Simulation-Guided Parameter Synthesis for Chance-Constrained Optimization of Control Systems

Yan Zhang*, Sriram Sankaranarayanan† and Benjamin M. Gyori‡
*School of Computing, National University of Singapore
†Department of Computer Science, University of Colorado at Boulder
‡Department of Systems Biology, Harvard Medical School
Email: *zhangyan@comp.nus.edu.sg, †srirams@colorado.edu, ‡benjamin_gyori@hms.harvard.edu

***Abstract*—We consider the problem of parameter synthesis for black-box systems whose operations are jointly influenced by a set of "tunable parameters" under the control of designers, and a set of uncontrollable stochastic parameters. The goal is to find values of the tunable parameters that ensure the satisfaction of given performance requirements with a high probability. Such problems are common in robust system design, including feedback controllers, biomedical devices, and many others. These can be naturally cast as chance-constrained optimization problems, which however, are hard to solve precisely. We present a simulation-based approach that provides a piecewise approximation of a certain quantile function for the responses of interest. Using the piecewise approximations as objective functions, a collection of local optima are estimated, from which a global search based on simulated annealing is performed. The search yields tunable parameter values at which the performance requirements are satisfied with a high probability, despite variations in the stochastic parameters. Our approach is applied to three benchmarks: an insulin infusion pump model for type-1 diabetic patients, a robust flight control problem for fixed-wing aircrafts, and an ODE-based apoptosis model from system biology.**

I. Introduction

Model-based design has become a common practice for developing a variety of systems ranging from electronic circuits to embedded control systems. It also helps scientists to understand the behavior of complex biochemical systems that govern fundamental biological processes such as programmed cell death. Working with large nonlinear mathematical models requires sophisticated tools to synthesize and verify system behaviors. This is often hard in the presence of numerous model parameters, whose changes can drastically affect the functionality of models and thus the overall behavior of a system. Usually these parameters arise in two categories, "tunable parameters" that can be set by the designers, and stochastic parameters that represent modeling error, inherent uncertainties and environmental disturbance.

In this paper, we examine the problem of adjusting tunable parameters in the presence of uncontrollable stochastic parameters specified by a probability distribution. Examples of tunable parameters include gains and switching thresholds for controller design, and kinetic rate constants for biochemical reaction models. Our goal is to find values of tunable parameters that satisfy given performance requirements. Rather than optimizing the worst-case performance, we use a *chance-constrained optimization* formulation that optimizes a given quantile of the relevant response. Stated otherwise, we wish to find a threshold such that the probability that the performance level stays below the threshold is at least some value θ under stochastic variations. For example, the specification of a PID controller requires that with a probability of at least 95%, the settling time of its step response is less than 10 ms.

We tackle chance-constrained problems for which models and performance functions are not available in closed forms, or too complex to handle even if the closed forms are available. Therefore, we adopt a simulation-based approach. Conceptually, we cast the chance-constrained problem into a related optimization problem over the θ-quantile of a performance function. Under this formulation, we use a quantile regression procedure to construct a piecewise approximation of the θ-quantile function [1]. Such a model defines a set of locally optimal tunable parameter values. Once the model is constructed, we perform a global optimization over the set of locally optimal points, which results in a globally optimal solution. The solution is checked against the performance function to determine whether the performance requirements are satisfied. If not, the procedure refines the approximation and iterates.

Our approach is evaluated over three applications. The first one is a parameter tuning problem for a nonlinear insulin-glucose system [2], which has 3 tunable and 15 stochastic parameters. The second is a robust flight control system, which has 27 tunable and 73 stochastic parameters. We also study a biochemical model which describes the dynamics of externally triggered programmed cell death [3]. It involves 71 tunable parameters for kinetic rate constants and 17 stochastic parameters for uncertain initial concentrations. Our approach tunes the rate constants so that the behavior of the model matches experimental observations. In each application, we show how our approach can find tunable parameter values that result in satisfactory performance with high probabilities.

A. Related Work

Chance-constrained optimization has been studied and algorithms have been proposed for restricted situations [4]–[6]. It provides an alternative solution to traditional robust control design problems. Tempo *et al.* summarizes the research of this area in their book [7]. We refer the reader to this book and references therein for further details. The main distinction between this line of work and our work lies in

This work was partially supported by Singapore Ministry of Education Research (Grant No. T2-2-033) and U.S. National Science Foundation (Award No. 1320069). All opinions expressed are those of the authors.

the assumptions about performance functions. Their approach deals with convex performance functions such as $\mathcal{H}_\infty$ over (linear) models given in closed forms. Like ours, their approach samples the stochastic parameters. However, well-known inequality bounds such as the Chernoff-Hoeffding bounds and the Uniform Convergence of Empirical Means are used to relate the required sample size with error bounds between the estimated performance and the actual value. Using these building blocks, they propose approaches such as sequential design and scenario-based design that find parameter values by solving convex problems [8]. In contrast, our approach does not make assumptions about the performance functions and treats the model as a black box. Simulation through sampling and regression are used to construct models. This generality comes at a cost: whereas our approach provides probability guarantees on the final answer (if one is found), it does not provide guarantees on the search process for this answer.

The parameter tuning problem has been considered by Zhang *et al.* [9]. Their approach fits a linear approximation of the quantile function, which is used to search for a design point in the entire space. This is suboptimal since single linear approximations can hardly capture complex quantile functions. In contrast, our approach uses a piecewise approximation that combines models built over subsets of the parameter space. It is shown that our approach outperforms theirs which could not handle the apoptosis model studied in Section V-C.

Our approach builds on advances in statistical model checking (SMC). SMC combines simulation with hypothesis testing to deduce useful bounds on the probability of satisfying temporal properties with high confidence [10]–[12]. It can be viewed as a synthesis approach, which finds parameter values that enables a statistical model checker to certify the probabilistic performance requirements of a system. Such an example can be found in the work of Palaniappan *et al.* [13] and Jha *et al.* [14]. Our approach differs in that before attempting to optimize, it first uses simulation to model the quantile function. As a result, our approach utilizes far fewer simulations. There are also work in the SMC community that aim to verify the correctness properties under the worst case values of non-deterministic parameters [15], [16].

II. Problem Setup

This section introduces some preliminary notions and provides an overview of the proposed approach. First, we define the notion *black-box systems* as follows.

