New Foundations for Separation Logic

Hiep, H.A.

New Foundations for Separation Logic

H.A. Hiep

May 23, 2024

ISBN 978-90-831826-1-2 (paperback)
ISBN 978-90-831826-2-9 (e-book, PDF without DRM)
ISBN 978-90-831826-3-6 (digital artifact)
NUR 123 Exacte vakken en informatica (hoger onderwijs)
Engelstalig

Omslag: Ulysses and the Sirens, John William Waterhouse (Google Art Project)
Typografie door auteur zelf met behulp van LaTeX 2_ε.

Uitgave: eerste uitgave
Oplage: 64
Aantal pagina's: 256
Drukkerij: proefschriftenprinten.nl (Kelvinstraat 27, 6716 BV, Ede)

New Foundations for Separation Logic

Proefschrift

ter verkrijging van
de graad van doctor aan de Universiteit Leiden,
op gezag van rector magnificus prof.dr.ir. H. Bijl,
volgens besluit van het college voor promoties
te verdedigen op donderdag 23 mei 2024
klokke 13:45 uur

door

Hans-Dieter Anton Hiep
geboren te Hoorn
in 1991

Promotor:
prof.dr. F.S. de Boer

Co-promotores:
dr. C.P.T. de Gouw (Open Universiteit)
dr. A.W. Laarman

Promotiecommissie:
prof.dr. J.-P. Katoen (Universiteit Twente, RWTH Aken)
dr. J.A. Pérez (Rijksuniversiteit Groningen)
dr. H. Basold
prof.dr. M.M. Bonsangue
prof.dr. H.C.M. Kleijn
prof.dr. A. Plaat

Part of the IPA Dissertation Series: No. 2024-04

The research presented in this dissertation was carried out at the Dutch national research laboratory for mathematics and computer science Centrum Wiskunde & Informatica (CWI) in Amsterdam, and the Leiden Institute of Advanced Computer Science (LIACS) of Leiden University, under the auspices of the research school IPA (Institute for Programming research and Algorithmics).

The author was partially supported by funding from NGI ASSURE, a fund established by NLnet with financial support from the European Commission's Next Generation Internet programme, under the aegis of DG Communications Networks, Content and Technology under grant agreement No. 957073.

Abstract

This thesis presents new foundations for separation logic, an important field within the formal sciences such as theoretical computer science. Around the turn of the millennium, separation logic was introduced by J.C. Reynolds with the goal to make reasoning, about the correctness of computer programs that work with so-called 'pointers', more efficient than earlier formal methods. Reynolds' method and the other earlier methods are both extensions of the proof system introduced by C.A.R. Hoare for reasoning about the correctness of simple **while**-programs. The essence of Reynolds' initial work, which was researched and put into practical use by many other scientists, consists of two extensions of the work initiated by Hoare: the first extension adds to first-order predicate logic two new propositional connectives (the so-called separating conjunction and separating implication); the second extension adds to Hoare's program logic new proof rules for reasoning about (the primitive operations of) pointer programs. These primitive operations are used for reading memory ('lookup'), writing memory ('mutation'), reserving memory ('allocation'), or freeing memory ('deallocation').

The new foundations are presented in two parts. The following paragraphs summarize the contents of these two parts. The first part contains a model-theoretic and proof-theoretic exploration of the classical interpretation of separation logic, the logic used in Reynolds' assertion language. This first part proves a result for separation logic, that is as fundamental as the corresponding result by Gödel in first-order logic—the completeness theorem. The second part contains a new interpretation of Reynolds' program logic, and introduces—for the first time—so-called dynamic separation logic. Dynamic separation logic is an extension of dynamic logic by D. Harel. Using dynamic separation logic, an alternative weakest precondition axiomatization and strongest postcondition axiomatization is given. These alternative axiomatizations, in contrast to earlier axiomatizations of Reynolds' program logic, do have the property of *gracefulness*: the earlier axiomatizations of Reynolds' program logic are not graceful, because they unnecessarily increase the complexity in the use of separating connectives when generating weakest preconditions or strongest postconditions.

