st p1<>p2 & p3<>p2 &
    (angle(p1,p2,p3)=PI/2 or angle(p1,p2,p3)=3/2*PI) holds
    (|.p1-p2.|^2+|.p3-p2.|^2=|.p1-p3.|^2
    proof let p1,p2,p3; assume A1: p1<>p2 & p3<>p2 &
    (angle(p1,p2,p3)=PI/2 or angle(p1,p2,p3)=3/2*PI);
  then A2: euc2cpx(p1)<> euc2cpx(p2) by Th6;
  A3: euc2cpx(p3)<> euc2cpx(p2) by A1,Th6;
  A4: euc2cpx(p1)-euc2cpx(p2)=euc2cpx(p1-p2) by Th19;
  A5: euc2cpx(p3)-euc2cpx(p2)=euc2cpx(p3-p2) by Th19;
  A6: euc2cpx(p1)-euc2cpx(p3)=euc2cpx(p1-p3) by Th19;
   A7:  angle(p1,p2,p3)=angle(euc2cpx(p1),euc2cpx(p2),euc2cpx(p3))
by Def4;
  A8: |.euc2cpx(p1-p2).|=|.p1-p2.| by Th31;
  A9: |.euc2cpx(p3-p2).|=|.p3-p2.| by Th31;
   |.euc2cpx(p1-p3).|=|.p1-p3.| by Th31;
  hence thesis by A1,A2,A3,A4,A5,A6,A7,A8,A9,COMPLEX2:91;
end;
```

Substantial mathematics in Mizar

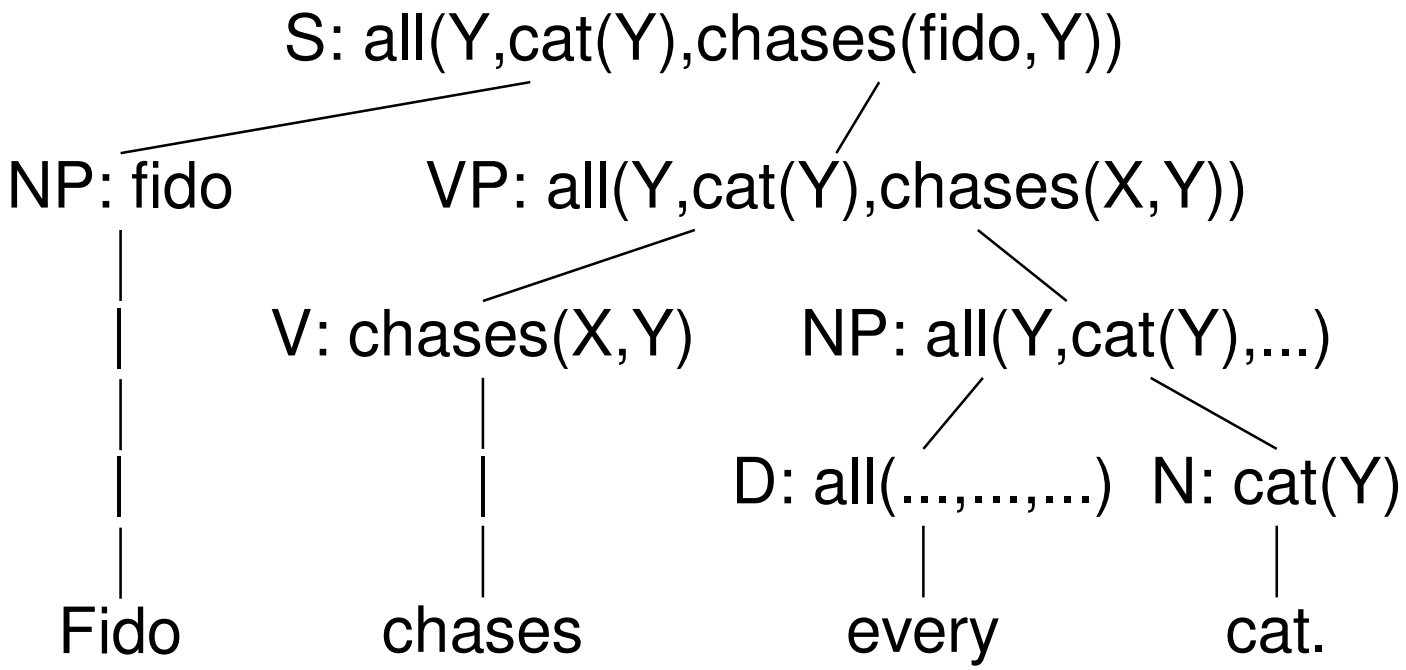
Banach Fixed Point Theorem for compact spaces, the Brouwer Fixed Point Theorem, the Birkhoff Variety Theorem for manysorted algebras, Fermat's Little Theorem, the Fundamental Theorem of Algebra, the Fundamental Theorem of Arithmetic, the Gödel Completeness Theorem, the Hahn-Banach Theorem for complex and real spaces, the Jordan Curve Theorem for special polygons, ...

