S-TaLiRo: A Tool for Temporal Logic Falsification for Hybrid Systems*

Yashwanth Annapureddy[1], Che Liu[1], Georgios Fainekos[1]
Sriram Sankaranarayanan[2]

[1] Arizona State University, Tempe, AZ.
{Yashwanthsingh.Annapureddy,Che.Liu,fainekos}@asu.edu.
[2] University of Colorado, Boulder, CO. srirams@colorado.edu

Abstract. S-TaLiRo is a Matlab (TM) toolbox that searches for trajectories of minimal robustness in Simulink/Stateflow diagrams. It can analyze arbitrary Simulink models or user defined functions that model the system. At the heart of the tool, we use randomized testing based on stochastic optimization techniques including Monte-Carlo methods and Ant-Colony Optimization. Among the advantages of the toolbox is the seamless integration inside the Matlab environment, which is widely used in the industry for model-based development of control software. We present the architecture of S-TaLiRo and its working on an application example.

1 Introduction

Temporal verification involves the ability to prove as well as falsify temporal logic properties of systems. In this paper, we present our tool S-TaLiRo for temporal logic falsification. S-TaLiRo searches for counterexamples to *Metric Temporal Logic* (MTL) properties for non-linear hybrid systems through global minimization of a *robustness metric* [4]. The global optimization is carried out using stochastic optimization techniques that perform a random walk over the initial states, controls and disturbances of the system. In particular, the application of *Monte-Carlo techniques* that use sampling biased by robustness is described in our HSCC 2010 paper [6]. In [1], we report on our experience with other optimization techniques including *Ant-Colony Optimization*.

At its core, S-TaLiRo integrates robustness computation for traces of hybrid systems (TaLiRo) [4, 6] with stochastic simulation [9]. The search returns the simulation trace with the smallest robustness value that was found. In practice, traces with negative robustness are falsifications of temporal logic properties. Alternatively, traces with positive - but low - robustness values are closer in distance to falsifying traces using a mathematically well defined notion of distance

* This work was partially supported by a grant from the NSF Industry/University Cooperative Research Center (I/UCRC) on Embedded Systems at Arizona State University and NSF awards CNS-1017074 and CNS-1016994.

