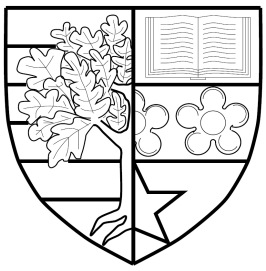
FROM FORMAL SPECIFICATION TO FULL PROOF: A STEPWISE METHOD

by

Lavinia Burski



Submitted for the degree of
Doctor of Philosophy

Department of Computer Science
School of Mathematical and Computer Sciences
Heriot-Watt University

March 2016

Acronyms

ASM Abstract state machine.

CGa Core Grammatical aspect.

DRa Document Rhetorical aspect.

GPSa General Proof Skeleton aspect.

Gpsa General Proof Skeleton aspect.

GpsaOL General Proof Skeleton ordered list.

Hol-Z Hol-Z.

IEC International Electrotechnical Commission.

MathLang MathLang framework for mathematics.

PPZed Proof Power Z.

SIL Safety Integrity Levels.

SMT Satisfiability Modulo Theories.

TSa Text and Symbol aspect.

UML Unified Modeling Language.

UTP Unifying theories of programming.

ZCGa Z Core Grammatical aspect.

ZDRa Z Document Rhetorical aspect.

ZMathLang a toolkit for checking various degrees of correctness for Z specifications.

Glossary

computerisation The process of putting a document in a computer format.

formal methods Mathermatically rigorous techniques and tools for the specification, design and verification of software and hardware systems.

formalisation The process of extracting the essence of the knowledge contained in a document and providing it in a complete, correct and unambiguous format.

halfbaked proof The automatically filled in skeleton also known as the Half-Baked Proof.

partial correctness A total correctness specification [P] C [Q] is true if and only if, whenever C is executed in a state satisfying P and if the execution of C terminates, then the state in which Cs execution terminates satisfies Q.

semi-formal specification A specification which is partially formal, meaning it has a mix of natural language and formal parts.

total correctness A total correctness specification [P] C [Q] is true if and only if, whenever C is executed in a state satisfying P, then the execution of C terminates, after C terminates Q holds.

Chapter 1

Background

Formal methods are a specific type of mathematical notation which is based on the techniques of the specification, verification and development of software and hardware systems [2]. Since our thesis presents a toolkit for checking various degrees of correctness for Z specifications (ZMathLang) we go right to the beginning of the framework, to describe how mathematical notation came about. Then we describe the original MathLang framework (the framework which ZMathLang is an adaptation of) and then give the reader an idea of other formal methods and languages. In the next section we wish to describe what is the language of Z and give more details of it's syntax and semantics. We then highlight other proving techniques which have been done for maths, formal methods and Z.

1.1 Mathematical Notations

Computer science (and thus computer systems) have evolved from basic mathematics. We can say that formal specification writers are practicing mathematicians as they write system specifications in a formal manner. Therefore we must start right at the beginning at the foundation of mathematical notation.

1.1.1 Right from the begininning

The relationship between mathematical reasoning and practicing mathematicians started out early on during the ancient Greeks where logic was already being studied.

Reasoning in logic was used for just about anything not just mathematics such as law, medicine and farming. This very early form of mathematics made very famous discoveries such as Aristotles logic [11], Euclid's geometry [4] and Leibniz Calculus [7].

Further on in the 1800's, Frege wrote *Die Grundlagen der Arithmetik* [5] and other works where he noted that mathematics is a branch of logic. In this works, he began building a solid foundation for mathematics. This early foundation along with Cantors set theory [3] was argued to be incosistant and thus Russel found a pardox in this work.

In the late 19th century and beginning of 20th century, Russell & Whitehead [10] started to form a basis for mathematical notation. Their three volume work describes a set of rules from which all mathematical truths could be proven. In these early stages the authors try to dervive all maths from logic. This ambitious project was the first stepping stone in collaborating all mathematics under one notation.

Further to Russell & Whitehead's work, Bourbanki[1] wrote a series of books beginning n the 1935's with the aim of grounding mathematics. Their main works is included in the Elements of Mathematics series [1] which does not need any special of knowledge of mathematics. It describes mathematics from the very beginning and goes through core mathematical concepts such as set theory, algebra, function etc and gives complete proofs for these concepts.

Adding to Russell's work, Zermelo introduced an axiomatisation of set teory which was later extended by Frankel and Skolem to form ZF set theory [9]. This new theory is what we will later see the Z notation is based on and the notation this thesis checks the correctness of.

1.1.2 Computerisation of Maths and Proof Systems

- typesetting systems like LaTeX
- proof assistants and automated theorem provers
- Semantical oriented document representations like OpenMath and OMDoc

[1] A name given to a collective of mathematicians

1.1.3 Conclusion

In summary....

1.2 MathLang for mathematics

Intro....

1.2.1 Overview and Goals

1.2.2 Detailed information on CGa

- Reference Zenglars quote
- Weak type theory into CGa

1.2.3 Detailed information on DRa

- relations
- instances
- Dependency and goto graph

1.2.4 Detailed information on skeletons

- General Proof Skeleton
- Half baked proof
- Filled in skeleton

1.2.5 information on TSa

TSa is used to formalise mathematical texts, mathematicians write differently eg ‘a=b=c‘ is the same as ‘a=b and b=c‘. We do not need tsa as formal spec are already written formally.

