Decision Procedure for Equivalence Relations

Internship Supervisors	Pierre Corbineau `Pierre.Corbineau@univ-grenoble-alpes.fr` Lionel Rieg `Lionel.Rieg@univ-grenoble-alpes.fr`
Research Team	Verimag Lab, Formal Proofs theme `http://www-verimag.imag.fr` Université Grenoble Alpes, Grenoble, France
Collaborations	Karine Altisen, Université de Genève, Geneva, Switzerland
Keywords	Logic & Verification, Automated reasoning, Proof assistants

Scientific Context Automated reasoning is a widely-used technique for system and program verification, where problems are translated into logical formulas. These formulas can be checked by automated theorem provers. However, an automated prover may fail to answer a specific problem because of undecidability.

By contrast, interactive proof assistants allow the user to manually guide the proof process. This allows for more expressive proof techniques to be used at the cost of more user time and expertise. In order to reduce the burden on the expert user, it is desirable to provide suitable automation, especially for the seemingly trivial parts of interactive proofs.

The goal of the proposed research is to design a decision procedure for the Coq/Rocq proof assistant [1] that would extend equality-based reasoning (e.g., congruence-closure) to heterogeneous problems where equalities are expressed using multiple equivalence relations.

Scientific Problem Equality reasoning is a very common paradigm for proofs both in the field of automated theorem proving and when using interactive proof assistants such as Coq/Rocq. The Coq/Rocq proof assistant comes with a natural notion of equality which encompasses the notion of *computation* within the language (mostly typed λ-calculus). When reasoning over functions, the Coq/Rocq equality captures equality of the *programming code* (as a λ-term) of the function rather than *pointwise equality* (for all inputs). However, most properties about functions are *extensional*, *i.e.*, they are in fact properties of the images of the function.

A common approach to circumvent the problem is to work in a *setoid* (that is, a set equipped with an equivalence relation): instead of using Coq/Rocq equality, one can define an *ad hoc* user-defined equality. Note that even if this relation is an equivalence for base types, this cannot be ensured for function types; we can only obtain partial (i.e., non-reflexive) equivalence relations (PER). The main consequence is that replacement of equivalent objects can only occur in a suitable context, under adequate *relation morphisms*.

For instance, if $\sim$ denotes list permutation, the fact that permutation preserves length (that is, the length function is a morphism between list permutation and equality over integers) is described by the relation $(\sim \Rightarrow =)$: $f(\sim \Rightarrow =)g := \forall l_1 l_2, l_1 \sim l_2 \implies f(l_1) = g(l_2)$. If we set $f = g =$ `length`, we exactly recover the fact that `length` is a morphism from permutation to equality.

This concept of *extensional relation* allows to state the following setoid-congruence rule:

$$\frac{f\ (R_1 \Rightarrow R_2)\ g \qquad x\ R_1\ y}{f(x)\ R_2\ g(y)}\ \text{Cong}$$

In practice, user-defined as well as Coq/Rocq equalities are very often involved in the same proofs and reasoning. The Coq/Rocq proof assistant is equipped with procedures for replacing equivalent objects under morphisms and with an efficient decision procedure for quantifier-free equality-based reasoning [2, 3, 4]. However, this mechanism lacks a procedure which could combine Coq/Rocq and user-defined equalities and provide a generic equality reasoning procedure.

Current research has shown that the full quantifier-free problem is undecidable. Therefore it is desirable to understand the limit of the decidability frontier and to identify fragments where the problem remains decidable and heuristics to apply when outside these fragments. Incidently, recent work [6] proves on paper that decidability can indeed be obtained for a fragment where enough base PERs are in fact equivalence relations (i.e. reflexive).

Proposed Work Here are the tasks proposed for the internship :

- Study the properties of the historical congruence-closure algorithm, and identify arguments for the proof of decidablity.

- Study the Rocq proof for the undecidability of deciding setoid congruence with PERs.
- Study the recent work on decidability for the fragment with reflexive base relations.
- Propose an formal definition of this fragment, with the extension to n-ary function symbols.
- Establish a formal proof in Rocq of decidability for this fragment.

Required Skills The main goal of the internship is to establish a formal proof in Coq/Rocq, therefore we expect an adequate prior knowledge of Coq/Rocq. We also expect the candidate to be familiar with basic first-order logic.

Followup research Depending on the result, this research could lead to a PhD proposal along the given subjects:

- Design/adaptation of existing algorithms to decide the previously identified decidable fragment
- Further study of decidability in this class of problems
- Design/study of heuristics to handle undecidable situations for quantifier-free problems
- Design/study of heuristics to handle undecidable situations with quantified hypotheses

Bibliography

[1] The Coq Development Team. The Coq Proof Assistant Reference Manual. `https://coq.inria.fr`.

[2] P. Corbineau. Deciding equality in the constructor theory. In *TYPES 2006 : Types for Proofs and Programs*, volume 4502 of *Lecture Notes in Computer Science*, pages 78–92. Springer-Verlag, 2007.

[3] Downey, Peter J. and Sethi, Ravi and Tarjan, Robert Endre. Variations on the Common Subexpression Problem. In *Journal of the ACM*, volume 27, issue 4, October 1980.

[4] G. Nelson and DC. Oppen. Fast decision procedures based on congruence closure In *Journal of the ACM*, volume 27, issue 2, April 1980.

[5] Sébastien Michelland. Une procédure de décision pour relations d'équivalence. Internship report, Verimag, June 2020

[6] Emilie Uthaiwat. Décider ou ne pas décider, là est la question : Comment décider si deux termes sont en relation ? Internship report, Verimag, August 2025