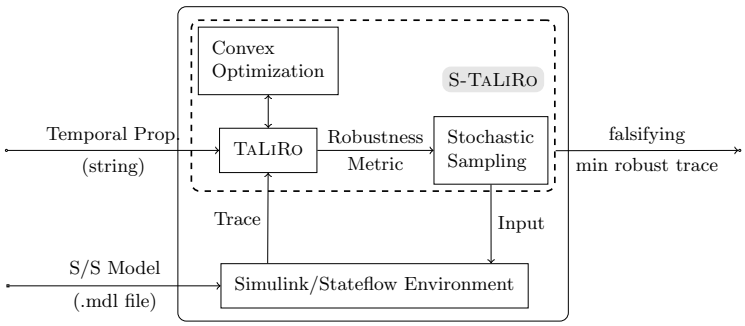


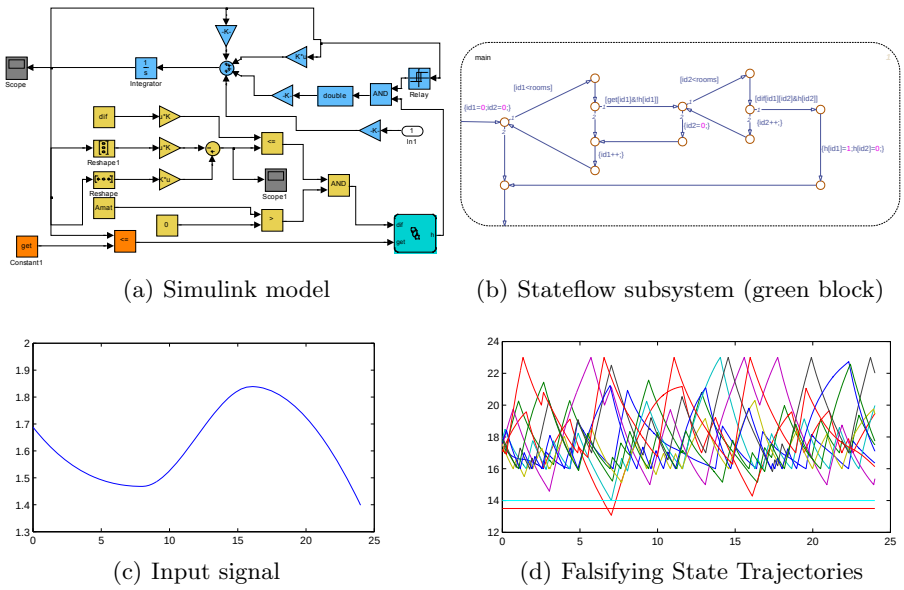
Fig. 1. The architecture of the S-TaLiRo tool.

between trajectories and temporal logic properties. Such traces may provide valuable insight to the developer on why a given property holds or how to refocus a failed search for a counter-example.

S-TaLiRo supports systems implemented as Simulink/Stateflow (TM) models as well as general *m-functions* in Matlab. Other frameworks can be readily supported, provided that a Matlab (TM) interface is made available to their simulators. S-TaLiRo has been designed to be used by developers with some basic awareness of temporal logic specifications. Simulink/Stateflow (TM) models are the *de-facto* standard amongst developers of control software in many domains such as automotive control and avionics. S-TaLiRo also supports the easy input of MTL formulae through an in-built parser. It has been designed and packaged as a Matlab toolbox with a simple command line interface. S-TaLiRo also contains an optimized implementation of the computation of the robustness metric (TaLiRo) over the previous version [3] along with the ability to plug-in other stochastic optimization algorithms.

2 The S-TaLiRo Tool

Figure 1 shows the overall architecture of our toolbox. The toolbox consists of a temporal logic robustness analysis engine (TaLiRo) that is coupled with a stochastic sampler. The sampler suggests input signals/parameters to the Simulink/Stateflow (TM) simulator which returns an execution trace after the simulation. The trace is then analyzed by the robustness analyzer which returns a robustness value. The robustness is computed based on the results of convex optimization problems used to compute signed distances. In turn, the robustness score computed is used by the stochastic sampler to decide on a next input to analyze. If a falsifying trace is found in this process, it is reported to the user. The trace itself can be examined inside the Simulink/Stateflow modeling environment. If, on the other hand, the process times out, then the least robust trace found by the tool is output for user examination.



(a) Simulink model (b) Stateflow subsystem (green block)

(c) Input signal (d) Falsifying State Trajectories

Fig. 2. Room heating benchmark HEAT30 and results obtained from S-TaLiRo run.

3 Usage

S-TaLiRo has been designed to be seamlessly integrated in the model based design process of Matlab/Simulink (TM). The user designs the model in the Simulink/Stateflow (TM) environment as before. At present, the only requirement is that input signals must be provided to the Simulink model through input ports. Then S-TaLiRo is executed with the name of the Simulink model as a parameter along with the set of initial conditions, the constraints on the input signals (if any) and the MTL specification. Currently, the user may select one of the two available stochastic optimization algorithms: Monte Carlo or Ant Colony Optimization. However, the architecture of S-TaLiRo is modular and, thus, any other stochastic optimization method can be readily implemented.