Definition 2.1 (Black-Box System): A black-box system $\mathcal{M}$ is a tuple $\langle \mathcal{P}, \mathcal{Q}, r_1, \ldots, r_n \rangle$ where $\mathcal{P} \subseteq \mathbb{R}^m$ is a set of *tunable parameter*, $\mathcal{Q} \subseteq \mathbb{R}^k$ is a set of *stochastic parameters*, and $r_1, \ldots, r_n$ are response functions $r_i : \mathcal{P} \times \mathcal{Q} \mapsto \mathbb{R}$ mapping each parameter value to a real-valued response.

In a black-box system, tunable parameters can be assigned to fixed values. In contrast, stochastic parameters are uncontrollable and follow an assumed distribution $\mathcal{D}$. For simplicity, we fix $n = 1$, considering just one response function. Response functions are assumed computable, either through simulation or physical measurements. Having fixed the tunable parameter values, a response can be regarded as a random variable whose distribution depends on the stochastic parameters and the dynamics of the system. Let $r_i(\mathbf{p})$ denote the random variable as a function of $\mathbf{p}$ for response r_i for $\mathbf{q} \sim \mathcal{D}$.

Problem statement: Given a black-box system $\mathcal{M}$, we intend to find tunable parameter values $\mathbf{p} \in \mathcal{P}$ such that the with a probability of at least θ, the response is below a threshold r_0.

$$\text{find } \mathbf{p} \in \mathcal{P} \text{ s.t. } \underset{\mathbf{q}\sim\mathcal{D}}{\mathsf{Prob}}(r(\mathbf{p}, \mathbf{q}) \leq r_0) \geq \theta \,. \tag{1}$$

Eq. (1) provides a constraint r_0 on the θ-quantile of the response r. As such, Eq. (1) is called a *chance constraint*. We may alternatively express a related formulation wherein the maximum permissible level r_0 is formulated as an objective function. To do so, we recall the notion of a θ-quantile.

Definition 2.2 (Quantile): Let X be a random variable and $\theta \in (0, 1)$. The θ-quantile of X, $\ell_\theta(X)$, is the value x below which the cumulative density is θ, i.e., $\mathsf{Prob}(X \leq x) = \theta$.

Recall that $r(\mathbf{p})$ is a random variable that characterizes the spread in response for a fixed $\mathbf{p}$. Denote $\ell_{r,\theta}(\mathbf{p})$ as the θ-quantile of the random variable $r(\mathbf{p})$. Eq. (1) can be written as the problem of finding $\mathbf{p} \in \mathcal{P}$ such that

$$\ell_{r,\theta}(\mathbf{p}) \leq r_0 \,. \tag{2}$$

Eq. (2) is a feasibility problem that answers "yes" or "no". In many cases, it is better to consider the optimization version of this problem that minimizes the quantile function,

$$\min_{\mathbf{p}\in\mathcal{P}} \ \ell_{r,\theta}(\mathbf{p}) \,. \tag{3}$$

For responses that larger values are preferred to smaller ones, their negations are taken into Eq. (2) and Eq. (3).

Our goal is to solve the chance-constrained problem (1) given a system $\mathcal{M}$, a probability θ and a response threshold r_0. The search for a tunable parameter $\mathbf{p}$ is formulated as an optimization in Eq. (3) that terminates as soon as $\mathbf{p}$ is found such that $\ell_{r,\theta}(\mathbf{p}) \leq r_0$. In the case that responses are not required to be optimized (e.g., Section V-C), the feasibility formulation in Eq. (2) is used instead.

Example 2.1: Figure 1 shows a closed-loop control system [17]. The plant consists of two rigid bodies and a spring. The masses and the spring constants are uniformly distributed stochastic parameters, $m_1 = 1.0 \pm 20\%$, $m_2 = 1.0 \pm 20\%$ and $k = 1.0 \pm 20\%$. A force u is applied to m_1 and the position of m_2 is measured as the output y. A lead compensator controls y to track the change of y_{ref}, a reference position. It has two tunable parameters: the pole $p \in [-1200, -800]$ and the zero $z \in [-1.2, -0.8]$. Consider two responses (1) r_s - the settling time, and (2) r_o - the overshoot as a percentage of the steady state value. With the constraints $\mathsf{Prob}(r_s \leq 2.5) \geq 0.95$ and $\mathsf{Prob}(r_o \leq 15\%) \geq 0.95$, our approach constructs approximations of the quantile functions $\ell_{r_s,0.95}$ and $\ell_{r_o,0.95}$. A global search on the approximations leads to the optimal tunable parameter values $p = -1200$ and $z = -0.93$.

A. Overview of Proposed Approach

Figure 2 shows an overview of the proposed approach. The idea is to construct a piecewise model $h_{r,\theta}(\mathbf{p})$ of the quantile function $\ell_{r,\theta}(\mathbf{p})$ using simulation, which substitutes $\ell_{r,\theta}$ in Eq. (2) or Eq. (3). Simulation is done with randomly sampled tunable and stochastic parameters. The former are

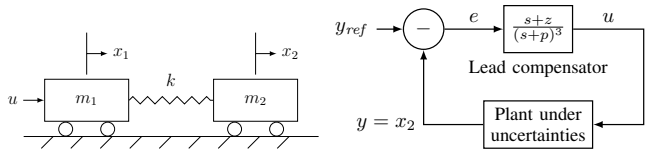

Fig. 1: The plant (left) and the closed loop system (right).

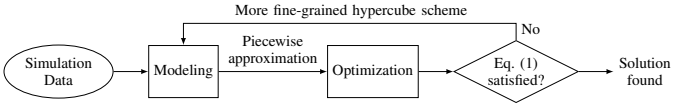

Fig. 2: A high-level flow of the proposed approach.

sampled uniformly to explore the space $\mathcal{P}$, and the latter are sampled according to the distribution $\mathcal{D}$. For systems with a large number of tunable parameters, it is expensive to generate data that sufficiently explore the space $\mathcal{P}$. To best utilize the existing data, we introduce a procedure called *marginalization*. It constructs a series of restricted models $g_1(\mathbf{p}), \ldots, g_k(\mathbf{p})$ for overlapping subsets $\mathcal{S}_1, \ldots, \mathcal{S}_k \subseteq \mathcal{P}$. For any point $\mathbf{p} \in \mathcal{P}$, the approximated quantile value is implicitly defined by averaging the models that cover $\mathbf{p}$.

