Encoding Kleene Algebra (with tests) in Coq *

Nelma Moreira[1], David Pereira[1**] and Simão Melo de Sousa[2]

[1] DCC-FC & LIACC – University of Porto
Rua do Campo Alegre 1021, 4169-007
Porto, Portugal
{nam,dpereira}@ncc.up.pt
[2] LIACC & DI – University of Beira Interior
Rua Marquês d'Ávila e Bolama
6201-001 Covilhã, Portugal
desousa@di.ubi.pt

Abstract

Kleene algebra [1], (KA) normally called *the algebra of regular events*, is an algebraic system that axiomatically captures properties of several important structures arising in Computer Science, and has been applied in several contexts like automata and formal languages, semantics and logic of programs, design and analysis of algorithms, among others. *Kleene algebra with tests* (KAT) [2] extends KA with an embedded *Boolean algebra* and is particularly suited for the formal verification of propositional programs. In particular, KAT subsumes *propositional Hoare logic* (PHL) [3], a weaker Hoare logic without the assignment axiom. This part of our formalization is described in detail by Pereira and Moreira in [4].

Here we describe a formalization of a fragment of formal languages in the Coq theorem prover. This formalization's goal is to provide a Coq library that contains proof tactics for automatically proving equivalence of KA and KAT's equational logics. Having these tactics available requires the codification of KA and KAT, and also providing proofs that they are complete for their standard models, that is, regular languages and Kozen's *automata on guarded strings* [5], respectively. In order to provide a proof that regular languages are a model of KA, we have encoded regular languages, by extending Coq's *Ensembles* library of basic set theory with new inductive types for the concatenation and Kleene's star operations, based in the work of J.C. Filliâtre [6].

In what concerns to KAT, besides the Coq modules describing KAT's signature and of proofs of its main properties, we have encoded PHL deductive rules as KAT expressions and proved that they are KAT theorems. We have also proved correct an annotated version of PHL's deductive rules in our framework.

Currently, we are implementing a decision procedure for the equivalence of KA terms, that leads to a decidable procedure for the equational theory of KA, based

* This work was partially funded by Fundação para a Ciência e Tecnologia (FCT) and program POSI, and by RESCUE project PTDC/EIA/65862/2006.
** David Pereira is funded by FCT grant SFRH/BD/33233/2007

on the notion of Brzozowski's *derivative* [7] of a regular expression. This approach differs from the standard approach for deciding regular expression equivalence in the sense that it does no rely on comparing the minimal deterministic automata corresponding to the regular expressions being tested. We have encoded the notion of derivative of a regular expression and also proved that the derivative of a regular expression correspond to the left-quotient of the language of the original regular expression. We are currently proving that the number of derivatives of a set of regular expressions modulo ACI (associativity, commutativity and idempotence) is finite. This proof will then serve as an argument for a general recursive function that implements Brzozowski's decision procedure [8]. Since this decision procedure cannot be described by a structurally recursive function, we don't have program termination for free. In Coq, a standard solution is to use as an artificial argument that is structurally decreasing and that reflects the behaviour of the decision procedure. In particular, we are interested in using the known upper-bounds of the number of derivatives of a regular expression to be such argument. We intend to extend this procedure to KAT by using Kozen's co-algebraic approach [9], where derivatives of regular expressions were extended to KAT.

We are also particularly interested in *Schematic* KAT (SKAT) [10], a specialization of KAT involving an augmented syntax to handle first-order constructs and restricted semantic actions whose intended semantics coincides with the semantics of first-order *flowchart schemes* over a ranked alphabet Σ. SKAT programs can be transformed into KAT expressions, by converting SKAT's logical constructs into KAT Boolean tests, and converting SKAT variable assertions to KAT program symbols. In this setting, we can prove the correctness of programs using full first-order Hoare logic within our formalization, by manually converting SKAT programs into KAT expressions. We intend to automatize this task, following the lines of Aboul-Hosn and Kozen in the development of the KAT-ML [11] interactive theorem prover.

Our motivation for this work comes from the fact that we envision the usage of (an extension of) our formalization as the formal system where we can be encode and prove *proof obligations* in the context of *Design by Contract* [12] for the *Proof Carrying Code* [13] paradigm.

References

1. Kleene, S. In: Representation of Events in Nerve Nets and Finite Automata. Shannon, c. and mccarthy, j. edn. Princeton University Press, Princeton, N.J. 3–42
2. Kozen, D.: Kleene algebra with tests. Transactions on Programming Languages and Systems **19**(3) (May 1997) 427–443
3. Kozen, D., Tiuryn, J.: On the completeness of propositional Hoare logic. In: RelMiCS. (2000) 195–202
4. Pereira, D., Moreira, N.: KAT and PHL in Coq. Computer Science and Information Systems **05**(02) (December 2008) ISSN: 1820-0214.
5. Kozen, D.: Automata on guarded strings and applications. Technical report, Cornell University, Ithaca, NY, USA (2001)

6. Filliâtre, J.C.: Finite Automata Theory in Coq: A constructive proof of Kleene's theorem. Research Report 97–04, LIP - ENS Lyon (February 1997)
7. Brzozowski, J.A.: Derivatives of regular expressions. JACM **11**(4) (October 1964) 481–494
8. Brzozowski, J.A.: Roots of star events. Journal of the ACM (JACM **14**(3) (Jul 1967)
9. Kozen, D.: On the coalgebraic theory of Kleene algebra with tests. Computing and information science technical reports, Cornell University (March 2008)
10. Angus, A., Kozen, D.: Kleene algebra with tests and program schematology. Technical Report TR2001-1844, Cornell University (2001)
11. Aboul-Hosn, K., Kozen, D.: KAT-ML: An interactive theorem prover for Kleene algebra with tests. Journal of Applied Non-Classical Logics **16**(1–2) (2006) 9–33
12. Meyer, B.: Applying "design by contract". Computer **25**(10) (1992) 40–51
13. Necula, G.C.: Proof-carrying code. In: POPL '97: Proceedings of the 24th ACM SIGPLAN-SIGACT symposium on Principles of programming languages, New York, NY, USA, ACM (1997) 106–119