

# Fine-grained mathematical justifications

Jesse Alama

CENTRIA, New University of Lisbon

December 7, 2010 / Brouwer Seminar / Radboud University Nijmegen

# Contents

- Introduction to a meta-mathematical program
- Dependencies in mathematics and logic: various senses.
- Current work on dependencies



# My program

- Long-term: use tools from interactive and automated theorem proving in the service of foundations of mathematics.
- Medium-term: break formal mathematical texts into their smallest meaningful parts and develop tools for understanding the relationships between these parts, emphasizing the *dependency* of one part upon another.
- Short-term: carrying out this task for texts in the mizar language.



# About the long-term program

There are a variety of views about and goals for computers in mathematics, such as:

- making substantial part of mathematics 'computer-understandable' and formally checked;
- providing deductive and inductive tools for mathematics;
- supporting large collaborative mathematical efforts.

But what to do with the 'data' that we get thanks to these tasks? I suggest that we can take on some classical problems in the foundations of mathematics:

- the role of infinity in mathematics;
- identity of proofs;
- necessary and sufficient conditions for theorems.



# About the medium-term program

- Aim: to understand *dependence* in mathematics by focusing on large corpora of formalized mathematical knowledge.
- These corpora are the result of enormous efforts by formalizers around the world.
- In various systems, formalizers have worked hard to ensure that their formalized mathematical proofs are spelled out in great detail.



# About the short-term program

- Focus on the mizar Mathematical Library (MML) a rather large corpus of mathematical knowledge formalized in classical first-order set theory.
- The MML is divided into about 1100 ‘articles’, averaging about 2200 lines of text.
- The mizar language allows various kinds of ‘items’, such as theorems, proofs, lemmas, and definitions.
- **Task:** Break the mizar Mathematical library into its smallest meaningful pieces, so that we can have a fine-grained analysis of the dependence of one item upon another.

Although the focus at the moment is on mizar, one can imagine that this kind of work will bear fruit for other systems, such as coq.



# Kinds of dependencies

Many kinds of 'dependency' are available:

- conceptual/semantic dependencies
  - A proof of a theorem can involve notions that do not 'figure' or 'appear' in the theorem itself.
- logical dependencies:
  - dependency on axioms
  - dependency on language
  - dependency on rules of inference
  - dependency on analysis of propositions (propositional vs. first-order logic vs. second-order logic)



## Kinds of dependencies (II)

- system dependence:
  - mizar has its own notion of what counts as a ‘step’ in a proof, or what makes a formula well-formed; other systems do things differently.



# Dependency: examples: I

*On what does the theorem " $p \rightarrow p$ " depend?*

- Answer 1: classical logic: it's equivalent to  $p \vee \neg p$ , a characteristic classical axiom.
- Answer 2: classical logic: it's a tautology.
- Answer 3: intuitionistic logic: it is acceptable under the Brouwer-Heyting-Kolmogorov interpretation
- Answer 4: intuitionistic logic: in a Hilbert calculus, it follows from the B, C, K, and I axioms
- Answer 5: intuitionistic logic, because it follows from (since it is identical to) the K-axiom.



# Dependency: examples: I (continued)

- Answer 6: relevance logic.
- Answer 7: a *sui generis* logic axiomatized by itself.
- Answer 8: ...



# Dependency: examples: II

*On what does the theorem “ $1 + 1 = 1 + 1$ ” depend?*

- Answer -1: a first-order signature with ‘+’, ‘1’, and ‘=’.
- Answer 1: classical first-order logic with identity.
- Answer 2: intuitionistic first-order logic with identity.
- Answer 3: classical first-order logic *without* identity, with an axiom saying that ‘=’ is a reflexive relation.
- Answer 4: intuitionistic first-order logic *without* identity, with an axiom saying that ‘=’ is a reflexive relation.



# Dependency: examples: III

*On what does the Bolzano-Weierstrass theorem depend?*

- (All of) classical analysis, considered as a classical second-order theory
- The axioms for real ordered fields, considered as a classical first-order theory
- (reverse mathematics) **ACA**<sub>0</sub>, with respect to **RCA**<sub>0</sub>.



# Dependency: examples: IV

*On what does the theorem “Every triangle has 180 degrees” depend?*

- Answer -1: geometry
- Answer 1: (a theory of) real numbers
- Answer 2: Euclid’s axioms
- Answer 3: the parallel postulate



# Implicit vs explicit dependencies

- Sometimes dependencies can be explicitly marked as such: ‘We have  $\varphi$  by theorems  $t_1$  and  $t_2$ ’.
  - In formal mathematical texts, these can often be recovered easily.
- Other dependencies are implicit:
  - Background assumptions/theory
  - Requirements for the statements in the theorem/proof to be well-formed at all.

This suggests that part of the dependency problem in formalized mathematics is analogous to *computational pragmatics*, but carried out in tightly controlled domain of mathematical language.



# Undetectable dependencies

Some dependencies are, presumably, (mechanically) undetectable:

- all mizar proofs depend on classical first-order logic and Tarski-Grothendieck set theory
- all proofs in coq depend on lambda calculus, in particular the calculus of inductive constructions
- formalized proofs depend on the existence of computers
- mathematics depends on the existence of conscious beings

These kinds of dependencies are not ‘contentful’ for our purposes, since (to borrow a Kantian expression) they are just *conditions for the possibility* of our investigations.



# Identity of proofs

- The concept of dependency leads to another motivation for the problem of identity of proofs:
- Given two formalized proofs  $p_1$  and  $p_2$  of a theorem  $\varphi$  formalized in, say, mizar and coq,  $p_1 \sim p_2$  if, putting aside all ‘system dependencies’,  $p_1$  and  $p_2$  have the same dependencies.



