# ProB: System Description for SMT-COMP 2019

Sebastian Krings and Michael Leuschel
University of Düsseldorf
Niederrhein University of Applied Sciences
sebastian.krings@uni-duesseldorf.de

## System Description

ProB [8, 9, 7] is a constraint solver, animator and model checker for software models written using the specification languages B [1] and Event-B [2]. In order to get access to the benchmarks and test cases collected in the SMT-LIB, we developed a translator from SMT-LIB to B. An SMT-LIB model is transformed into a (partial) B specification and fed into ProB's constraint solver. A solution found corresponds to a reachable state of the software model. The solution is then translated back to the corresponding SMT model.

Internally, ProB uses constraint logic programming, relying on the CLP(FD) library of SICStus Prolog [3]. Custom extensions to it are used to handle infinite domains and to add addtional inferrence rules [5]. Additionally, ProB itself can use different SMT solvers as additional backends [6]. For the 2019 SMT competition however, we only use our main backend.

How the constraint solving kernel of ProB can be used as a prover is outlined in [4], where we compared its perfomance to other SMT solvers using B and Event-B proof obligations as benchmarks. Now, the tables are turned and SMT problems will be used as benchmarks in SMT-COMP.

### Supported Logics

The translation from SMT-LIB to B supports all logics not including real arithmetic. In particular, due to the characteristics of CLP(FD), we expect the performance on linear and non-linear integer arithmetic to be quite well. In fact, in 2016, ProB won the main track of the NIA division of SMT-COMP.

However, performance on uninterpreted functions, arrays and bit vectors is rather weak when compared to state-of-the-art SMT solvers. This is mostly caused by ProB relying too much on enumeration rather than propagation in these cases. Hence, we suspect that using our solver on certain theories will not be worthwhile for now. Summarizing, for 2019, we are only participating in the following logics:

- linear integer arithmetic, both with and without quantifiers,
- non-linear integer arithmetic, both with and without quantifiers, and
- integer difference logic.

We plan to improve translation and support for other logics in the future.

## Availability and Usage

ProB is released under the Eclipse Public License v1.0 and is available from `www.prob2.de`. Source code is available there as well. In order to compile or run from source you need SICStus Prolog. The pre-build version can be used without a SICStus license.

You can start ProB on SMT-LIB files by passing them on the command line: `probcli benchmark.smt2`. A REPL can be started using a command line switch: `probcli --smtlib-cli`.

## Acknowledgments

Several other authors have made contributions to ProB: Michael Butler, Jens Bendisposto, Daniel Plagge, Lukas Ladenberger, Ivaylo Dobrikov, Joshua Schmidt, Jannik Dunkelau, Marc Fontaine, Fabian Fritz, Corina Spermann, Michael Jastram, Dominik Hansen, Philipp Körner, Joy Clark, Edward Turner, Dennis Winter, Sonja Holl, Jens Krüger, Michael Birkhoff, Carla Ferreira, Stéphane Lo Presti, Leonid Mikhailov, Laksono Adhianto, and many more.

## References

[1] J.-R. Abrial. *The B-book: Assigning Programs to Meanings*. Cambridge University Press, New York, NY, USA, 1996.

[2] J.-R. Abrial. *Modeling in Event-B: System and Software Engineering*. Cambridge University Press, New York, NY, USA, 1st edition, 2010.

[3] M. Carlsson, G. Ottosson, and B. Carlson. An Open-Ended Finite Domain Constraint Solver. In *Proceedings PLILP*, pages 191–206. Springer, 1997.

[4] S. Krings, J. Bendisposto, and M. Leuschel. From Failure to Proof: The ProB Disprover for B and Event-B. In *Proceedings SEFM'2015*, volume 9276 of *LNCS*. Springer, 2015.

[5] S. Krings and M. Leuschel. Constraint Logic Programming over Infinite Domains with an Application to Proof. In *Proceedings WLP*, volume 234 of *EPTCS*. Electronic Proceedings in Theoretical Computer Science, 2016.

[6] S. Krings and M. Leuschel. Smt solvers for validation of b and event-b models. In *Proceedings iFM 2016*, volume 9681 of *LNCS*, 2016.

[7] M. Leuschel, J. Bendisposto, I. Dobrikov, S. Krings, and D. Plagge. From Animation to Data Validation: The ProB Constraint Solver 10 Years On. In J.-L. Boulanger, editor, *Formal Methods Applied to Complex Systems: Implementation of the B Method*, chapter 14, pages 427–446. Wiley ISTE, Hoboken, NJ, 2014.

[8] M. Leuschel and M. Butler. ProB: A model checker for B. In *Proceedings FME*, volume 2805 of *LNCS*, pages 855–874. Springer, 2003.

[9] M. Leuschel and M. Butler. ProB: An automated analysis toolset for the B method. *Software Tools for Technology Transfer*, 10(2):185–203, 2008.