Pareto Front Exploration for Parametric Temporal Logic Specifications of Cyber-Physical Systems

Bardh Hoxha and Georgios Fainekos

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

The abstract will be presented orally.

One of the advantages of adopting a Model Based Development (MBD) process is that it enables testing and verification at early stages of development. However, it is often desirable to not only verify/falsify certain formal system specifications, but also to automatically explore the properties that the system satisfies. In this work, we present algorithms that enable the temporal logic property exploration for Cyber-Physical Systems. Namely, given a parametric specification in Parametric Metric Temporal Logic (PMTL) with multiple state and/or timing parameters, our solution can automatically infer the set of parameters for which the property does not hold on the system.

Such an exploration framework would be of great value to the practitioner. The benefits are twofold. One, it allows for the analysis and development of specifications. In many cases, system requirements are not well formalized by the initial system design stages. Two, it allows for the analysis and exploration of system behavior. If a specification can be falsified, then it is natural to inquire for the range of parameter values that cause falsification. That is, in many cases, the system design may not be modified, but the guarantees provided should be updated.

In the case of single parameter mining, the solution of the problem is a one dimensional interval. On the other hand, in the case of specifications with multiple parameters, finding a solution to the problem becomes more challenging since the optimization problem is converted to a multi-objective optimization problem where the goal is to determine the Pareto front. To solve this problem, we present a method for effective one-sided exploration of the Pareto front and provide a visualization method for the analysis of parameters. The algorithms presented in this work are incorporated in the toolbox S-TaLiRo [4]. For an overview of the toolbox see [2].

As a running example, we utilize the Automatic Transmission model described in [1]. There are two inputs to the system: the throttle and break. The physical system has two continuous-time state variables which are also its outputs: the speed of the engine ω (RPM) and the speed of the vehicle v (mph). In this work, we provide answers to queries like "Always the vehicle speed v and engine speed ω need to be less than parameters θ_1, θ_2, respectively". In PMTL, we can define such specification as follows: $\phi[\vec{\theta}] = \Box((vehicle\ speed < \theta_1) \wedge (rpm < \theta_2))$. Formally, the parameter mining problem is defined as follows:

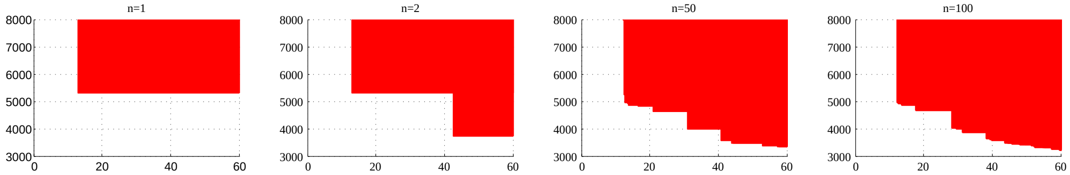

Figure 1: Illustration of the iterative process for the RGDA algorithm. Specification: $\phi[\vec{\lambda}] = \neg(\Diamond_{[0,\lambda_1]} q \wedge \Box p[\lambda_2])$ where $p[\lambda_2] \equiv (\omega \leq \lambda_2)$ and $q \equiv (v \geq 100)$. The red colored set represents the set $\Psi = \{\theta \in \Theta \mid \Sigma \not\models \phi[\theta]\}$, i.e., the set of parameter values such that the system does not satisfy the specification. In each iteration of the algorithm, the set Ψ expands by the optimal falsifying parameter which is guided by the robustness landscape and the random weight in the priority function.