As a demonstration, we applied S-TaLiRo to the room heating benchmark from [5] (see Fig. 2). We chose the benchmark instance HEAT30. This is a hybrid system with 10 continuous variables (10 rooms) and 3360 discrete locations ($\binom{10}{4}2^4$ where 4 is the number of the heaters). The set of initial conditions is $[17, 18]^{10}$ and input signal u can range in $[1, 2]$. The goal is to verify that no room temperature drops below $[14.50\ 14.50\ 13.50\ 14.00\ 13.00\ 14.00\ 14.00\ 13.00\ 13.50\ 14.00]^T$. The input signal was parameterized using a piecewise cubic Hermite interpolating polynomial with 4 control points evenly distributed in the simulation time. S-TaLiRo found a falsifying trace with robustness value of -0.429. Figure 2 shows the trace and the input signal discovered by S-TaLiRo. In detail, the initial conditions were $x_0 = [17.4705\ 17.2197\ 17.0643\ 17.8663\ 17.4316\ 17.5354\ 17.9900\ 17.6599\ 17.8402\ 17.2036]^T$.

4 Related Work

The problem of testing hybrid systems has been investigated by many researchers (see the related research section in [6]). Most of the research focuses on *parameter estimation* [8, 2]. Recently, however, the problem of temporal logic falsification for hybrid systems has received some attention [7, 6]. Unfortunately, the publicly available tool support has been fairly low in this space. The only other publicly available toolbox that supports computation of robustness for temporal logic formulas with respect to real-valued signals is BREACH [2]. However, BREACH currently does not support temporal logic falsification for arbitrary Simulink/Stateflow models. Along the lines of commercial products, Mathworks provides a number of tools such as SystemTest[1] (TM) and Simulink Design Verifier[2] (TM). S-TaLiRo does not attempt to be a comprehensive test tool suite as the above, but rather to solve a very targeted problem, i.e., the problem of temporal logic falsification for hybrid systems. In the future, we hope to extend S-TaLiRo and the theory of robustness to estimate properties such as worst-case timings and integrate it into the statistical model checking framework.

References

1. Y. S. R. Annapureddy and G. Fainekos. Ant colonies for temporal logic falsification of hybrid systems. In *Proceedings of the 36th Annual Conference of IEEE Industrial Electronics*, 2010.
2. A. Donze. BREACH, a toolbox for verification and parameter synthesis of hybrid systems. In *Computer Aided Verification*, volume 6174 of *LNCS*, pages 167–170. Springer, 2010.
3. G. E. Fainekos and G. J. Pappas. A user guide for TaLiRo. Technical report, Dept. of CIS, Univ. of Pennsylvania, 2008.
4. G. E. Fainekos and G. J. Pappas. Robustness of temporal logic specifications for continuous-time signals. *Theoretical Computer Science*, 410(42):4262–4291, 2009.
5. A. Fehnker and F. Ivanĉić. Benchmarks for hybrid systems verification. In *Hybrid Systems: Computation and Control*, volume 2993 of *LNCS*, pages 326–341. Springer, 2004.
6. T. Nghiem, S. Sankaranarayanan, G. Fainekos, F. Ivančić, A. Gupta, and G. Pappas. Monte-Carlo techniques for the falsification of temporal properties of non-linear systems. In *Hybrid Systems: Computation and Control*, pages 211–220. ACM Press, 2010.
7. E. Plaku, Lydia E. Kavraki, and Moshe Y. Vardi. Falsification of LTL safety properties in hybrid systems. In *Tools and algorithms for the construction and analysis of systems*, volume 5505 of *LNCS*, pages 368 – 382. Springer, 2009.
8. A. Rizk, G. Batt, F. Fages, and S. Soliman. On a continuous degree of satisfaction of temporal logic formulae with applications to systems biology. In *6th International Conference on Computational Methods in Systems Biology*, volume 5307 of *LNCS*, pages 251–268. Springer, 2008.
9. R. Y. Rubinstein and D. P. Kroese. *Simulation and the Monte Carlo Method*. Wiley Series in Probability and Mathematical Statistics, 2008.

[1] `http://www.mathworks.com/products/systemtest/`

[2] `http://www.mathworks.com/products/sldesignverifier/`