

Implementing Separation Logic using an SMT-backed Frame Rule

Kirill Golubev

Alcides Fonseca

alcides@ciencias.ulisboa.pt

LASIGE, Faculdade de Ciências da Universidade de Lisboa

Lisboa, Portugal

Abstract

Symbolic execution is a technique frequently used to reason about code. In symbolic execution, the analyser keeps track of a logical representation of state, and correctness verifications are SMT queries. Separation logic is frequently used to express and verify properties of programs with pointers or references. However, most SMT solvers (like the popular z3) do not support Separation Logic natively. CVC5 has introduced partial support for separation logic, which has not yet been integrated into a more high-level tools.

This work aims to address this gap, by providing a proof of concept for implementing the Frame Rule using SMT queries in the Symbolic Heap fragment of separation logic, supported by CVC5. We conclude that this encoding can simplify the machinery dealing with separation logic, such as that present in Viper, Smallfoot, and others.

CCS Concepts: • Computer systems organization → Embedded systems; Redundancy; Robotics; • Networks → Network reliability.

ACM Reference Format:

Kirill Golubev and Alcides Fonseca. 2018. Implementing Separation Logic using an SMT-backed Frame Rule. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation email (Conference acronym 'XX)*. ACM, New York, NY, USA, 1 page. <https://doi.org/XXXXXXX.XXXXXXX>

1 Introduction

Todo ▶ Introduce the topic of Program Verification.◀

Todo ▶ Introduce the topic of Separation Logic◀

Todo ▶ Mention tools that use separation logic, and describe how they are encoded.◀

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

Conference acronym 'XX, June 03–05, 2018, Woodstock, NY

© 2018 Association for Computing Machinery.

ACM ISBN 978-1-4503-XXXX-X/18/06...\$15.00

<https://doi.org/XXXXXXX.XXXXXXX>

Todo ▶ Describe the partial support of SL in CVC5, and identify that z3 has no such support.◀

Todo ▶ Describe the goal and contribution of this paper.◀

2 Frame Rule

Todo ▶ Describe the algorithm and how it works.◀

3 Conclusions

Todo ▶ Describe how this can be used in other contexts, and what is the advantage.◀

References

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009