Technical report outline
(intended to the VSR staff)

Tahina Ramananandro
Daniel Jackson

July 25, 2006

Also get some stuff from the slides presented in England.

1 The Alloy method

1.1 Introduction : Alloy language and Analyzer

Here, I don't think it would be necessary to describe something very precise about Alloy. Just recall that we write a specification, and analyze it through the Alloy Analyzer, with the model-finding method.

1.2 A naive method : following Z spec

A signature can represent either an object or a predicate. Objects are constrained.

1.3 A more rigorous method : using Alloy idiom

Make constraints local, instead of global.

This way, every instance has to be considered.

2 Technical issues

2.1 Integers : using Alloy ordering tools

The current implementation of the Alloy Analyzer does not support integers very well. So, I was advised to avoid them. The properties of integers used in the SEQNO are only the order, no operations.

2.2 CLEAR codes

Implemented as such in the naive version. But, as we did in the new model, we may say that a CLEAR code can be represented by the corresponding exceptionLogClear message itself.

2.3 Coin sharing

It would have been very difficult to compute the sum of sets of values with integers. Thanks to the join operator, use coins instead.

This involves developing particular constraints. First, try those making sense.

But the constraints used in the naive model are too strong.

2.4 Existential theorems

Quite the same as in the non-technical report.

3 Bugs found in the Z specification

The model-finding method finds bugs through giving counterexamples to the constraints and checked assertions.

3.1 Authenticity

As many other methods found, the PayDetails of purses in epv or epa have to be relevant to them.

3.2 Operation composition and framing schema

To decompose operations that first abort, you use the framing schema lemma. Even though the general lemma itself is true, it is useless here, and using it with BOp, COp is false because BOp, COp are not framing schemas because of m output messages.

3.3 Splitting constraint for Abort

The Abort splitting constraint is wrong : it gave me a counterexample where the aborting purse was the from in epa.

This shows the importance of also checking textual comments between proofs. But finding such bugs is quite difficult with model-finding approach because this method is intended to be as automatic as possible. Where is the limit between automation and proof-checking, then ? Theorem provers do not have this issue because they (are supposed to) provide their own proofs.

4 Results

4.1 Theorems tackled

In the naive model.

General schemas of the models.

4.2 Running times

Here insert the tables, and probaby even also the plotting graph.

4.3 Limits to the method

These results are valid only for finite scopes.

5 Extended analysis : Using theorem provers

5.1 Alloy and FOL

Alloy semantics in FOL. Problem : transitive closure.

5.2 Results with the Mondex case study

Currently working. Brought some proofs (hoping so, then we might compare them to).

5.3 Limits to the method

It actually depends on the ability to express the model in FO without TC. Mondex fits well if finiteness issues are dropped. Also thanks to coin sharing, ...

6 Conclusion and future work

Quite the same as in the ENS report, but probably add technical details (how to write a Z spec).