Once the approximation $h_{r,\theta}$ is built, we perform simulated annealing to find a global minimum $\mathbf{p}^*$ of $h_{r,\theta}$. For an approximation with small error, $\mathbf{p}^*$ is expected to be close to the actual minimizer. A check is performed to determine whether $\mathbf{p}^*$ satisfies Eq. (1). Failing this, we refine our approximation by a finer-grained subdivision of the parameter space.

B. Feasibility Check

The check of whether $\mathbf{p} \in \mathcal{P}$ satisfies Eq. (1) serves as the termination criterion for our search and is performed at the end of each iteration. However, unless the response function r is given in a simple closed form and the distribution $\mathcal{D}$ is explicitly defined (e.g., normal distribution), performing this check can be prohibitively expensive if not impossible.

We resort to statistical hypothesis testing to decide Eq. 1. It consists of deciding between two competing hypotheses,

$$\mathcal{H}_1 : \underset{\mathbf{q}\sim\mathcal{D}}{\mathsf{Prob}}(r(\mathbf{p},\mathbf{q}) \le r_0) \ge \theta\,,\ \mathcal{H}_2 : \underset{\mathbf{q}\sim\mathcal{D}}{\mathsf{Prob}}(r(\mathbf{p},\mathbf{q}) \le r_0) < \theta\,,$$

and has been studied using various sequential hypothesis testing techniques in statistics, including the use of Wald's sequential probability ratio test [18] and Jeffreys' Bayes factor test [19]. Recent work on *statistical model checking* has proved the power of these approaches to check probabilities of property satisfaction in stochastic systems [10]–[12].

Another approach using Chernoff-Hoeffding bounds is suggested by Tempo *et al.* [6], [7]. It estimates θ^*, the probability of $r(\mathbf{p}^*, \mathbf{q}) \le r_0$, with an error bound ϵ and confidence level $\delta \in (0,1)$. The idea is to choose a sample $(\mathbf{q}_1, \ldots, \mathbf{q}_N)$ of size $N \ge -\log(\delta)/2\epsilon^2$, and compute the fraction $\hat{\theta}_N$ that $r(\mathbf{p}^*, \mathbf{q}_i) \le r_0$. Through Chernoff bounds it is shown that

$$\mathsf{Prob}(|\theta^* - \hat{\theta}_N| > \epsilon) \le \delta\,.$$

Example 2.2: For approaches based on Chernoff bounds, with $\epsilon = \delta = 0.01$, a sample of size $N \sim 23000$ is required. The sample guarantees that we can approximate θ^* with an error bound $\epsilon = 0.01$ with a probability of $1 - \delta = 0.99$. On the contrast, sequential hypothesis tests do not attempt to estimate θ^*. Rather, they examine enough evidence to decide between $\mathcal{H}_1$ and $\mathcal{H}_2$. Furthermore, the nature of the guarantees are "softer" than those provided by the application of Chernoff bounds that hold regardless of the actual value of θ^*. Therefore, in our experience, hypothesis testing techniques need much smaller sample sizes.

In this paper, we use Bayesian sequential hypothesis testing "out-of-the-box" [11] to test Eq. (1). We set the threshold for the Bayes factor to 100 in our experiments.

III. Quantile Function Modeling

This section introduces how to model a quantile function $\ell_{r,\theta}$ using simulation data. Our approach relies on two techniques, quantile regression and marginalization. The former is a regression technique that fits functions to a desired quantile of data. The latter is heuristic to construct piecewise models. It is particularly useful when data are limited and cannot "cover" the entire parameter space. These two techniques together result in a piecewise approximation of the quantile function.

A. Quantile Regression

Consider the problem of approximating the quantile function $\ell_{r,\theta}(\mathbf{p})$ given a set of simulation data. For a simple case, suppose that a linear function of the form

$$g_\theta(\mathbf{p};\mathbf{c}) = c_0 + \sum_{i=1}^{k} c_i p_i$$

is used to model $\ell_{r,\theta}(\mathbf{p})$, where $\mathbf{c} = (c_0, c_1, \ldots, c_k)$ are unknown coefficients and p_i is the ith tunable parameter. Conceptually, the coefficients $\mathbf{c}$ are calculated by minimizing the error between $\ell_{r,\theta}$ and g_θ. However, it is unlikely that $\ell_{r,\theta}$ is available in practice. In fact, it is usually impossible to explicitly evaluate $\ell_{r,\theta}(\mathbf{p})$ for a given $\mathbf{p}$.

One solution to this problem is known as quantile regression, which fits functions to some quantile of data. We provide a brief sketch and refer the interested readers elsewhere [1]. Given the simulation data $\{\mathbf{p}_i, \mathbf{q}_i, r(\mathbf{p}_i, \mathbf{q}_i)\}$ of size N, quantile regression uses the following loss function,

$$\rho_\theta(\mathbf{e}) = \theta \sum_{e_i > 0} |e_i| + (1-\theta) \sum_{e_i \le 0} |e_i|\ , \tag{4}$$

where $\mathbf{e} = (e_1, \ldots, e_N)$, and $e_i = r(\mathbf{p}_i, \mathbf{q}_i) - g_\theta(\mathbf{p}_i)$ is the error between the response and the approximation. Unless $\theta = 0.5$, Eq. (4) places asymmetric losses on positive and negative errors. For $\theta > 0.5$ ($\theta < 0.5$), positive (negative) errors incur more loss and are minimized. The loss function (4) leads to the following optimization problem.

$$\min_{\mathbf{c}} \rho_\theta(\mathbf{e})\ . \tag{5}$$

Since Eq. (4) is piecewise linear (and thus convex), the optimization has a unique minimum. Eq. (5) can be solved

as a linear program [1] by adding auxiliary variables $\mathbf{s} = (s_1, \ldots, s_N)$ and $\mathbf{t} = (t_1, \ldots, t_N)$, which is then written as

$$\begin{aligned} &\min_{\mathbf{c}} \ \theta \sum_{i=1}^{N} s_i + (1-\theta) \sum_{i=1}^{N} t_i \\ &\text{subject to} \\ &r\left(\mathbf{p}_i, \mathbf{q}_i\right) - g_\theta\left(\mathbf{p}_i; \mathbf{c}\right) = s_i - t_i, \quad i = 1, 2, \ldots, N\,, \\ &\mathbf{s} \geq 0\,, \ \mathbf{t} \geq 0\,. \end{aligned} \tag{6}$$

Intuitively, $\mathbf{s}$ encodes the positive errors and $\mathbf{t}$ encodes the negative ones. Since the difference between s_i and t_i equals to the error e_i, at least one of them should be zero so that the objective value is minimized.