# Complexity of computing dependencies: I

A version of the dependency problem:

**Problem:** *Given a list  $\Gamma$  of propositional formulas and a formula  $\varphi$  that is assumed to be logically implied by  $\Gamma$ , find a minimal subset  $\Gamma'$  of  $\Gamma$  such that  $\Gamma' \vdash \varphi$ .*

**NP**-complete in propositional case: **SAT** reduces to this problem: given a formula  $\varphi$ , choose  $\Gamma := \{\varphi\}$ . (It's not clear whether this problem reduces to **SAT**.)



# Complexity of computing dependencies: II

**Problem:** *Given a first-order signature  $\pi$ , a first-order  $\pi$ -sentence  $\varphi$ , and a set  $\Gamma$  of  $\pi$ -sentences, find a minimal subset  $\Gamma'$  of  $\Gamma$  such that  $\Gamma' \models \varphi$ .*

The general first-order validity/satisfiability problem reduces to this one. So this problem is undecidable in general (that is, quantifying over all signatures  $\pi$ , formulas  $\varphi$  and sets  $\Gamma$ ).

These theoretical bounds are well-known in the interactive and automated theorem proving communities.



# Dealing with dependency relations

- Given a dependency relation, we would like to operate on it. Two operations of interest: *transitive closure* and *transitive reduction*.
- With the transitive closure of our dependency graph, we can ask determine quickly whether one item depends on another.
  - But the transitive closure, since it has so many more edges than the original dependency graph, will require much space.
- With a transitive reduction of our dependency graph, we can store a minimal representation of it, saving space.
  - But answering dependency questions by querying *this* graph will require more time.



# Experiment I: axiom of replacement in set theory

- Axiom of replacement in set theory: “the image of a function applied to a set is a set”.
- Sometimes needed, sometimes not. (It is a genuine axiom; it cannot be wholly eliminated:  $\mathbf{Z} \neq \mathbf{ZF}$ . But despite its ‘naturalness’ in some proofs, often alternative proofs are available that avoid replacement.)
  - Discussion of the axiom with regard to category theory on MathOverflow.



# Experiment I: axiom of replacement in set theory (continued)

- Replacement is an axiom in mizar:

```
scheme Fraenkel { A()-> set, P[set, set] }:  
  ex X st for x holds  
    x in X  
  iff  
    ex y st y in A() & P[y,x]  
  provided  
    for x,y,z st P[x,y] & P[x,z]  
    holds y = z;  
end;
```



# Experiment I: axiom of replacement in set theory (continued)

- Thus, in any particular proof, we can trace precisely which instance of this scheme is being used.
- We can thereby generate a theorem proving problem: *find a proof of this instance of the scheme, using the other axioms*, thereby eliminating our commitment to replacement (in this case).



# Experiment II: eliminating Tarski's universe axiom

- Universe axiom: for every set  $a$  there exists a set  $\mathcal{U}$  such that
  - $a \in \mathcal{U}$ ,
  - $\mathcal{U}$  is closed under taking power sets,
  - $\mathcal{U}$  is closed under union,
  - for every subset  $A$  of  $\mathcal{U}$  for which  $|A| < \mathcal{U}$ , we have that  $A \in \mathcal{U}$
- Personal challenge: my mizar proof of Euler's polyhedron formula (2007).  
Need to demonstrate that Tarski's universe axiom is not necessary



# Experiment III: isolating infinity

- Axiom of infinity in set theory: an infinite set exists.
  - A consequence of Tarski's universe axiom
- But many particular theorems and proof steps in theorems don't need infinity—*which ones?*



# Work in progress

- mizar itemization: breaking up articles into the smallest meaningful parts.
- On average, articles in the mizar Mathematical Library have about 75 'items'.
- This permits a rather fine-grained division of the MML, capturing both implicit and explicit dependencies.



# Concrete Applications

- Complementing Josef Urban's MPTP (mizar Problems for Theorem Provers)
  - Sometimes finds new proofs of mizar theorems using different (or fewer) hypotheses.
- More fine-grained dependency information can help with the MathWiki project going on in Nijmegen:
  - fast recompilation



# Conclusions

- Working with large-scale corpora of formalized mathematics, a wealth of opportunities are available for taking on some problems in foundations of mathematics.
- Many of these problems are likely to be intractable, but they are no more intractable than many other problems taken on in the automated and interactive theorem proving communities.
- One can hope to discover new proofs and new results (most likely modest ones) thanks to the kinds of dependency investigations suggested here.



# References

- A. Arana, “On formally measuring and eliminating extraneous notions in proofs”, *Philosophia Mathematica* **17** (2009), 189–207.
- The mizar Homepage, <http://mizar.org>.
- A. Quaife, *Automated Development of Fundamental Mathematical Theories*, Kluwer Academic Publishers, 1992.
- S. Simpson, *Subsystems of Second-Order Arithmetic*, 2nd edition, Cambridge University Press, 2010.
- J. Urban, “MPTP—Motivation, implementation, first experiments”, *Journal of Automated Reasoning*, **33**(3–4) (2004), pp. 319–339.



## References (continued)

- C. Zinn, “Understanding mathematical discourse”, in *Proceedings Amstelogue '99 Workshop on Semantics and Pragmatics of Dialogus*, University of Amsterdam, 1999.
- “Axiom of Replacement in Category Theory”, *Ars Mathematica* blog, <http://www.arsmathematica.net/archives/2010/11/14/axiom-of-replacement-in-category-theory/>.