Mathematical statements

“1 divides every integer.” ⟷“Fido chases every cat.”

Linguistic analysis

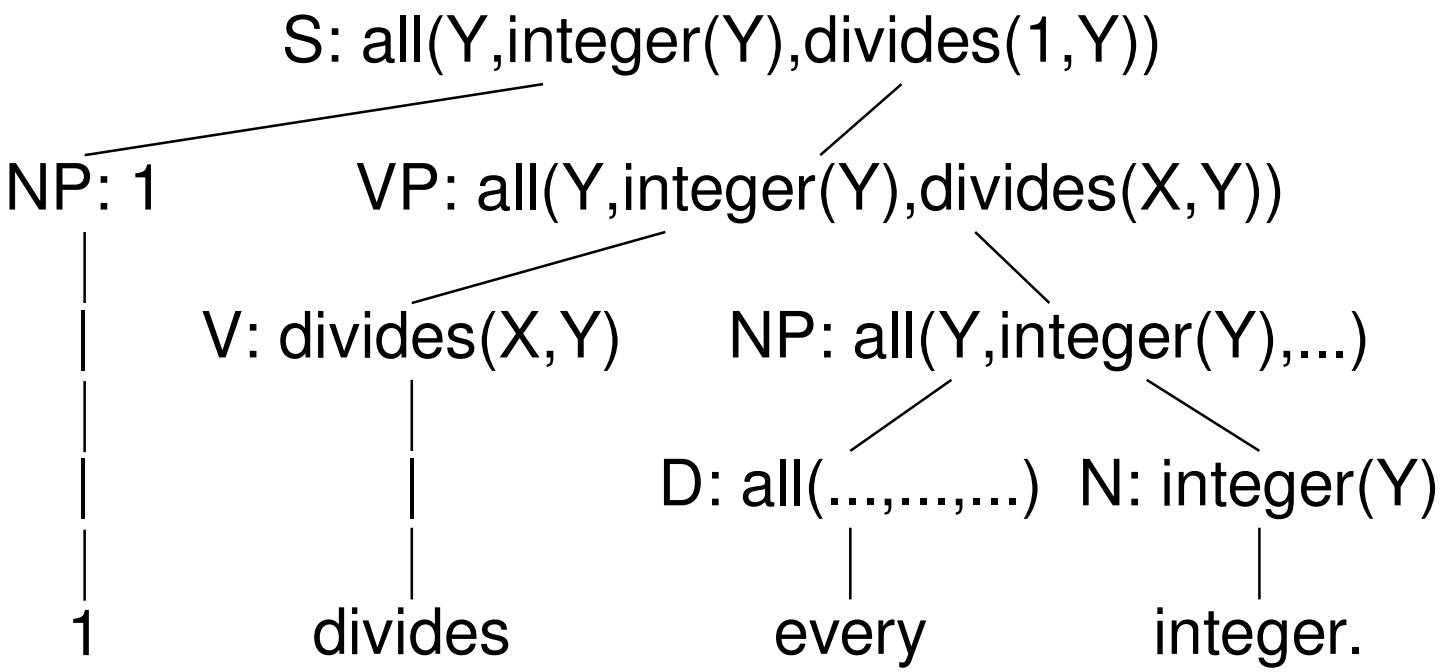
“Fido chases every cat.”