B. Piecewise Model

Quantile regression extends naturally from fitting linear models to nonlinear (e.g., polynomial) ones that are expressible as a linear combination of chosen basis functions. Hence, it can directly construct a single model of $\ell_{r,\theta}(\mathbf{p})$ that minimizes the approximation error. This approach has been used by Zhang *et al.* [9]. However, it requires a large sample to adequately "cover" the parameter space $\mathcal{P}$. Furthermore, since the shape of $\ell_{r,\theta}$ is not known *a priori*, the use of a single functional form to fit the entire space is unlikely to be scalable.

An alternative is to subdivide $\mathcal{P}$ into hypercubes by choosing a partition along each dimension p_i, and fit a model for each hypercube. If the hypercubes are small enough and we are able to draw sufficiently large samples, guarantees can be obtained to bound the error between the approximations and $\ell_{r,\theta}$ restricted to each hypercube. However, subdividing $\mathcal{P}$ is expensive for even a few parameters. With each dimension p_i partitioned into K_i subsets, $\mathcal{P}$ is subdivided into $\prod_{i=1}^{n} K_i$ hypercubes. Furthermore, constructing a separate model for each hypercube is also impractical unless we can afford drawing samples in proportion to the number of hypercubes. For a fixed sample size, there will be very few observations in each hypercube due to the curse of dimensionality.

As such, we see that both extremes are problematic. Single functional models cannot be accurately constructed in general, whereas a full tiling of the parameter space cannot scale beyond a few tunable parameters.

C. Marginalization

To address these difficulties, we propose a solution called *marginalization* to implicitly construct piecewise models without incurring an exponential cost. The idea is that for a hypercube, instead of drawing samples within the hypercube and fitting a model, we fit multiple models, each of which is a superset of the hypercube. These models are called *M-models*. For any point $\mathbf{p}$ in the hypercube, an approximation of $\ell_{r,\theta}(\mathbf{p})$ is taken to be the average of the approximations from the M-models, thus implicitly defining an approximation for the hypercube. Let m be the number of tunable parameters. Marginalization is done in the following steps.

1) The range I_i of each parameter p_i is partitioned into K_i intervals $I_{i,1}, \ldots, I_{i,K_i}$;
2) For each interval $I_{i,k}$ of parameter p_i, define a *M-cell* as
$$C_{i,k} = \{p_i \in I_{i,k} \text{ and } \forall j \neq i\,.\, p_j \in I_j\}\,.$$
A set of M-cells is *complete* if it has a size of m and each M-cell is for an interval of a unique parameter;
3) For each M-cell, apply quantile regression to fit a linear model $g_{i,k}$ (i.e., an M-model) of the quantile function $\ell_{r,\theta}$ using simulation data within the M-cell;
4) Each hypercube is the intersection of a complete set of M-cells, $H_{k_1,\ldots,k_m} = \bigcap_{i=1}^{m} C_{i,k_i}$. For any point $\mathbf{p} \in H_{k_1,\ldots,k_m}$, a linear approximation of $\ell_{r,\theta}$ is defined as
$$h_{k_1,\ldots,k_m}(\mathbf{p}) = \frac{1}{m} \sum_{i=1}^{m} g_{i,k_i}(\mathbf{p})\,.$$

We call the partitioning of parameter ranges in Step 1 a *subdivision* of the parameter space. Note that Step 4 in the above procedure is not actually executed. It merely shows how to define the piecewise model from M-models. In Section IV, the piecewise model is constructed "on-the-fly" during optimization. It can be seen that the process of averaging models results in averaging the error between the quantile function $\ell_{r,\theta}$ and the individual models $g_{i,k}$. Figure 3 shows an illustration of the procedure for a simple two-dimensional case.

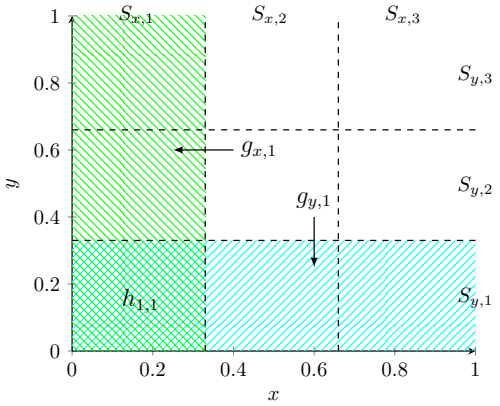

Fig. 3: Illustration of marginalization. The ranges of both x and y are partitioned into three intervals. The model $h_{1,1}$ of the bottom-left cube is the average of the models $g_{x,1}$ and $g_{y,1}$ for M-cells $S_{x,1}$ and $S_{y,1}$.

For a subdivision that partitions each dimension p_i into K_i intervals, a system with m tunable parameters has $\sum_{i=1}^{m} K_i$ M-cells. Therefore, marginalization takes $\sum_{i=1}^{m} K_i$ quantile regressions to approximate $\ell_{r,\theta}$. Compared with the case of explicit model construction for each hypercube, this approach clearly incurs much lower cost in computation. Furthermore, it does not require data in each hypercube, which is convenient and also reduces the need for simulation.

IV. Optimization

Given a subdivision of the parameter space, the marginalization procedure leads to an implicitly defined piecewise linear approximation $h_{r,\theta}$ of the true quantile function $\ell_{r,\theta}$. It is defined as for all $(k_1, \ldots, k_m)$,

$$h_{r,\theta}(\mathbf{p}) = \frac{1}{m} \sum_{i=1}^{m} g_{i,k_i}(\mathbf{p}) \text{ if } \mathbf{p} \in H_{k_1,\ldots,k_m}\,. \tag{7}$$

An optimization with piecewise linear objective is obtained by substituting Eq. (7) for $\ell_{r,\theta}$ in Eq. (3). However, there is no

guarantee that the resulting problem is well-defined, since $h_{r,\theta}$ is generally discontinuous (thus non-convex) and implicitly defined with exponentially many pieces.

