EPiC Series in Computer Science

Volume 34, 2015, Pages 113–119

ARCH14-15. 1st and 2nd International Workshop on
Applied veRification for Continuous and Hybrid Systems

Using S-TaLiRo on Industrial Size Automotive Models

Bardh Hoxha, Houssam Abbas, and Georgios Fainekos

Arizona State University,
Tempe, AZ, USA
{bhoxha, hyabbas, fainekos}@asu.edu

Abstract

In Model Based Development (MBD) of embedded systems, it is often desirable to verify or falsify certain formal specifications. In some cases it is also desirable to find the range of specification parameters for which the specification does not hold on the system. We illustrate these methods on a challenge problem from the automotive industry on a high-fidelity, industrial scale engine model.

1 Introduction

Incidents such as [8] reinforce the need for design, verification, and validation methodologies for safety-critical systems. Due to the importance of the problem, we have investigated the testing of embedded and hybrid systems with respect to formal requirements in Metric Temporal Logic (MTL) [1]. MTL enables system engineers to express complex requirements. We use the robustness estimate, as presented in [7], to cast the falsification problem of MTL formulas as an optimization problem. The robustness of a trajectory with respect to an MTL specification is a quantitative evaluation, where negative values indicate that the trajectory does not satisfy the specification, and positive values indicate that the trajectory does satisfy the specification. The magnitude of the robustness value indicates how close the trajectory is to falsifying or satisfying the specification. The robust semantics can be computed with different algorithms and guarantees [5, 7].

We demonstrate our methods and framework with our Matlab toolbox S-TaLiRo [2] using a high-fidelity, industrial size engine model from the SimuQuest Enginuity Matlab/Simulink tool package.

2 Preliminaries

2.1 Falsification

Falsification is the process of finding a system trajectory, a counter example, for which the specification does not hold. S-TaLiRo searches for counterexamples to MTL properties for non-linear hybrid systems through global minimization of the robustness metric [7].

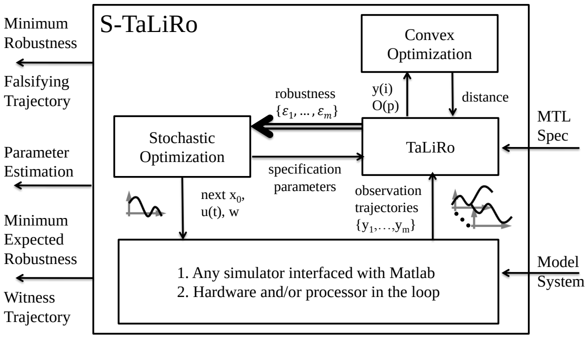

Figure 1: Architecture of S-TaLiRo.

S-TaLiRo integrates robustness computation for trajectories of hybrid systems with stochastic optimization. The search returns the simulation trajectory with the smallest robustness value that was found. Trajectories with positive - but low - robustness values are closer in distance to falsifying trajectories, using a mathematically well-defined notion of distance between trajectories and temporal logic properties. Such trajectories provide valuable insight to the developer on why a given property fails, or to our search algorithms on how to refocus a search for a counter-example.

2.2 Parameter Estimation

In Model Based Development (MBD) of embedded systems, it is often desirable to not only verify/falsify certain formal system specifications, but also to automatically explore the properties that the system satisfies. Namely, given a parametrized specification, we would like to automatically infer the ranges of parameters for which the property does not hold on the system. We consider parametric specifications in MTL. Using robust semantics for MTL, the parameter estimation problem can be converted into an optimization problem which can be solved by utilizing stochastic optimization methods. In [12], we demonstrate a method for solving this problem for specifications whose robustness function is monotonic with respect to the set of parameters. S-TaLiRo currently supports parameter estimation for parametric MTL formulas that contain one or more parameters. A different estimation approach is presented in [10].