$\forall Y(\mathrm{cat}(Y) \rightarrow \mathrm{chases}(\mathrm{fido}, Y))$.

Linguistic analysis

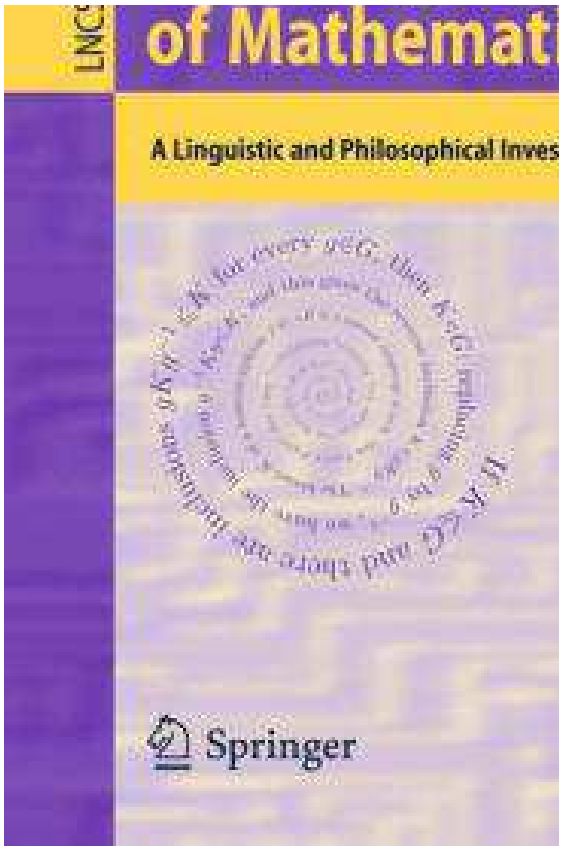
“1 divides every integer.”



$\forall Y(\text{integer}(Y) \rightarrow 1|Y).$

The Language of Mathematics

- Mohan Ganesalingam: *The Language of Mathematics*,



The Naproche project: Natural language proof checking

- models natural language proofs using computer-supported methods of formal linguistics and formal logic

- “reverse engineering” of natural proofs: to devise a strictly formal system for mathematics, implemented by computer, whose input language is an extensive part of the common mathematical language, and whose proof style is close to proof styles found in the mathematical literature.

- joint work with Bernhard Schröder, linguistics; Bonn, Essen, Cologne; www.naproche.net

Layers of the Naproche system:

↓ Standard or web editor

LaTeX-style input text

↕ Natural language processing (NLP)

Proof representation structure (PRS)

↕ First-order translation

First-order logic format (TPTP)

↕ Proof checker or automatic theorem prover (ATP)

“Accepted”/“Not accepted”, with error messages

Andrei Paskevich' System of Automatic Deduction (SAD)

- started by Victor Glushkov, continued with Alexander Lyaletski and Konstantin Verchinine

- simple phrase structure grammar

- has reasoner and ATP

- http://nevidal.org/sad.en.html

Combining Naproche and SAD

Theorem 1. *The set of prime numbers is infinite.*

Proof. Let A be a finite set of prime numbers. Take a function p and a number r such that p lists A in r steps. ran $p \subseteq \mathbb{N}^+$. $\prod_{i=1}^{r} p_i \in \mathbb{N}^+$. Take $n = \prod_{i=1}^{r} p_i + 1$. n is nontrivial. Take a prime divisor q of n .