Instead of using $h_{r,\theta}$ as a single function, notice that each piece is a linear model for the corresponding hypercube. Finding the optimum in the hypercube with respect to the linear model only involves selecting an appropriate vertex. Therefore, we restrict our search to the discrete vertices in the tiling of the state space and utilize simulated annealing over this lattice to find a global optimal point for $h_{r,\theta}$.

Conceptually, the simulated annealing algorithm performs a random walk over the lattice by moving from one hypercube to a neighboring hypercube. Upon doing so, the optimal loss value of the new hypercube is computed. If it improves over the previous best loss, the move is accepted and the search proceeds from the new hypercube. Otherwise, the move may be accepted or rejected depending on the outcome of a coin toss. The probability of accepting non-optimal moves is initially high but subsequently lowered according to a *cooling schedule*. When temperature is high, the algorithm explores the state space randomly since there is a large chance to accept a hypercube with a worse loss value. As temperature cools, the algorithm focuses on local search, resulting in locally optimal solutions. It can be shown that with a proper cooling schedule, simulated annealing converges to the global optimal solution [20]. In this paper, we adopt a linear cooling schedule with a cooling coefficient of 0.8.

Once a solution $\mathbf{p}^*$ is found, we need to check whether it indeed satisfies $\mathsf{Prob}(r(\mathbf{p}^*, \mathbf{q}) \leq r_0) \geq \theta$. As mentioned in Section II-B, this is done by Bayesian sequential hypothesis testing. If the check fails, a finer-grained subdivision of the tunable parameter space is chosen, and the modeling and optimization procedure is performed again. In practice, there is not a general guild-line for the granularity of subdivisions, which should be decided experimentally.

Optimizing over multiple responses. In the presence of multiple responses, optimality needs to be defined in sophisticated ways. For example, let ℓ_{r_1} and ℓ_{r_2} be the quantiles of two responses r_1 and r_2. Is $\ell_{r_1} = 1$ and $\ell_{r_2} = 100$ better than $\ell_{r_1} = 0.9$ and $\ell_{r_2} = 110$? The answer depends on both the importance of the responses and their "usual" magnitude of variations. To handle multiple responses, we introduce the following scalarized objective function

$$\min_{\mathbf{p}\in\mathcal{P}} \sum_{i=1}^{n} w_i z_i h_{r_i,\theta}(\mathbf{p}),$$

where n is the number of responses, w_i is the weight that represents the relative importance of the ith response, and z_i is a normalization constant that brings the values of $h_{r_i,\theta}$ to a comparable range. The weights are set to 1 by default and can be specified by designers. The normalization constant z_i is taken to be the inverse of $h_{r_i,\theta}(\mathbf{p})$ at the nominal tunable parameter values. The rationale is that the nominal parameter values represent the performance level on average.

V. Experimental Evaluation

We evaluate the proposed approach using three benchmark examples. They include an insulin pump model for type-1 diabetic patients, a flight control model for fixed-wing aircraft, and an extrinsic apoptosis reaction model. The first two models are control systems suffering from stochastic variations. The third one is an ODE model derived from the biochemical kinetics of protein interactions, used to study the fundamental process of cell death. We include it to demonstrate that our parameter synthesis technique can be applied to benchmarks other than control systems.

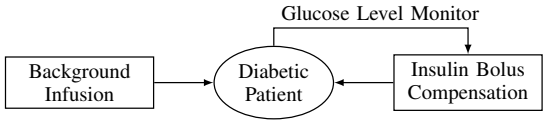

Fig. 4: A model of an insulin pump.

For all the benchmarks, we consider three cases, each of which utilizes different numbers of simulations for regression (i.e., 500, 1000 and 2000 respectively). We fix the initial number of subdivisions to be 5 along each dimension. Upon each iteration, it is multiplied by a factor of 1.5, which is experimentally determined.

For comparison, we implement the approach by Zhang *et al.* [9]. All the experiments are performed on a Macbook Pro running OSX 10.10.3, which has a 2.6GHz Intel Core i5 CPU and 8GB memory. The implementation is done in Matlab R2015a. The convex optimization solver CVX [21] is used to solve linear programs.

A. Insulin Infusion Pump

Insulin infusion pumps are commonly used by type-1 diabetic patients to control their blood glucose levels. These pumps supply insulin at programmable rates over time. Typically, the use of insulin infusion pumps has two components: (a) continuous background infusion provided at a fixed *basal rate* to offset the endogenous glucose production, and (b) a fixed amount of insulin bolus to cover elevated glucose levels, especially after a meal. Figure 4 shows a diagram for such pumps. Usually, the basal rate (basalRate) is set by trial and error until the glucose level remains steady during fasting conditions (e.g., overnight). Likewise, the bolus dosage is calculated based on a fixed insulin to carbohydrate ratio (icRatio) and a correction factor (cor). The parameter icRatio reflects for a unit of carbohydrate in a meal, how much insulin is required to compensate the increase of glucose levels. On the other hand, the correction factor cor determines the dosage when glucose levels become higher than desired. Note that all the three parameters are patient-specific.

Ideally, human blood glucose level should be between 70 $\mathrm{mg/dL}$ and 180 $\mathrm{mg/dL}$. A level below 70 $\mathrm{mg/dL}$ induces *hypoglycemia*, and a level above 180 $\mathrm{mg/dL}$ is considered *hyperglycemia*. In recent years, mathematical models of the insulin-glucose regulatory system emerge to help find robust parameter values that yield a good control over glucose levels. This is especially important since varying timing and carbohydrate amount of meals can deteriorate the effectiveness of a predefined infusion scheme, thus causing hypoglycemia or hyperglycemia. The goal of this case study is to find robust parameter values under meal uncertainties.