Chapter 2 of the first part demonstrates the inadequacy of the standard interpretation of separation logic, because it lacks compactness. This chapter also introduces a new interpretation, called the full interpretation of separation logic, that is based on the possibility of evaluating formulas also with respect to infinite heaps. However, also this full interpretation is inadequate. We continue with a search for necessary and sufficient conditions for embedding the standard interpretation into the full interpretation, and we introduce so-called relational separation logic with the goal to compare separation logic with second-order predicate logic. It is an interesting fact that the full interpretation of separation logic lies close to the standard interpretation of second-order logic: expressivity of a so-called binding operator is sufficient for showing that these two logics coincide.

Chapter 3 of the first part introduces a proof theory with a corresponding new interpretation of separation logic that is based on first-order definable heaps. This new interpretation allows us to show that the resulting proof system is in fact adequate. The proof system is presented as a sequent calculus, but also a second proof system is introduced in the natural deduction style. The sequent calculus is sound and complete with respect to first-order definable heaps. The natural deduction calculus is sound and complete with respect to structures that satisfy a semantic comprehension condition. The second proof system works with more general formulas than what is allowed in separation logic, by introducing a connective that is closely related to the binding operator of the previous chapter.

Chapter 4 of the second part comprises: general interpretations of separation logic, and a class of structures based on so-called memory models. The latter class is used in the proof of soundness and relative completeness of Reynolds' program logic. We arrive at dynamic separation logic by introducing a program modality in the assertion language. We research an alternative weakest precondition axiomatization and strongest postcondition axiomatization for classical separation logic. This work directly leads us to solving an open problem, where it is the question whether the global axioms can be inferred from the local axioms and the frame rule of Reynolds' program logic but without additionally using the separating implication connective. Our method is robust, since it can also be applied to obtain a weakest precondition axiomatization and strongest postcondition axiomatization for intuitionistic separation logic (this result is not yet published).

The thesis also includes an extensive appendix with background material, concerning higher-order predicate logic and Hoare's program logic, that is needed to understand and appreciate the above novel results. The appendix also describes an accompanying Coq formalization of the soundness and completeness of the alternative axiomatizations of Reynolds' program logic, that aims to increase the trust one may place in the validity of the results.

Preface

This thesis is the result of my promotion trajectory, that ran from 1 November 2018 until 31 October 2023 (5 years), executed at two institutes: Centrum Wiskunde & Informatica (CWI) in Amsterdam from 1 November 2018 until 31 October 2020 (2 years); and Leiden Institute of Advanced Computer Science (LIACS) in Leiden from 1 November 2020 until 31 October 2023 (3 years). At the CWI, I was part of the Formal Methods group (FM), and at LIACS, I was part of the theory group. The CWI provided me with office space for the entire duration of the trajectory. The promotion was initiated by Frank de Boer (promotor) and Stijn de Gouw (co-promotor), with the initial goal of verifying standard libraries of the Java programming language using the KeY system.

The first years were quite productive and this would not be possible without Frank giving me a lot of freedom to explore, to develop independently, and to take initiative. We often had productive meetings, and structured our collaboration by means of writing papers together. I also collaborated with Stijn's master student, Olaf Maathuis, while we were both learning how to use the KeY system to verify Java's `LinkedList` class. Jinting Bian joined our group at CWI, and I helped her with learning how to use KeY so we could collaborate on Java library verification.

In these initial years I also submitted grant proposals, and collaborated with Benjamin Lion, Kasper Dokter, Roy Overbeek, and Farhad Arbab. Some of the grant proposals were accepted: a new project with the code name Reowolf started that was later extended in a project named Reowolf 2.0. As part of these projects we were able to hire scientific programmers, Christopher Esterhuyse and Max Henger, with whom I collaborated on completing the deliverables of the projects. Christopher convinced me to use the Rust programming language for the project, which I had not used before. This allowed me to gain more practical experience with programming under a linear typing discipline. Max has taught me to take the concerns of efficiency and scalability more seriously than I did before.