Let us show that q is not an element of A. Assume the contrary. Take j such that ($1 \leqslant j \leqslant r$ and $q = p_j$). p_j divides $\prod_{i=1}^{r} p_i$ (by MultProd). Then q divides 1 (by DivMin). Contradiction. qed.

Hence A is not the set of prime numbers. □

A natural proof that is fully formal

Theorem 2. *The set of prime numbers is infinite.*

Proof. Let A be a finite set of prime numbers. Take a function p and a number r such that p lists A in r steps. ran $p \subseteq \mathbb{N}^+$. $\prod_{i=1}^{r} p_i \in \mathbb{N}^+$. Take $n = \prod_{i=1}^{r} p_i + 1$. n is nontrivial. Take a prime divisor q of n .

Let us show that q is not an element of A. Assume the contrary. Take j such that $(1 \leqslant j \leqslant r$ and $q = p_j)$. p_j divides $\prod_{i=1}^{r} p_i$ (by MultProd). Then q divides 1 (by DivMin). Contradiction. qed.

Hence A is not the set of prime numbers. □

Euclid's Proof. For any finite set $\{p_1, ..., p_r\}$ of primes, consider the number $n = p_1\, p_2 \cdots p_r + 1$. This n has a prime divisor p.

But p is not one of the p_i : otherwise p would be a divisor of n and of the product $p_1\, p_2 \cdots p_r$, and thus also of the difference $n - p_1\, p_2 \cdots p_r = 1$, which is impossible.

So a finite set $\{p_1, ..., p_r\}$ cannot be the collection of *all* prime numbers. <box>

What is a mathematical proof?

- argumentative text about the/some mathematical “reality”?
- argumentative text within some system of initial assumptions (axioms)?
- abbreviation for some (long) formal derivation?
- recipe for building a formal derivation if required?
- a formal derivation in some rich formal system (compare with: R. Montague: English as a formal language)?
- there are natural(ly looking) proofs that are fully formal with respect to some system like Naproche

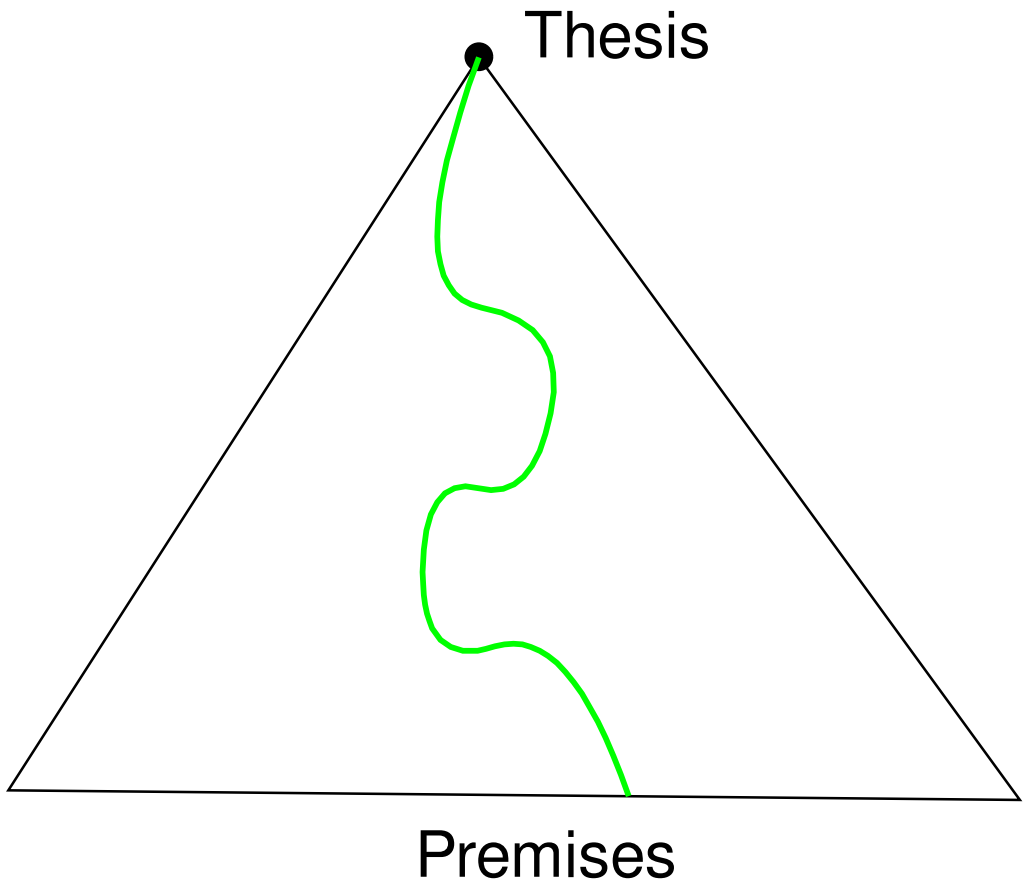