We use a pump model developed by Sankaranarayanan *et al.* [2], which is based on the human insulin-glucose regulation model proposed by Dalla Man *et al.* [22]. As mentioned, the model has three tunable parameters, basalRate, icRatio and cor. Simulation is done for a 24-hour period, during which patients take five meals. The start time and the duration of each meal, and the amount of carbohydrate in each meal are assumed to follow certain distributions, leading to a total of 15 stochastic parameters. A virtual set of in-silico patients published by Dalla Man *et al.* [22] is employed for parameter synthesis. In this experiment, we consider hypoglycemia more critical than hyperglycemia since it can cause seizures, unconsciousness and even death. Hence, with G_{min} being the lowest glucose level during simulation, we require that the probability of hypoglycemia is at most 1%,

$$\mathsf{Prob}(G_{min} \geq 70\mathrm{mg/dL}) \geq 0.99\,.$$

Besides, we intend to optimize the overall glucose level of patients, which sets up the following optimization problems

$$\min \ell_{-G_{min},\theta}\,,\ \min \ell_{G_{max},\theta}\,, \text{and } \min \ell_{-R,\theta}\,.$$

The response G_{max} is the highest glucose level and should be minimized. R is the percentage of time that the glucose level is between 70 $\mathrm{mg/dL}$ and 180 $\mathrm{mg/dL}$. Both G_{min} and R should be maximized, leading to the use of their negations in the above formulation. We set $\theta = 0.99$. Since it is acceptable as long as $G_{min} \geq 70\mathrm{mg/dL}$, we assign a weight of 0.2 to G_{min} and 1 to the other two responses.

We synthesize the three tunable parameters, basalRate $\in [0.1, 1]$, icRatio $\in [0.01, 0.1]$ and cor $\in [0.01, 0.1]$, for three different patients. The results are shown in Table I. The nominal parameter values are tuned for each patient without taking the meal uncertainties into account. Comparing the performance at the predicted parameter values by our approach (column "Performance") and the nominal performance (column "Nominal") we see that for all three patients, the peak glucose level is lowered and the time without hyperglycemia is extended. For patient 1 and 3, the resulting parameter values also move the glucose level trajectories upward, thus staying away from hypoglycemia. We also include a comparison with the approach proposed by Zhang *et al.* [9] under the "zhang2014" column [1]. In most cases, our approach results in a better-performed system in terms of G_{max} and R. For G_{min}, both approaches bring it above 70 $\mathrm{mg/dL}$. Hence, we conclude that our approach outperforms their approach in general.

B. Aircraft Flight Control

Orientation control in three dimensions is a critical subject in the study of fixed-wing aircraft dynamics. Rotations about the vehicle's center of mass generate three flight dynamics parameters, pitch, roll and yaw, which are angles from a defined steady flight equilibrium state. These angles are also known as angle of attack (AOA), bank angle and sideslip angle (i.e., heading), respectively.

In aerospace engineering, fixed-wing aircrafts are often modeled as control systems. Such a system includes actuators,

[1]Due to limited space, we only show the resulting performance values of their approach. A complete table is available at https://www.dropbox.com/s/7813xjp5x66fe1s/iccad_table.pdf?dl=0.

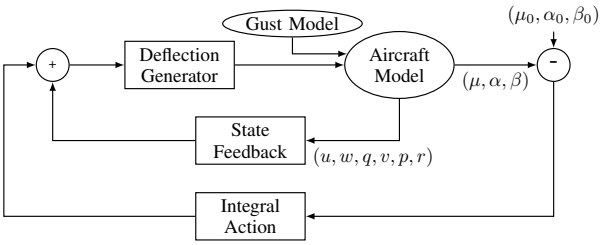

Fig. 5: An aircraft flight control model.

which exert forces in various directions, and generate rotational forces or moments about the aerodynamic center of the aircraft, and thus rotate the aircraft in pitch, roll, or yaw. Figure 5 shows a control system model, which is available from the Matlab Robust Control Toolbox (R2015a). The aircraft is modeled as a 6th-order linear system with state matrix $\mathbf{A}$ and input matrix $\mathbf{B}$. The state variables include the velocities (u, v, w) on the three dimensions, the pitch rate q, the roll rate p and the yaw rate r. The output consists of the bank angle μ, the angle of attack α, and the sideslip angle β. By setting proper amount of deflections for elevators, ailerons and the rudder, the controller controls the three angles μ, α and β so that they track changes of the reference angles μ_0, α_0 and β_0.

The controller consists of a state feedback component and an integral control component, which are modeled by two gain matrices, $\mathbf{K}_x$ and $\mathbf{K}_i$, respectively. The matrix $\mathbf{K}_x$ has a size of 3×6, and $\mathbf{K}_i$ of 3×3. Thus, we have a total of 27 tunable parameters. The stochastic parameters arise from modeling uncertainties in the state matrix $\mathbf{A}$ and the input matrix $\mathbf{B}$, and the stochastic wind disturbance. This leads to a total of 73 stochastic parameters.

This case study aims to find values of $\mathbf{K}_x$ and $\mathbf{K}_i$ that yield the best tracking performance on μ, α and β. The performance is defined by the settling time and the percentage of overshoot from the steady state of each angle. These performance metrics should satisfy

$$\mathsf{Prob}(\text{settling time} \leq 7.5s) \geq 0.99\,,$$
$$\mathsf{Prob}(\text{overshoot} \leq 8\%) \geq 0.99\,.$$

To achieve this, we minimize the 0.99-quantile of these performance metrics. With $\theta = 0.99$, this leads to the minimization of the following quantile functions (s stands for "settling time" and o for "overshoot")

$$\ell_{r,\theta}\,, r \in \{\mu_s, \mu_o, \alpha_s, \alpha_o, \beta_s, \beta_o\}\,.$$

For parameter synthesis, a reasonable range is assigned to each parameter in the matrix $\mathbf{K}_x$ and $\mathbf{K}_i$.

An interesting comparison with our method is to search for tunable parameter values at random and test if the resulting responses satisfy Eq. (1). We randomly choose 200 points in the tunable parameter space and use Bayesian sequential hypothesis testing to test (1). The process takes around 8 hours, since each test itself requires tens or hundreds of simulations. Only 6 out of 200 pass the hypothesis testing. As a result, finding a design point at random that satisfies our criteria has a chance as low as 3%, which makes it a hard problem for a pure guessing approach.