From the start, I was involved in teaching in the Program Correctness course at Leiden University. In this course we explain Hoare's logic to bachelor students, both for simple **while**-programs and programs with recursive procedures, and we practice with a simplified version of the KeY system. Later on, teaching both the Program Correctness and Concepts of Programming Languages courses also became my responsibility. For the latter course, I redeveloped the course material and recorded an on-line lecture video series.

After years of progress in this initial direction and several semesters of involvement in teaching, Frank invited me to collaborate on an article, submitted to the Association for Computing Machinery (ACM) journal Transactions on Programming Languages and Systems (TOPLAS), on the completeness and complexity of a Hoare-like logic for reasoning about call-by-value procedures. For that article, I contributed the Coq formalization proving several tricky supporting lemmas and came up with the idea of applying techniques from proof theory to do proof normalization in Hoare's logic to prove the complexity result that correct programs have linear proofs. On my own initiative, I presented the basis of this work at the PhD Day organized by the VvL (the Dutch Association for Logic and Philosophy of the Exact Sciences) on July 1st, 2022.

Afterwards, I wanted to change the direction of my own research towards the investigation of the foundations of separation logic (while continuing collaboration with Jinting Bian on verifying Java libraries, continuing work on the Reowolf project, and continuing teaching Concepts of Programming Languages and Program Correctness). There were several reasons for considering this change of direction: firstly, we received numerous anonymous reviews in response to our earlier articles about Java program verification that mentioned separation logic as related work. Secondly, on several occasions Frank indicated he was a contrarian in this subfield, of separation logic, so I was inclined to become a *meta-contrarian*[1]. Lastly, I had many interesting discussions about separation logic during the PhD Day organized by the VvL. Although at that time I had only superficial knowledge, I started to wonder: what the hell is separation logic, really? Already, I had done several years of work of a practical nature, in actual Java program verification, and in the mean time I had learned more about higher-order logic, set theory, model theory, proof theory, and foundational issues in mathematics. Now I wanted to do more work of a theoretical nature in mathematical logic, and continue my work on formalizing Hoare's logic.

So I convinced Frank that it was a good idea to investigate separation logic. Our approach would be from a foundational point of view, and my goal was to understand what were the issues in separation logic that Frank refrained from articulating in the past twenty years or so. What emerged was a symbiotic relationship between me and my promotor: I gladly took Frank as a confident oracle, and saw myself as a skeptical verifier. In this period we worked together intensively, often spending many hours a day discussing next to a whiteboard. Also I used the Coq proof assistant to meticulously check my work. But at other times, the roles reversed, and I saw myself as the oracle while Frank was verifying my 'nonsense', critically and skeptically questioning my position until we obtained something reasonable. The benefit of our symbiosis was that we discovered many of our own mistakes, that we were able to repair ourselves. As such, I was deeply involved in the discovery, the refinement, the verification, and the presentation of the subject matter that is presented in this thesis. Frank and I collaborated on a

[1]Thanks to Benjamin for explaining to me why I am an 'intellectual hipster': whereas Frank is a contrarian (i.e. opposing separation logic), I took an opposite position in Frank's contrariness (thus opposing Frank's opposition, in defense of separation logic).

paper until no longer there would be any ground to oppose each other, and then we involved my first co-promotor, Stijn, to check the intermediate paper—whether what we did made sense. Finally, after this thesis was written, also my second co-promotor, Alfons Laarman, was involved to check whether the thesis as a whole made sense. I found this way of working to be very productive.

<rant> Whereas the collaboration between me and my promotor and co-promotors was very productive, I found that there was also a source of counter-productivity: the anonymous reviewers of our articles about separation logic. As mentioned earlier, we had structured our collaboration by means of writing articles that were submitted for presentation at several conferences. However, that last part, submission to conferences and subjecting our articles to objective anonymous reviewers, was severely frustrating our productivity. Contrary to the positive symbiotic relation between me and my promotor, I had experienced the relation between me and my anonymous reviewers as negative, even alienating. It felt I was in a toxic burn pit that slowly burned me out. The epitome of toxicity was when a reviewer was rejecting, insistingly, our article on the basis of a counter-example to our result, that was also a counter-example to Gödel's completeness theorem! Communication with anonymous reviewers was very limited and the reviewers did not respond to requests to discuss the matter further.</rant> In an attempt to prevent any confusion about what the background material is, I spent several months writing the appendix.