Formalism and computations

- ”small scale” formalism: syllogisms
- systematic formalism: axiomatics, formal mathematics
- theoretical power of formalism: completeness theorem
- ”large scale” formalism: computer supported formal mathematics
- ”natural” formalism, using computational linguistics
- will “natural” formalism be accepted in mathematical practice?

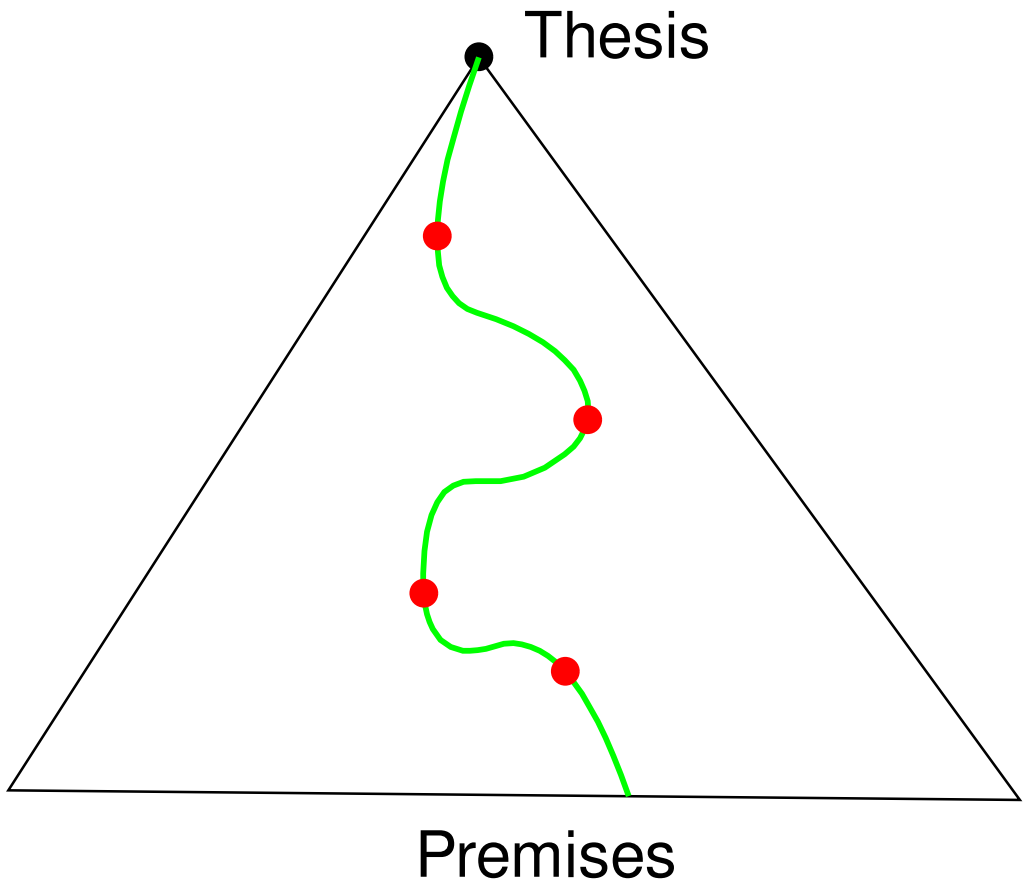
A similar situation: natural number arithmetic

- ”small scale” reckoning
- systematic reckoning: long multiplication
- correctness of multiplication algorithm
- ”large scale” reckoning with computers

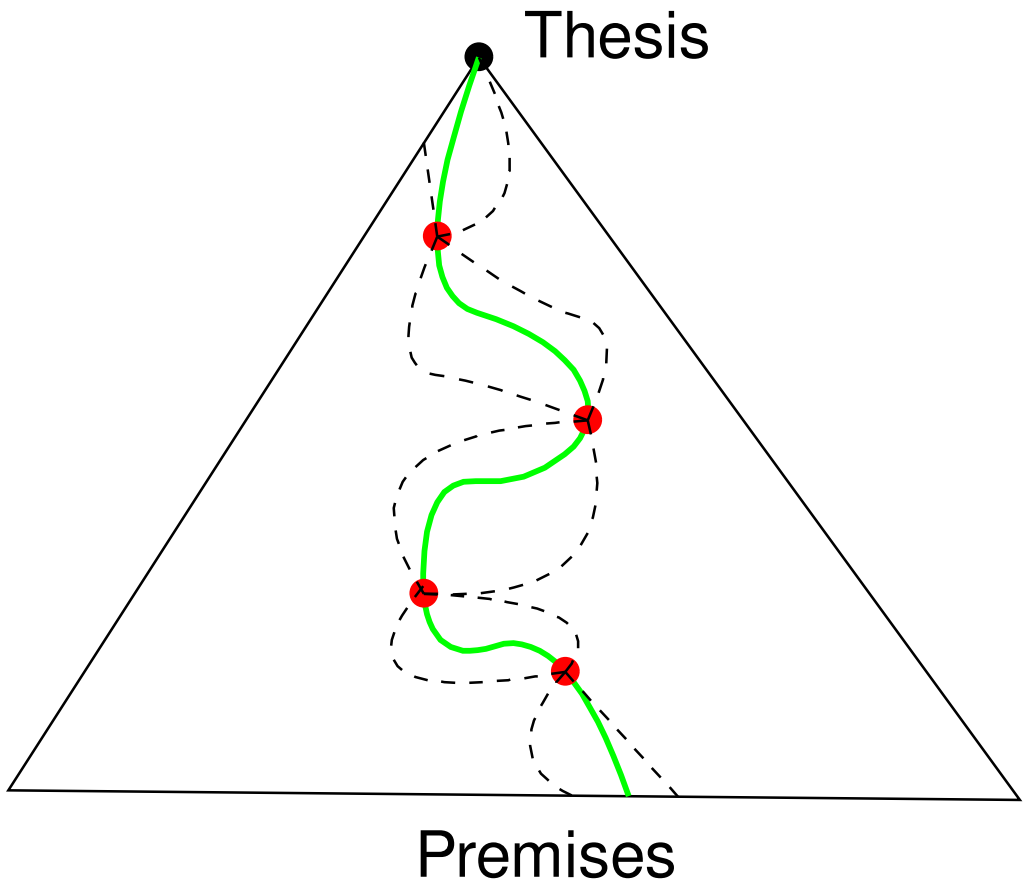
Complexity of formal proof search



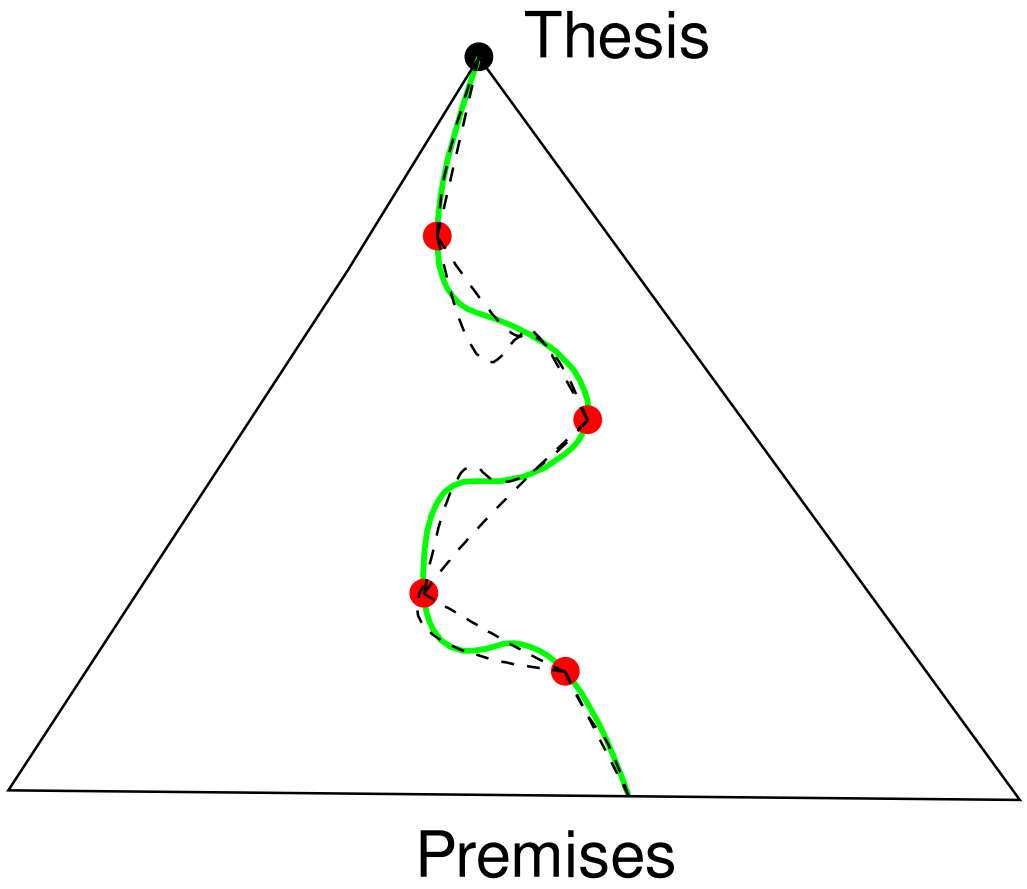
Complexity of formal proof checking a mathematical text



Complexity of formal proof checking a mathematical text



Complexity reduction by automated “reasoning”



Complexity issues in formal mathematics

- low complexity for some trivial examples
- combinatorial blow-up, (hyper-)exponential growth of search spaces
- large complexity reduction by proof checking (= local proof search) versus (global)) proof search
- currently, human input is required for feasible complexities
- more efficient search by mathematical, software and hardware advances: adequate axiomatics, automatic theorem provers for “obvious” arguments, parallel computations

Philosophical aspects

- formalism as a philosophy of mathematics
- criticism of formalism because of arbitrariness, impracticality, and unnaturalness
- natural, computer-supported formalism may bridge the gap between formalism and naturalism ... in the philosophy of mathematics
- does this strengthen the formalist position in the philosophy of mathematics?
- What would be the implications of a widespread use of formal mathematics for the methodology, practice, and philosophy of mathematics?

Thank You!