1.2.6 A full worked examples in mathlang

show step by step translation of mathematical text into isabelle from laamars phd thesis.

1.2.7 Conclusion

1.3 Formal Methods and Languages

- definitions of 'formal language', 'formal method' and 'formal specification'
- the first formal language is thought to be used by Frege in his Begriffsschift (1879), Begriffschift meaning 'concept of writing' described as 'formal language of pure thoguht'
- broad histroy of formal methods
  - 1940's, Alan Turing annotated the properties of program states to simplify the logical analysus of sequential programs
  - 1960's Floyd, Hoare and Naur recommended using axiomatic techniques to prove programs meet their specification.
  - 1970's Dijkstra used formal caluculus to aid development of non-deterinist programs
- Formal methods today
- why use formal methods in industry (design errors like Therac-25 1985, NASAs Checkout Launch and Control System (CLCS) cancelled 9/2002, , added level of rigor)
- types of formal methods (Z, B method, ABS)
- Success of formal methods (B27 Traffic Control System, SHOLIS project, Data Acquisition, Monitoring and Commanding of Space Equipment)
- Weakness of formal methods (Low-level ontologies, Limited Scope,Cost,Poor tool feedback)

- What needs to be done to make formal methods industrial strength?
  - Bridge gap between real world and mathematics
  - Mapping from formal specifications to code (preferably automated)
  - Patterns identified
  - Level of abstraction should be supported
  - Tools needed to hide complexity of formalism
  - Provide visualization of specifications
  - Certain activities not yet formulizable methods
  - No one model has been identified which should be used for software)

1.3.1 Conclusion

1.4 Z Syntax and semantics

1.4.1 Why Z would work with MathLang

Bridges formal method and discrete mathematical notation.

1.4.2 Introduction to Z

Z is based on predicate Calculus, Zermelo-Frankel set theory...

Ivented by j-R Abriel, ISO standard.+ Spivey standard

1.4.3 Propositional and predicate logic

1.4.4 Sets and Types

1.4.5 Definitions

- AxDef
- Freetypes
- Schema (declarations and expressions)

1.4.6 A full example in Z

1.4.7 Conclusion

1.5 Proving systems for Z

Intro....

1.5.1 Levels of Rigor

- Level 1 represents the use of mathematical logic to specify the system.
- Level 2 uses pencil-and-paper proofs.
- Level 3 is the most rigorous application of formal methods.

1.5.2 Proving systems for maths

e.g. Mizar, Isabelle, Coq

1.5.3 Proving systems for formal method

e.g. Dafny, ALC2, PVS

1.5.4 Proving Systems specific for Z

e.g. Fuzz, Hol-z ProofPower-z

1.5.5 Other proeprties to prove

1.5.6 Conclusion

1.6 Background Conclusion

1.6.1 MathLang for Z

- [8] states what ro do to make formal methods industrial strength

- [6] stating in future work mathlang should be developed to cope with more mathematics (formal spec is a type of mathematics)
- diagram of math text to theorem prover using mathlang + diagram of specification to theorem prover using mathlang

ZMathLang covers items 1, 3, 5, 6, 7 from section 1.3.

Bibliography

[1] N. Bourbaki. *General topology. Chapters 1-4.* Elements of mathematics. Springer-Verlag, Berlin, Heidelberg, Paris, 1989. Trad. de : Topologie gnrale chapitres 1-4.

[2] R. Butler, G. Hagen, J. Maddalon, C. Muñoz, A. Narkawicz, and G. Dowek. How formal methods impels discovery: A short history of an air traffic management project. In C. Muñoz, editor, *Proceedings of the Second NASA Formal Methods Symposium (NFM 2010), NASA/CP-2010-216215*, pages 34–46, Langley Research Center, Hampton VA 23681-2199, USA, April 2010. NASA.

[3] G. Cantor. Ueber eine Eigenschaft des Inbegriffs aller reellen algebraischen Zahlen. *Journal fr die reine und angewandte Mathematik*, 1874(77):258–262, 1847.

[4] R. Fitzpatrick and J. Heiberg. *Euclid's Elements.* Richard Fitzpatrick, 2007.

[5] F. Gottlob. Die grundlagen der arithmetik. *Eine logisch mathematische Untersuchung ber den Begriff der Zahl, Bres*, 292, 1884.

[6] R. Lamar. *A Partial Translation Path from MathLang to Isabelle.* PhD thesis, Heriot-Watt University, 2011.

[7] L. Mastin. 17th century mathematics - Leibniz. `http://www.storyofmathematics.com/17th_leibniz.html`, 2010.

[8] C. V. Stringfellow. Formal methods presentation. September 2016.

[9] E. W. Weisstein. Zermelo-fraenkel set theory.

[10] A. Whitehead and B. Russell. *Principia Mathematica*. Number v. 2 in Principia Mathematica. University Press, 1912.

[11] J. Woods. *Aristotle's Earlier Logic (2nd Edition)*. College Publications, 2014.