3 Experimental Results

We initially present results on a simplified powertrain model which was first published by Ford [3]. The question posed is whether there are constant operating conditions that can cause a shift from gear two to gear one and then back to gear two. That implies that the transition was not necessary in the first place. In [5], we demonstrated that S-TaLiRo [2] can successfully solve the challenge problem on a simplified powertrain model. The specification in natural language is stated as follows: Does a transition exist from gear two to gear one and back to gear two in less than τ seconds? This requirement is formalized with the following MTL specification

$$\phi = \Box((gear_2 \wedge X gear_1) \rightarrow \Box_{(0,\tau]} \neg gear_2) \tag{1}$$

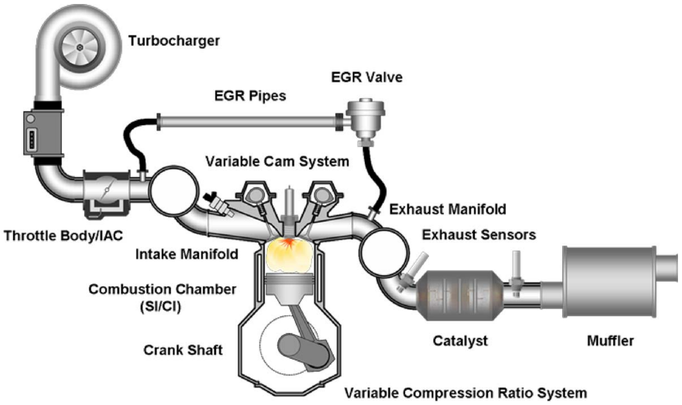

Figure 2: SimuQuest Enginuity model components. Used with permission, ©SimuQuest[11].

Note that through the simulation or sampling process, we are forced to use discrete-time semantics. Thus, the Next operator X is well defined. The relationship between discrete and continuous-time semantics is discussed in [6]. The interval $(0, \tau]$ implies that the current sample is ignored.

In [12], not only did we show that the specification could be falsified for $\tau = 2.5$ sec, but we also showed the the specification can be falsified with a τ parameter which is as low as $\tau = 0.4273$ sec. Due to the monotonicity of the robustness function with respect to the parameter, we demonstrated that the system is falsified for every $\tau \geq 0.4273$ sec using about 300 tests of the system.

In the following, we present our work on a high fidelity engine model from the SimuQuest Enginuity [11] Matlab/Simulink tool package. The goal is to illustrate the MTL falsification and parameter estimation methods on an industrial size and complexity model.

The Enginuity tool package includes a library of modules for engine component blocks. It also includes pre-assembled models for standard engine configurations. In this work, we will use the Port Fuel Injected (PFI) spark ignition, 4 cylinder inline engine configuration. It models the effects of combustion from first physics principles on a cylinder-by-cylinder basis, while also including regression models for particularly complex physical phenomena. Simulink reports that this is a 56 state model. Note that this number represents only the visible states. It is possible that more states are present in the blackbox s-functions which are not accessible. This is high dimensional non-linear system for which reachability analysis is very difficult. It also includes lookup tables, non-linear components, and inputs that affect the switching guards. The model includes a tire-model, brake system model, and a drive train model (including final drive, torque converter and transmission). The model is based on a zero-dimensional modeling approach so that the model components can all be expressed in terms of ODE's.

We test requirement (1) with $\tau = 3$ on the SimuQuest Enginuity engine model. The inputs to the system are the throttle and break schedules, and the road grade, which represents the incline of the road. The throttle and break at each point in time can take any value between 0 to 100. The road grade at each point in time can take any value between -33.5 and 33.5. The gradeability of the road, the highest grade a vehicle can ascend while maintaining a particular speed, is estimated to be 33.5.

