In this work we apply the program verification concept of symbolic execution to the verification of model transformations written in the Turing-incomplete DSLTrans language. Current state of the art model transformation verification techniques rely on SMT solvers or theorem proving to deal with the complexity of model transformations [1, 2, 3]. Our symbolic execution construction algorithms leverage on DSLTrans’ reduced expressiveness in order to cope with the classical state space explosion problem typical to symbolic execution. As a result our verification technique [4, 5] is simple, relies on an in-house model transformation symbolic execution engine, is fully axiomatized, can be mathematically proved and has the potential to scale to real size GM transformations.

DSLTrans Model Transformations and Properties

Symbolic Execution Construction

Property Proof

Experimental Results

Implementation

Conclusion and Contributions

Bibliography


http://msdl.cs.mcgill.ca/