Parametric Verification and Synthesis based on Gaussian Processes

Luca Bortolussi[1], Laura Nenzi[1], and Simone Silvetti[2]

[1] DMG, University of Trieste, Trieste, Italy
[2] Esteco S.p.A., Italy

Abstract. Parameter verification and synthesis problems for stochastic systems are still hard challenges. The tasks become even more prohibitive when the systems are large-scale and the verification/synthesis is respect the satisfaction of some linear temporal logic formulae. In this paper, we review a number of techniques that we designed to tackle such problems. The methods exploit the Gaussian Processes (GP) regression and an efficient Bayesian optimization algorithm.

1 Introduction and Problem Formulation

The world around us is complex, and when we try to model some aspects of it mathematically, we need to deal with underspecification and uncertainty in our descriptions. Stochastic processes offer a powerful framework to account consistently for such an uncertainty. However, stochastic models are still plagued by underspecification, in particular for what concerns the value of some defined parameters. While some of these parameters represent intrinsic and unknown properties of the system at hand, and may be learned from observations, others may be controllable or tunable in a design phase. This is the case in complex engineered systems interacting with uncertain environments, like Cyber-Physical Systems. Hence, there is the need of efficient model-based design approaches capable of dealing with uncertain parameters.

Formal methods are classical techniques used for specifications of complex behaviours, which can be applied also to stochastic systems, but they suffer from scalability issues. A possibility is to use simulation combined with statistical methods [1]. However, if our model presents also uncertain parameters we have additional computational cost and even these methods can be unfeasible. Here we review a number of techniques [2, 4, 5, 7] that we designed to do parameter verification and synthesis of such models that exploit Machine Learning and Bayesian Optimization.

Let us consider a stochastic model $\mathcal{M}_\theta$, where $\theta = (\theta_1, \dots, \theta_k)$ is a parameter vector, taking values in a space $\Theta \subset \mathbb{R}^k$. Usually we consider stochastic models defined as *Population Continuous Time Markov Chain* (PCTMC), but similar results can be obtained for different classes of stochastic systems that depend continuously on model parameters.

Running example. SIR epidemic models [10] are used to describe the spread of a disease in a population, and often represented as a stochastic process $\mathbf{X}(t) = (X_S(t), X_I(t), X_R(t))$, where each $X_i(t) \in \mathbb{N}$ counts the number of Susceptible (S), Infected (I) and Recovered (R) individuals at time t. We call $x(t) \in \mathcal{D}$ a single simulation, with $\mathcal{D}$

the trajectory space; k_i (infected rate) and k_r (recovery rate) are the uncertain parameters. Examples of behaviours that can be interesting to identify are: "the number of infected individuals peaks above 30 between 35 and 45 time units" and "the epidemic will terminate between 100 and 200 time units".

Signal Temporal Logic. Behavioural properties can be expressed using formal languages as *Temporal Logics*. Here we consider *Signal Temporal Logic* (STL) [11], which is a linear time temporal logic, with dense-time interval and real-valued trajectories and it comes with very efficient monitoring algorithms to check if a property is satisfied in a trace. The logic features a number of temporal operators that predicate properties about future or past configurations. The two behaviours previously described formally read as. $\varphi_{peak} = \mathcal{G}_{[35,45]}(X_I > 30)$ (using the globally or always operator $\mathcal{G}$) and $\varphi_{extinc} = (X_I \geq 1)\mathcal{U}_{[100,200]}(X_I \leq 0)$ (using the until operator $\mathcal{U}$). The Boolean semantics of the logic checks if a single trajectory (simulation) of the system satisfies or not a formula ($(x, 0) \models \varphi$). Combining it with the probability distribution P_θ over the trajectory space $\mathcal{D}$ induced by the stochastic model M_θ, we can define the satisfaction probability as $P_\varphi(\theta) = P_\theta\{x(t) \in \mathcal{D} \mid (x, 0) \models \varphi\}$. A similar definition can be done for the quantitative semantics of STL [8, 9]. This semantics assigns a real-valued $\rho(\varphi, x, 0)$ to each formula and trajectory x. The sign is positive (resp. negative) if the formula is true (resp. false), while the absolute value measures some robustness of the truth value. The key indicator here is the expectation $R_\varphi(\theta) = \mathbb{E}_\theta[\rho(\varphi, x, 0)]$ that can be seen as the average robustness of satisfaction in our model.