We search for a particular input for the throttle schedule, break schedule, and grade level. The inputs are parametrized using 34 search variables, where 14 are used for the throttle schedule, 14 for the break schedule, and 6 for the grade level. The search variables for each

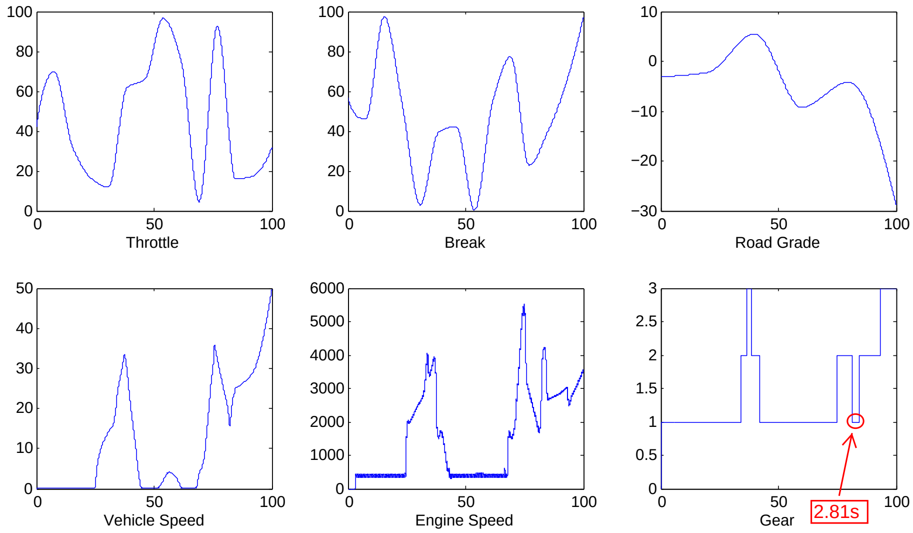

Figure 3: Falsifying trajectory for specification in Eq 1 with $\tau = 3$ on the SimuQuest Enginuity engine model. The specification is falsified since there is a case where at a specific point in time the model is not in gear one, and next transition to gear one, and stays in gear one for less than 3 seconds, specifically 2.81 sec.

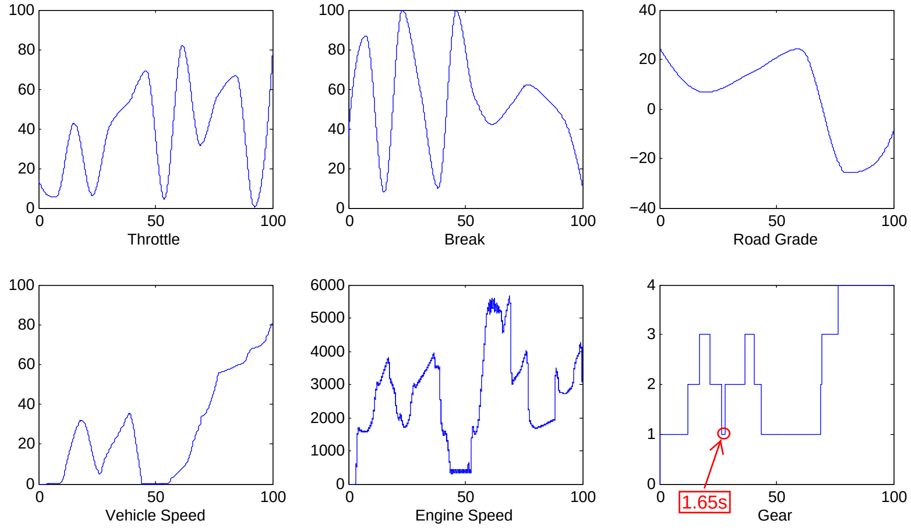

Figure 4: Falsifying trajectory for specification in Eq 1 with $\tau = 1.68$ on the SimuQuest Enginuity engine model. The specification is falsified since there is a case where at a specific point in time the model is not in gear one, and next transition to gear one, and stays in gear one for less than 1.68 seconds, specifically 1.65 sec.

input are interpolated with the Piecewise Cubic Hermite Interpolating Polynomial (PCHIP) function. The simulation time for the system is 100 sec.