In the end, I am deeply indebted to Frank for his encouragement: to continue to defend these new foundations for separation logic, and to regard less of negative and discouraging comments by anonymous reviewers. I am also delighted by the fact that Stijn and Alfons were always able to give useful and constructive feedback on papers or this thesis. I am grateful to all the members of the PhD committee for reading a preliminary version of this thesis and giving valuable feedback that has lead to an improved and final version.

Acknowledgments

I wish to thank, besides the people already mentioned in the preface, also the following people for providing a most pleasant working environment and/or their (direct or indirect) support of me while I was working on this thesis:

Vlad Serbanescu, Keyvan Azadbakht, Jana Wagemaker, Jurriaan Rot, Jan Rutten, Luc Edixhoven, Sung-Shik Jongmans, Carl Schulz, Maarten Dijkema, Marco Floor, Henk Roose, Emil Gorter, Vera Sarkol, Annette Kik, Minnie Middelberg, Nada Mitrovic, Doutzen Abma, Dick Broekhuis, Margriet Brouwer, Martine Anholt Gunzeln, Susanne van Dam, Ramona Rijff, Remco Westra, Krzysztof Apt, Jos Baeten, Erik de Vink, Marten van Dijk, Chenglu Jin, Niloufar Sayadi, Chao Yin, Sirui Shen, Steven Pemberton, and all other collegues at the CWI;

Mike Preuss, Walter Kosters, Hendrik Jan Hoogeboom, Rudy van Vliet, Jeannette de Graaf, Mitra Baratchi, Frank Takes, York-kam Kwok, Esme Caubo, Joyce Glerum, Riet Derogee, Chris Flinterman, Caroline de Bruin, Hui Feng, Lieuwe Vinkhuijzen, Luc Edixhoven, Dalia Papuc, Tanjona Ralaivaosaona, Miguel Blom, Tim Coopmans, Sebastiaan Brand, and all other collegues at LIACS;

Wan Fokkink, Jasmin Blanchette, Herbert Bos, Robbert Krebbers, Frits Vaandrager, Marieke Huisman, Loek Cleophas, Thomas Neele, the late Eelco Visser, Wolfgang Ahrendt, Reiner Hähnle, Bernhard Beckert, Richard Bubel, Mattias Ulbrich, Einar Broch Johnsen, Silvia Lizeth Tapia Tarifa, Volker Stolz, Violet Ka I Pun, Crystal Chang Din, Dominic Steinhöfel, Eduard Kamburjan, Michael Kirsten, Alexander Weigl, Lars Tveito, Asmae Heydari Tabar, Alexander Knüppel, Michiel Leenaars, Mirko Ross, Stephen Farrell, Chris Verhoef, Bas van Bockel, and all other (international) collegues I worked with;

Wesley Shann, Daniel Roos, Lazlo de Wijs, Oualid Azzeggarh, Zahir Bingen, Andy Tatman, Wessel van der Goot, Renz Roos, Roos Wensveen, Dominique Lawson, Perri van den Berg, Dirck van den Ende, and all my other students;

Joris Bierkens, David van Oldenhof, Anders Rehult, Diederik Malien, Xavier Boot, Jim Lemmers, Suzanne Kraaij, Laurens van Kempen, Micha Klamer, Koen van Veen, Eric Ruts, Jacob Kooijman, Alan Hopman, Thijs Louwman, Maarten Dinkelberg, Thijs Dekker, and all my other friends;

Mara and Pandora Visser, Jorien van den Heuvel, my brother, and my parents;

and all others I forgot to mention here.

Contents