Problem Formulation. The type of problem that we want to solve is: let $f_\varphi(\theta)$ be the probability or expectation function. Given a model $\mathcal{M}_\theta$, and an STL property φ, we want to compute $f_\varphi(\theta)$ as a function of the model parameters $\theta \in \Theta$.

2 Methodology and Discussion

The exact evaluation of $f_\varphi(\theta)$, with $\theta \in \Theta$ a fixed parameter configuration, is extremely computationally costly. *Statistical Model Cheking* (SMC) is a viable approximate alternative: the model is simulated for a finite number of runs and the STL property is checked on each single trace, properties are evaluated and standard statistical techniques are then used to estimate $f_\varphi(\theta)$. As a further step, we propose Smoothed Model Cheking (SmMC), estimating $f_\varphi(\theta)$ as a function of model parameters leveraging *Gaussian Processes* (GP) regression.

Gaussian Processes. GPs are a Bayesian non-parametric approach used to solve supervised learning task. Roughly speaking, a GP is a distribution over a function space with the property that any finite dimensional distribution is Gaussian. In a regression task, a GP prior is combined with observations to provide a GP posterior. When integrals used to define the posterior are not analytically solvable, variational methods are typically used. We refer the reader to [12] for further details.

Smoothed Model Checking leverages GP to approximate the satisfaction probability (or average robustness) from a finite number of sampled trajectories at different parameter points evaluated via SMC, resulting in a Binomial likelihood model for observations.

Formally, a GP defines a latent function $g(\theta)$, which is then mapped into a probability value in $[0, 1]$ by the Probit function (the quantile function associated with the standard normal distribution), and inference of the posterior is carried out relying on an efficient variational approximation known as Expectation Propagation [4].

System Design via Robustness maximization In this setting, we want to find the parameter configuration θ^* for which the model $\mathcal{M}_{\theta^*}$ maximizes the average robustness $R_\varphi(\theta)$. Here GP reduces to a regression task using the noisy observations of $R_\varphi(\theta)$. The method exploits an active learning and Bayesian optimization algorithm, called GP-Upper Confidence Bound [13]. Instead of optimizing directly the emulated average robustness function, we optimize an upper confidence bound of the GP. This permits to have an exploration-exploitation trade-off, depending whether θ^* is coming from a region of high mean or of high variance. More details on this methodology can be found in [2]. In Fig. 1(left) we see a simulation of the SIR model with best θ^* that maximize $R_{\varphi_{Peak}}$.

Parameter synthesis. SmMC has been applied [7] to explore and identify the regions of the parameter space where the satisfaction probability of a target STL property φ is above or below a given threshold, i.e, $\{\theta \in \Theta \mid P_\varphi(\theta) > \alpha\}$. The key idea is to leverage the GP surrogate model learned by SmMC, in order to compute the GP probability p that the satisfaction probability of φ is above or below a specific threshold α, i.e., $p(P_\varphi(\theta) > \alpha)$. This confidence probability is then used to drive an active learning strategy which adds samples of parameters θ from regions of highest uncertainty on whether the probability of satisfaction of φ is above or below the predefined threshold. In Fig. 1(right) we show the parameter synthesis algorithm at work for the SIR model. Notice how the density of sampled points (black dots) is larger in the areas where the satisfaction probability is close to the threshold.

Discussion. Combining simulation with Bayesian machine learning enables and efficient statistical approach for parametric verification. The method can be extended to other linear time specifications [6], and is implemented in the open tool U-check [3].

