Algebraic Methods for Model Refinement

Roland Glück
Universität Augsburg

Abstract. The present work was stimulated by [8] and [9]. Our idea will be to algebraise the ideas present there. Therefore a quantal theory for model checking and model refinement will be introduced and some possibilities for its application will be sketched. Furthermore a short outlook on future work and open questions is given.

1 Introduction

Model refinement is, informally spoken, the art of restricting graphs. More formally, given a graph $G = (V, E)$ with nodeset $V$ and edgset $E$, one has to determine the greatest subgraph $G' = (V', E')$ of $G$ with $V' \subseteq V$ and $E' \subseteq E$ that fulfills a certain property. So, e.g. the greatest cyclic subgraph of a given graph could be the goal. In another variant of this problem one has a labelled graph $G = (V, E, l)$, where $(V, E)$ is a graph as above and $l : V \times E \rightarrow L$ is a labelling function that nodes and edges assigns values drawn from a set of labels $L$. In this case it is often required that the labelling is deterministic in the sense that every node can have at most one outgoing edge with a certain label. So the refinement can be seen as a policy which action from a given action set, modelled by $L$, has to be chosen in a certain state to achieve a desired behaviour. Moreover, the edges can be equipped with a cost function $c : E \rightarrow \mathbb{R}^+$ or even $c : E \rightarrow \mathbb{R}$, which serves to handle optimisation problems like shortest paths or maximum capacity problems.

For (labelled) graphs with a huge or even infinite number of states this turns out to be a difficult problem. The idea is to use bisimulations for solving this tasks. Therefore a coarsest bisimulation is constructed. It is an equivalence, so that a graph is obtained with the equivalence classes of this bisimulation as nodes and canonical defined edges. Then the task is solved on this (in general smaller, hopefully at least finite) instance and the result is played back in this instance. One question will be for which kind of problems this approach will work.

2 The Algebraic Setting

2.1 Quantales

As a basic tool for our algebraic setting we use quantales. They are defined as follows:
Definition 2.1 1. An idempotent semiring is a structure \( S = (M, +, 0, \cdot, 1) \) such that \( 0 \neq 1 \), \((M, +, 0)\) and \((M, \cdot, 1)\) are monoids, choice + is commutative and idempotent, and composition \( \cdot \) distributes through + and is strict in both arguments. + is called addition, \( \cdot \) is called multiplication.

2. The natural order \( \leq \) is given by \( x \leq y \iff df x + y = y \).

3. We call an element \( x \in N \) of a subset \( N \subseteq M \) atomic in \( N \) if \( x \neq 0 \) and \( \forall y \in N : y \neq 0 \land y \leq x \Rightarrow y = x \).

4. A subset \( N \subseteq M \) is atomic if every element \( x \in N \) is the supremum of the atoms of \( N \) below \( x \).

5. A quantale is an idempotent semiring that is a complete lattice under the natural order and in which composition distributes over arbitrary suprema.

The crucial point is that the set of relations \( Rel(X) \) over a set \( X \) form a quantale with union as addition, composition as multiplication, \( \emptyset \) as zero and the identity \( id_X \) as one. The previous definition lacks the possibility of modelling subsets oder single elements. This can be done using tests [6]:

Definition 2.2 The set \( test(S) \) of tests of an idempotent semiring \( S \) is the maximal Boolean subalgebra of the elements below 1. The complement of a test \( p \) w.r.t. 1 is denoted by \( \neg p \); it is the unique test \( q \) with \( p + q = 1 \) and \( p \cdot q = 0 \).

The set of atomic tests of \( S \), i.e., of atoms in \( test(S) \), is denoted by \( atest(S) \).

In the quantal \( Rel(X) \) tests are subrelation of \( id_X \); they correspond in an obvious way to subsets of \( X \). The atomic tests in \( Rel(X) \) can be used analogously to describe single elements \( x \in X \).

2.2 Using Modality

To characterise bisimulations we have to deal with converses of relations. For this purpose we introduce the following modal operators:

Definition 2.3 A modal (idempotent) semiring \( (M, +, 0, \cdot, 1, |\rangle, \langle |) \) consists of an idempotent semiring \( S = (M, +, 0, \cdot, 1) \) and the forward and backward diamond operators \( |\rangle, \langle | \colon M \to (test(S) \to test(S)) \), characterised by the following axioms (e.g. [3]): for all \( x, y \in M \) and \( p, q \in test(S) \),

\[
\begin{align*}
|x|q & \leq \neg p \iff p \cdot x \cdot q \leq 0 \iff \langle x \rangle p \leq \neg q \quad (1) \\
|x| (\langle y \rangle q) & = [x \cdot y] q \quad \langle x \rangle (\langle y \rangle q) = [y \cdot x] q \quad (2)
\end{align*}
\]

In \( Rel(X) \) the operators \( \langle x \rangle q \) and \( |x|q \) correspond to the image and the preimage of the subset of \( X \) modelled by \( q \) under the relation \( x \subseteq X \times X \). If \( R' \) is the converse relation of \( R \) the relation \( [x] = \langle x' \rangle \) holds. We will use this property soon.