ID	Sim	Time(s)	Iter	Params	Performance			Nominal			zhang2014		
				basalRate, icRatio, cor	G_{min}	G_{max}	R	G_{min}	G_{max}	R	G_{min}	G_{max}	R
1	500+471	59	1	0.64, 0.082, 0.028	97.3	228.6	80%	68.7	236.2	73%	100.1	251.4	72%
	1000+998	479	2	0.73, 0.100, 0.012	96.2	231.9	79%				103.0	250.7	71%
	2000+459	86	1	0.82, 0.082, 0.017	98.2	235.2	78%				99.5	245.8	74%
2	500+459	79	1	1.00, 0.100, 0.028	91.3	164.7	100%	81.1	185.3	97%	99.4	177.5	100%
	1000+554	60	1	0.46, 0.100, 0.066	90.2	159.9	100%				100.9	178.3	100%
	2000+1075	407	2	1.00, 0.100, 0.028	91.3	164.7	100%				95.4	169.2	100%
3	500+459	68	1	1.00, 0.100, 0.026	76.1	210.5	79%	70.7	223.8	67%	88.7	234.5	73%
	1000+459	63	1	1.00, 0.082, 0.028	85.9	209.1	74%				89.4	236.2	73%
	2000+488	83	1	1.00, 0.082, 0.023	92.5	211.0	68%				85.2	221.2	75%

TABLE I: Insulin infusion pump results for different patients. Meaning of the columns: "ID" - patient ID, "Sim" - number of simulations used for regression and for feasibility check respectively, "Time" - running excluding simulation, "Iter" - number of subdivisions tried, "Params" - predicted tunable parameter values, "Performance", "Nominal" and "zhang2014" - true performance at predicted values from our approach, nominal values, and predicted values from the approach of Zhang *et al.* [9], calculated from 500 simulations.

Sim	Time(s)	Iter	Performance						zhang2014					
			μ_s(s)	μ_o	α_s(s)	α_o	β_s(s)	β_o	μ_s(s)	μ_o	α_s(s)	α_o	β_s(s)	β_o
500+558	257	1	5.9	5.2%	6.6	7.9%	5.1	6.5%	Solution not found					
1000+459	400	1	5.9	5.0%	7.4	3.5%	4.7	1.2%	Solution not found					
2000+493	481	1	5.9	5.2%	7.4	3.6%	4.6	3.4%	6.1	5.5%	6.9	6.4%	7.1	3.8%

TABLE II: Results for aircraft flight control model. Columns have the same meaning as in Table I. The nominal performance for the six responses are 6.0s, 8.8%, 7.5s, 12.5%, 6.2s and 11.5%, respectively.

Our results are shown in Table II. The nominal values of $\mathbf{K}_x$ and $\mathbf{K}_i$ are tuned without taking modeling uncertainties and wind gust into account. Comparing "Performance" with the nominal performance mentioned in the table description, we observe significant performance improvement on overshoot and minor improvement on settling time. The approach "zhang2014" is not able to find solutions in the cases that 500 and 1000 simulations are used for regression. This is not surprising since their approach uses a single linear model to approximate the whole parameter space. For the case of 2000 simulations, our approach out-performs their approach on five out of six responses.

C. Extrinsic Apoptosis Reaction Model

Apoptosis, the process of programmed cell death, is essential for the healthy development of organisms. Disruption of apoptosis can lead to many diseases, such as Alzheimer's disease and cancer [23]. The extrinsic apoptosis reaction model (EARM) characterizes dynamics of the amounts of key proteins involved in apoptosis triggered by the TRAIL signaling molecule. The model is described as ODEs derived from the law of mass action for biochemical kinetics[2].

The model consists of 69 state variables representing the active and inactive form of proteins and their complexes. A diagram of the key proteins in the pathway are shown in Figure 6. Dynamics of the biochemical reactions are governed by kinetic rate constants, which represent the rates of binding, dissociation and catalysis reactions. Values of these parameters are unknown and cannot be directly measured. In this case study, we assume a total of 71 unknown rate constants, which are treated as tunable parameters. EARM is especially

[2]A specific version of the model, EARM1.3 [3], is used.

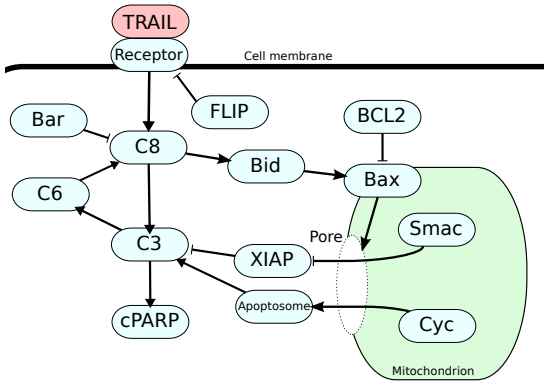

Fig. 6: A simplified diagram of EARM (version 1.3).

well suited to understand variability in the cell death process among individual cells. Key aspects of this variability can be explained by different total protein amounts in each cell [24]. Experiments have shown that the total protein amounts across cells follow a log-normal distribution [3]. Consequently, we regard the total protein amounts as stochastic parameters with a log-normal distribution.

Our goal is to tune the values of the 71 kinetic parameters such that the model responses are consistent with observed behaviors of the system. We investigate the delay between the initiation and the final commitment to cell death, which is known as effector caspase delay. The extent of the delay reveals whether cell death is reached through a fast, direct activation cascade or through a slower, indirect process involving the mitochondrion. The delay can be quantified by measuring the difference between the time points when cleaved Bid and cleaved PARP reach 10% of their total amounts. We require

Sim	Time(s)	Iter	Performance		Nominal	
			D_{min}	D_{max}	D_{min}	D_{max}
500+1541	809	3	180	1055		
1000+459	294	1	360	2162	2220	7800
2000+459	323	1	360	2130		

TABLE III: Results for EARM. Columns have the same meaning as in Table I. The performance values are in second.

that the delay D is nonnegative and at most one hour with 95% probability, i.e.,

$$\mathsf{Prob}(D_{min} \geq 0) \geq 0.95 \text{ and } \mathsf{Prob}(D_{max} \leq 3600s) \geq 0.95\,.$$

This is consistent with observations for cell death induced by a fast, direct pathway. Since there is no requirement to minimize or maximize the delay, this problem is formulated as a feasibility problem

$$\ell_{-D_{min},\theta} \geq 0 \text{ and } \ell_{D_{max},\theta} \leq 3600s\,.$$

Notice that $-D_{min}$ is used as the response in place of D_{min}.