The challenge encountered while running the experiments was in choosing the appropriate robustness metric. The specification is defined on gear transition sequences, i.e. the discrete dynamics. Therefore, a continuous state robustness metric alone would not be appropriate. Indeed we found that Simulated Annealing (SA) minimization of the state robustness did no better than Uniform Random search. On the other hand, a purely discrete metric, i.e., on the mode graph, would not be appropriate either, since the throttle controls the continuous state (vehicle and engine speeds) which in turn controls the gear sequence. Namely, the gear sequence is a *delayed* indicator of what's happening in the system, so the discrete robustness might not provide enough feedback to the optimizer. Moreover, intuitively, we are trying to minimize the switching delay between gears 2 and 1. So we tried the hybrid robustness [1] $\langle k, \theta \rangle$, which contains two components: the discrete component $k \in \mathbb{Z}$ is an integer that gives the distance between the discrete trajectory (i.e. sequence of locations) and the target falsifying location. The continuous component θ is the temporal robustness [4]; this measures by how much to shift the output signal to change its truth value w.r.t. the specification. This however didn't yield a falsifier either. One explanation is that temporal robustness does not measure timing distortions (which is what we need here): it only measures by how much to move the signal *rigidly* to falsify/satisfy the spec.

At this point, we used hybrid robustness $\langle k, r \rangle$, where k is as before, and r is the state robustness, which measures how far we are from satisfying the conditions that cause a jump to the next location on the shortest path to the target location (or the distance to the unsafe set if already at target location). Due to the complexity of the model, and the fact that parts of the model are black box functions, we can only determine the transition guards from the controller, which closely match the plant transitions but not exactly. Thus, the problem becomes more challenging.

We decided to take two different approaches to the problem:

1. We approximate the plant gear transition guards in order to compute the state robustness r component of the hybrid distance metric from the controller. We run our falsification algorithm. After 51 tests and 1752 sec, we find a counterexample (see Fig 3) that shows that the system does not satisfy the specification. We have falsified $\phi = \Box_{[0,100]}((\neg g_1 \wedge X g_1) \rightarrow \Box_{(0,\tau]} \neg g_2)$ for $\tau = 3$. The natural question that follows is: What is the minimum value of τ for which the system is not satisfied. Essentially, the falsification problem now turns into the parameter estimation problem described in Section 2.2. The smallest value τ found for which the specification is falsified is $\tau = 1.68s$, as shown in Fig. 4.

2. Another approach to this problem is make small modifications to the model and specification, thereby avoiding the issue of knowing the exact transitions for the gear change. We introduced two changes. First, guided by the intuition that we also want to minimize the switching time, we added the dwell time in a given gear as a state: $\dot{\tau} = 1, \tau^+ = 0$. Second, we changed the specification to $\phi_2 = \Box_{[0,100]}((g_2 \wedge X g_1) \rightarrow \Box_{(0,\lambda]}((\tau \leq \lambda) \rightarrow g_1)$. The hybrid distance now is composed of the hybrid robustness $\langle k, r \rangle$, where k is as before, and r is the state robustness that returns the distance of the current dwell time to a specific value, in this case λ. We run the parameter estimation algorithm and we find that the smallest value λ found for which the specification is falsified is $\lambda = 1.29s$, as shown in Fig. 5.

Acknowledgments: This work was partially funded under NSF awards CNS 1116136 and CNS 1319560. We would like to thank the Model Based Development group at Toyota Technical Center for providing a license to the Simuquest model. We would also like to thank Raymond Turin, Founder and CTO at SimuQuest, who provided assistance in working with the SimuQuest Enginuity model and Adel Dokhanchi for his help with the robustness computations.

