sujet de stage de Master 2 recherche (2024)

Reasoning on the execution time of programs with Kleene Algebra with Tests

Advisor: Patrick Baillot (patrick.baillot@univ-lille.fr)
Website: `https://pro.univ-lille.fr/patrick-baillot/`

Research lab: CRIStAL, Université de Lille, Équipe *SyCoMoRES*
Lab website: `https://www.cristal.univ-lille.fr/?lang=fr`

Duration: 4-6 months
Keywords: Kleene Algebras with Tests, Hoare logic, equational logic, verification

Context: In this internship we propose to analyse the execution time of imperative programs with equational reasoning, by adopting the approach of Kleene algebra with tests.

The theory of Kleene algebra is a generalization of regular expressions which are well-known to be equivalent to finite automata, by Kleene's theorem. In order to use Kleene algebra to reason on imperative programs it is useful to add a notion of test, so as to be able to model control flow (conditional and while program constructs). This leads to the notion of Kleene algebra with tests (KAT), introduced by Kozen [Koz97]. The axioms of KAT combine those of Kleene algebra and those of Boolean algebra. KAT allows to prove properties on the input/output behaviour on programs of a while-language in a simple way, by means of equational reasoning. For instance some equivalences of programs or the validity of some program transformations can be verified in this way [Koz00, KP00]. Various extensions of KAT have also been proposed in the literature, e.g. [AFG^{+}14, SFH^{+}20, GMB19].

Hoare logic is a classic and wide-spread approach for reasoning on imperative programs. It handles judgements of the form $P\ \{c\}\ Q$ where c is a program, and P (resp. Q) is a precondition (resp. postcondition) on memory, expressed in a suitable logic. The meaning of such a judgement is that *if a memory m satisfies predicate P and c's execution on m terminates with final memory m', then m' satisfies Q*. Kozen has shown that KAT is as expressive as (propositional) Hoare logic, by defining an encoding of the latter into the former [Koz00].

The standard presentation of Hoare logic only allows to prove input-output properties of programs (also called *extensional properties*). However some works have introduced extensions of Hoare logic for reasoning on non-extensional properties, such as computation time, probabilistic properties of randomized programs etc (see [GKOS21])

Objectives: We propose to define and study a generalization of KAT which would allow to reason on the computation time of imperative programs. For that we can take inspiration from the recent extension of KAT that we proposed in [GBG23] and which allows to reason on some probabilistic properties of randomized programs. Concretely it will take the form of a pair of a KAT and a partially ordered monoid, with a set of axioms on judgements. One will also need to consider a suitable semantics, which can be a trace semantics. We will want to prove that the extension of KAT will be expressive enough to be able to encode a Hoare logic for execution time [GKOS21].

Expected background: Some familiarity with first-order logic and its models, as well as with regular expressions is expected. Some basic knowledge about denotational semantics and Hoare logic for imperative programs would also be appreciated.

References

[AFG+14] Carolyn Jane Anderson, Nate Foster, Arjun Guha, Jean-Baptiste Jeannin, Dexter Kozen, Cole Schlesinger, and David Walker. Netkat: semantic foundations for networks. In Suresh Jagannathan and Peter Sewell, editors, *The 41st Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL '14, San Diego, CA, USA, January 20-21, 2014*, pages 113–126. ACM, 2014.

[GBG23] L. Gomes, P. Baillot, and M. Gaboardi. A Kleene algebra with tests for union bound reasoning on probabilistic programs. Technical report, HAL, September 2023. https://hal.science/hal-04196675.

[GKOS21] Marco Gaboardi, Shin-ya Katsumata, Dominic Orchard, and Tetsuya Sato. Graded hoare logic and its categorical semantics. In Nobuko Yoshida, editor, *Programming Languages and Systems - 30th European Symposium on Programming, ESOP 2021, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2021, Luxembourg City, Luxembourg, March 27 - April 1, 2021, Proceedings*, volume 12648 of *Lecture Notes in Computer Science*, pages 234–263. Springer, 2021.

[GMB19] L. Gomes, A. Madeira, and L. S. Barbosa. Generalising KAT to verify weighted computations. *Scient. Annals of Comp. Sc.*, 29(2):141–184, 2019.

[Koz97] D. Kozen. Kleene algebra with tests. *ACM Trans. on Prog. Lang. and Systems*, 19(3):427–443, 1997.

[Koz00] D. Kozen. On Hoare logic and Kleene algebra with tests. *ACM Trans. on Comp. Logic*, 1(212):1–14, 2000.

[KP00] Dexter Kozen and Maria-Christina Patron. Certification of compiler optimizations using kleene algebra with tests. In John W. Lloyd, Verónica Dahl, Ulrich Furbach, Manfred Kerber, Kung-Kiu Lau, Catuscia Palamidessi, Luís Moniz Pereira, Yehoshua Sagiv, and Peter J. Stuckey, editors, *Computational Logic - CL 2000, First International Conference, London, UK, 24-28 July, 2000, Proceedings*, volume 1861 of *Lecture Notes in Computer Science*, pages 568–582. Springer, 2000.

[SFH+20] Steffen Smolka, Nate Foster, Justin Hsu, Tobias Kappé, Dexter Kozen, and Alexandra Silva. Guarded Kleene algebra with tests: verification of uninterpreted programs in nearly linear time. *Proc. ACM Program. Lang.*, 4(POPL):61:1–61:28, 2020.