Now we are ready to describe the concept of bisimulation in our setting. In classical theory a bisimulation for a relation \( \to \subseteq X \times X \) is a relation \( R \subseteq X \times X \) such that

\[
R \circ ; \to \subseteq ; R \land R' \circ ; \to \subseteq ; R'.
\]

With the above property of the converse we can give a general definition of a bisimulation:
Definition 2.4 An element \( b \in M \) is called a bisimulation for \( g \in M \) iff

\[
|b|g| \leq |g|b| \wedge (b|g| \leq |g|b|).
\]

As it turns out bisimulations in this sense are closed under multiplication and arbitrary sums. So, if a property \( A \) is preserved under arbitrary sums and at least one bisimulation fulfilling \( A \) exists the greatest bisimulation with the property \( A \) exists, namely the sum of all the bisimulations with this property.

As last concept we adapt equivalences and partitions to our framework:

Definition 2.5 An element \( x \in M \) is called an equivalence, if \( 1 \leq x \), \( x \cdot x \leq x \) and \( |x| = \langle x| \) hold. An equivalence class of \( x \) is an atomic fixed point of \( f(p) = |x|p \).

In this definition the condition \( 1 \leq x \) corresponds to reflexivity, \( x \cdot x \leq x \) models transitivity and the requirement \( |x| = \langle x| \) describes symmetry.

With this definition a lot of properties of equivalences known from the traditional relational point of view are also fulfilled: The equivalence classes are the images of atomic tests under an equivalence, the equivalence classes form a partition, i.e. they sum up to one and are pairwise disjoint, and many others. Altogether we have a reasonable tool for algebraic reasoning about bisimulations and equivalences.

3 Applications

3.1 Recent Results

A generic algorithm for model refinement looks as follows:

**Input** Graph \( G = (V, E) \), Desired Property \( C \).

**Step 1** Construct the simplified graph \( G' = (V', E') \), where \( V' \) is the set of equivalence classes of a suitable coarsest bisimulation.

**Step 2** Use any algorithm to construct a subgraph \( G'' = (V'', E'') \) of \( G' \), which fulfills the property \( C \).

**Step 3** Construct from \( G'' \) a subgraph \( \hat{G} \) of \( G \), which fulfills the property \( C \).

**Output** The Graph \( \hat{G} \), which satisfies \( C \).

The crucial question is for which kind of desired properties \( C \) this algorithm will generate a correct output. In [4] it was shown that the algorithm is correct if one wants to find the greatest live part of \( G \), i.e. the greatest subgraph of \( G \) without sinks, i.e. in our context nodes with an incoming but without an outgoing edge. This proof therefore was already done in the proposed algebraic setting. Sintzoff showed in [9] that this methods works for shortest path problems. However, there the setting was a little bit different: In this case a labelled graph was considered, so the concept of bisimulation had to be slightly redefined. The proof in this work was done in the traditional context. In contrast, the work in [4] lacks the algebraic treatment of labelled graphs. This can be done be using
a vector \((b_l)_{l \in L}\) of quantale elements for a set of labels \(L\), each of it representing the edges with a certain label.

The optimisation problem of [8] gives also a hint for future work. It concerns the algebraic treatment of optimisation problems via this algorithm. It works for shortest paths with strictly positive integer edge weights, whereas no results for positive real or even negative edge weights are known. One conjecture is that it is correct for all optimisation problems that can be described by the Bellmann’s equations (see e.g. [5], p.68 ff.).

3.2 Temporal Logic

In [2] it is shown that all statements of the temporal logic CTL* are tractable by means of the above algorithm (this is not shown litterally, but it is an immediate consequence of theorem 14 of this book). [7] gives a quantale framework for CTL*, so it is not unrealistic to expect a proof of this property in a quantale setting (the proof in [2] uses induction over the structure of CTL*-formulae and considers specific paths).

A slight modification of our basic algorithm can also be used as a tool for model checking (see e.g. [1] and [2]): In order to prove a temporal logical property of a model \(M\) it suffices under certain coditions to show that this property holds for a model \(M'\) which results from \(M\) by a construction analogous to step 1 in our algorithm.

References