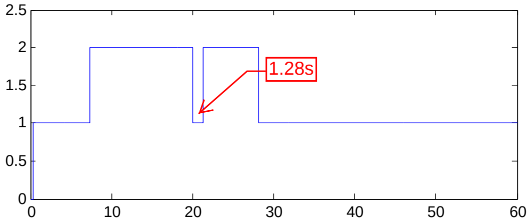

Figure 5: Falsifying trajectory for specification $\phi_2 = \Box_{[0,100]}((g_2 \wedge X g_1) \rightarrow \Box_{(0,\lambda]}((\tau \leq 1.29) \rightarrow g_1)$ with $\lambda = 1.29$ on the SimuQuest Enginuity engine model.

A Appendix

The scripts for running the falsification and parameter estimation methods are available through our Matlab Toolbox S-TaLiRo [2, 9], available at https://sites.google.com/a/asu.edu/s-taliro/s-taliro under the benchmarks/ARCH2014 subfolder. Running the scripts requires the SimuQuest Enginuity Matlab/Simulink tool package.

References

[1] Houssam Abbas, Georgios E. Fainekos, Sriram Sankaranarayanan, Franjo Ivancic, and Aarti Gupta. Probabilistic temporal logic falsification of cyber-physical systems. *ACM Transactions on Embedded Computing Systems*, 12(s2), May 2013.

[2] Yashwanth Singh Rahul Annapureddy, Che Liu, Georgios E. Fainekos, and Sriram Sankaranarayanan. S-taliro: A tool for temporal logic falsification for hybrid systems. In *Tools and algorithms for the construction and analysis of systems*, volume 6605 of *LNCS*, pages 254–257. Springer, 2011.

[3] Alongkrit Chutinan and Kenneth R. Butts. Dynamic analysis of hybrid system models for design validation. Technical report, Ford Motor Company, 2002.

[4] Alexander Donze and Oded Maler. Robust satisfaction of temporal logic over real-valued signals. In *Formal Modelling and Analysis of Timed Systems*, volume 6246 of *LNCS*. Springer, 2010.

[5] Georgios Fainekos, Sriram Sankaranarayanan, Koichi Ueda, and Hakan Yazarel. Verification of automotive control applications using s-taliro. In *Proceedings of the American Control Conference*, 2012.

[6] Georgios E. Fainekos and George J. Pappas. Robust sampling for MITL specifications. In Jean-Francois Raskin and P. S. Thiagarajan, editors, *Proceedings of the 5th International Conference on Formal Modelling and Analysis of Timed Systems*, volume 4763 of *LNCS*, pages 147–162. Springer, 2007.

[7] Georgios E. Fainekos and George J. Pappas. Robustness of temporal logic specifications for continuous-time signals. *Theoretical Computer Science*, 410(42):4262–4291, 2009.

[8] E. J. Hoffman, W. L. Ebert, M. D. Femiano, H. R. Freeman, C. J. Gay, C. P. Jones, P. J. Luers, and J. G. Palmer. The near rendezvous burn anomaly of december 1998. Technical report, Applied Physics Laboratory, Johns Hopkins University, November 1999.

[9] Bardh Hoxha, Nikolaos Mavridis, and Georgios Fainekos. VISPEC: A graphical tool for elicitation of MTL requirements. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2015.

[10] Xiaoqing Jin, Alexandre Donzé, Jyotirmoy V Deshmukh, and Sanjit A Seshia. Mining requirements from closed-loop control models. In *Proceedings of the 16th international conference on Hybrid systems: computation and control*, pages 43–52. ACM, 2013.

[11] Simuquest. Enginuity. `http://www.simuquest.com/products/enginuity`. Accessed: 2013-10-14.

[12] Hengyi Yang, Bardh Hoxha, and Georgios Fainekos. Querying parametric temporal logic properties on embedded systems. In *Int. Conference on Testing Software and Systems*, volume 7641, pages 136–151. Springer, 2012.