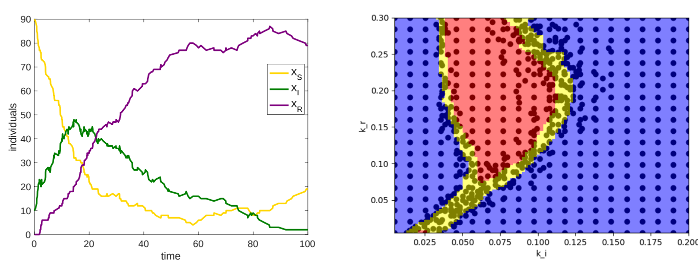

Fig. 1: Simulation of the SIR model with best θ^* optimizing $R_{\varphi_{peak}}$ (left) Parameter synthesis of infection k_i and recovery rate k_r, with probability below (resp. above) the threshold in the blue (resp. red) regions and in yellow the uncertain ones (right).

References

1. Agha, G., Palmskog, K.: A survey of statistical model checking. ACM Transactions on Modeling and Computer Simulation (TOMACS) 6, 28.1 (2018)
2. Bartocci, E., Bortolussi, L., Nenzi, L., Sanguinetti, G.: System design of stochastic models using robustness of temporal properties. Theoretical Computer Science 587, 3–25 (2015)
3. Bortolussi, L., Milios, D., Sanguinetti, G.: U-check: Model checking and parameter synthesis under uncertainty. In: Campos, J., Haverkort, B.R. (eds.) Quantitative Evaluation of Systems. pp. 89–104. Springer International Publishing, Cham (2015)
4. Bortolussi, L., Milios, D., Sanguinetti, G.: Smoothed model checking for uncertain continuous-time markov chains. Information and Computation 247, 235–253 (2016)
5. Bortolussi, L., Policriti, A., Silvetti, S.: Logic-based multi-objective design of chemical reaction networks. In: Cinquemani, E., Donzé, A. (eds.) Hybrid Systems Biology. pp. 164–178. Springer International Publishing, Cham (2016)
6. Bortolussi, L., Sanguinetti, G.: Learning and designing stochastic processes from logical constraints. Logical Methods in Computer Science 11(2) (2015), `https://doi.org/10.2168/LMCS-11(2:3)2015`
7. Bortolussi, L., Silvetti, S.: Bayesian statistical parameter synthesis for linear temporal properties ofstochastic models. In: Beyer, D., Huisman, M. (eds.) Tools and Algorithms for the Construction and Analysis of Systems. pp. 396–413. Springer International Publishing, Cham (2018)
8. Donzé, A., Maler, O.: Robust satisfaction of temporal logic over real-valued signals. In: Chatterjee, K., Henzinger, T.A. (eds.) Formal Modeling and Analysis of Timed Systems. pp. 92–106. Springer Berlin Heidelberg, Berlin, Heidelberg (2010)
9. Fainekos, G.E., Pappas, G.J.: Robustness of temporal logic specifications for continuous-time signals. Theoretical Computer Science 410, 4262–4291 (2009)
10. Gardner, T.S., Cantor, C.R., Collins, J.J.: Construction of a genetic toggle switch in escherichia coli. Nature 403(6767), 339–342 (2000)
11. Maler, O., Nickovic, D.: Monitoring temporal properties of continuous signals. In: Lakhnech, Y., Yovine, S. (eds.) Formal Techniques, Modelling and Analysis of Timed and Fault-Tolerant Systems. pp. 152–166. Springer Berlin Heidelberg, Berlin, Heidelberg (2004)
12. Rasmussen, C.E., Williams, C.K.I.: Gaussian processes for machine learning. Adaptive computation and machine learning, MIT Press (2006), `http://www.worldcat.org/oclc/61285753`
13. Srinivas, N., Krause, A., Kakade, S.M., Seeger, M.W.: Information-Theoretic Regret Bounds for Gaussian Process Optimization in the Bandit Setting. IEEE Transactions on Information Theory 58(5), 3250–3265 (2012)