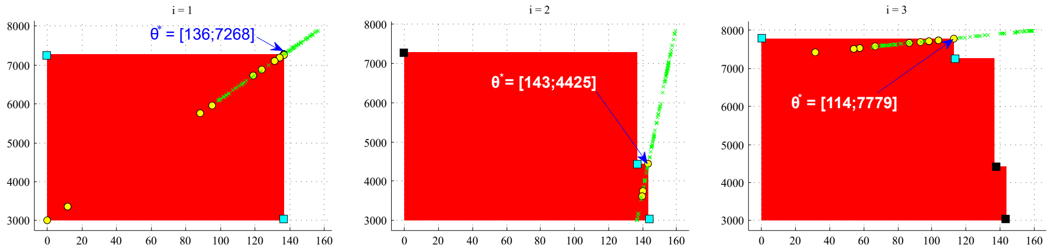

Figure 2: Illustration of the iterative process for the SDA algorithm. Specification: $\phi[\vec{\lambda}] = \Box(p[\lambda_1] \wedge q[\lambda_2])$ where $p[\lambda_1] \equiv (v \leq \lambda_1)$ and $q \equiv (\omega \leq \lambda_2)$. The parameter range for the specification is $\Theta = [0\ 160; 3000\ 8000]^T$. In each plot, the search is conducted in a specific direction $\vec{b}$. The red colored set represents set $\Psi = \{\theta \in \Theta \mid \Sigma \not\models \phi[\theta]\}$, i.e., the set of parameter values such that the system does not satisfy the specification.

Problem 1 (MTL m-Parameter Mining)**.** *Given a parametric MTL formula $\phi[\vec{\theta}]$ with a vector of m unknown parameters $\vec{\theta} \in \Theta = [\underline{\theta}, \overline{\theta}]$ and a system Σ, find the set $\Psi = \{\vec{\theta}^* \in \Theta \mid \Sigma \not\models \phi[\vec{\theta}^*]\}$.*

In the following, we present an overview of the solution. We utilize the theory of system robustness of MTL specifications in conjunction with monotonicity results to turn the falsification problem into an optimization problem. The notion of the robustness metric enables system developers to measure by how far a system behavior is from failing to satisfy a requirement. Given a model and a PMTL specification, the sampler produces a point x_0 from the set of initial conditions, input signal u and vector of mined parameters $\vec{\theta}$ for the PMTL specification. The initial conditions and input signal are passed to the system simulator which returns an execution trace (output trajectory and timing function). The trace, in conjunction with the mined parameters, is then analyzed by the MTL robustness analyzer which returns a robustness value. The robustness score computed is used by the stochastic sampler to decide on next initial conditions, inputs, and estimated parameters to utilize. This approach can be systematically utilized to estimate the solution to Problem 1. Namely, the set of falsifying parameters Ψ. In the following, we provide an informal description of two algorithms for solving Problem. 1.

Robustness Guided Parameter Falsification Domain Algorithm (RGDA): The algorithm explores the parameter falsification domain by slightly modifying the optimization function. The modifications, through assigning weights at parameter values, perturbes the landscape of the robustness estimate over the parameter values. By doing so, in each run of the algorithm, the solution returned is a different point in the Pareto front. We illustrate the iterative process of the algorithm in Fig. 1.

Structured Parameter Falsification Domain Algorithm (SDA): The algorithm explores the parameter falsification domain through a structured search in specific parameter directions. Each solution to the optimization problem in one direction, returns a marker which is utilized to initialize the search in future iteration. We illustrate the iterative process of the algorithm in Fig. 2.

The work in this abstract is presented in detail in [3].

References

[1] Bardh Hoxha, Houssam Abbas, and Georgios Fainekos. Benchmarks for temporal logic requirements for automotive systems. *Proc. of Applied Verification for Continuous and Hybrid Systems*, 2014.

[2] Bardh Hoxha, Hoang Bach, Houssam Abbas, Adel Dokhanchi, Yoshihiro Kobayashi, and Georgios Fainekos. Towards formal specification visualization for testing and monitoring of cyber-physical systems. In *International Workshop on Design and Implementation of Formal Tools and Systems*, 2014.

[3] Bardh Hoxha, Adel Dokhanchi, and Georgios Fainekos. Querying parametric temporal logic properties in model based design. *arXiv preprint arXiv:1512.07956*, 2015.

[4] S-TaLiRo: Temporal Logic Falsification Of Cyber-Physical Systems. `https://sites.google.com/a/asu.edu/s-taliro/s-taliro`, 2013. [Online; accessed April-2014].