As in the previous example, we randomly choose 200 points in the parameter space to learn how hard it is to pick up suitable parameter values by pure chance. Out of the 200 points, 12 of them result in delay values lying between 0 and 1 hour. Once again, this problem is not likely to be solved by a pure guessing strategy.

Figure III shows the results of synthesizing the parameter values. The nominal parameter values of this model have been estimated with respect to experimental data [3]. As shown by the columns under "Nominal", these values clearly do not support cell death through a fast, direct pathway. Parameter values estimated from our approach result in delay values between 0 and 1 hour, and thus find the parameters corresponding to the behavior of interest. Notice that although results from the case of 500 simulations are quite different than those from the other two cases, all of them satisfy the Bayesian sequential hypothesis testing procedure used to check the chance constraints. For this benchmark, the approach by Zhang *et al.* [9] is not able to find solutions.

VI. Conclusion

In summary, this paper introduces a simulation-based technique for parameter synthesis of black-box systems under stochastic variations. The parameter synthesis problem is treated as chance-constrained optimization, in which a response function of interest is expected to satisfy given requirements with a desired probability. The overall idea of the proposed approach is to construct a piecewise linear approximation of a certain quantile of the response function. A global optimization scheme searches for optimal tunable parameter values on this approximation. Experiments in this paper demonstrate that our approach is able to find parameter values that result in satisfactory performance, and scales well with the dimension of systems.

References

[1] R. Koenker, *Quantile regression*. Cambridge university press, 2005, no. 38.

[2] S. Sankaranarayanan, C. Miller, R. Raghunathan, H. Ravanbakhsh, and G. Fainekos, "A model-based approach to synthesizing insulin infusion pump usage parameters for diabetic patients," in *Proc. Allerton Conference*, 2012, pp. 1610–1617.

[3] S. Gaudet, S. L. Spencer, W. W. Chen, and P. K. Sorger, "Exploring the contextual sensitivity of factors that determine cell-to-cell variability in receptor-mediated apoptosis," *PLoS Computational Biology*, vol. 8, no. 4, p. e1002482, 2012.

[4] A. Ben-Tal and A. Nemirovski, "Robust convex optimization," *Mathematics of Operations Research*, vol. 23, no. 4, pp. 769–805, 1998.

[5] A. Nemirovski and A. Shapiro, "Convex approximations of chance constrained programs," *SIAM Journal on Optimization*, vol. 17, no. 4, pp. 969–996, 2006.

[6] C. M. Lagoa, F. Dabbene, and R. Tempo, "Hard bounds on the probability of performance with application to circuit analysis," *IEEE Transactions on Circuits and Systems*, vol. 55, no. 10, pp. 3178–3187, 2008.

[7] R. Tempo, G. Calafiore, and F. Dabbene, *Randomized Algorithms for Analysis and Control of Uncertain Systems With Applications*. Springer, 2010.

[8] M. C. Campi, S. Garatti, and M. Prandini, "The scenario approach for systems and control design," *Annual Reviews in Control*, vol. 33, no. 2, pp. 149–157, 2009.

[9] Y. Zhang, S. Sankaranarayanan, and F. Somenzi, "Statistically sound verification and optimization for complex systems," in *ATVA*, ser. Lecture Notes in Computer Science, vol. 8837. Springer, 2014, pp. 411–427.

[10] H. L. S. Younes and R. G. Simmons, "Probabilistic verification of discrete event systems using acceptance sampling," in *CAV*, 2002, pp. 223–235.

[11] S. K. Jha, E. M. Clarke, C. J. Langmead, A. Legay, A. Platzer, and P. Zuliani, "A Bayesian approach to model checking biological systems," in *CMSB*, 2009, pp. 218–234.

[12] P. Zuliani, A. Platzer, and E. M. Clarke, "Bayesian statistical model checking with application to simulink/stateflow verification," in *HSCC*. ACM, 2010, pp. 243–252.

[13] S. K. Palaniappan, B. M. Gyori, B. Liu, D. Hsu, and P. Thiagarajan, "Statistical model checking based calibration and analysis of bio-pathway models," in *Computational Methods in Systems Biology*. Springer, 2013, pp. 120–134.

[14] S. K. Jha, R. Datta, C. Langmead, S. Jha, and E. Sassano, "Synthesis of insulin pump controllers from safety specifications using bayesian model validation," in *Proceedings of 10th Asia Pacific Bioinformatics Conference, (APBC)*, 2012.

[15] D. Henriques, J. Martins, P. Zuliani, A. Platzer, and E. Clarke, "Statistical model checking for markov decision processes," in *QEST'12*, 2012.

[16] C. Ellen, S. Gerwinn, and M. Fränzle, "Statistical model checking for stochastic hybrid systems involving nondeterminism over continuous domains," *Journal of Software Tools for Technology Transfer*, Jan 2014.

[17] B. Wie and D. S. Bernstein, "A benchmark problem for robust control design," in *American Control Conference*, May 1990, pp. 961–962.

[18] A. Wald, "Sequential tests of statistical hypotheses," *The Annals of Mathematical Statistics*, vol. 16, no. 2, pp. 117–186, 1945.

[19] R. E. Kass and A. E. Raftery, "Bayes factors," *Journal of the American Statistical Association*, vol. 90, no. 430, pp. 774–795, 1995.

[20] D. Bertsimas, J. Tsitsiklis *et al.*, "Simulated annealing," *Statistical science*, vol. 8, no. 1, pp. 10–15, 1993.

[21] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.1," http://cvxr.com/cvx, Mar. 2014.

[22] C. Dalla-Man, R. Rizza, and C. Cobelli, "Meal simulation model of the glucose-insulin system," *IEEE Transactions on Biomedical Engineering*, vol. 54, no. 10, pp. 1740–1749, 2007.

[23] S. Shimohama, "Apoptosis in Alzheimer's disease – an update," *Apoptosis*, vol. 5, no. 1, pp. 9–16, 2000.

[24] S. L. Spencer, S. Gaudet, J. G. Albeck, J. M. Burke, and P. K. Sorger, "Non-genetic origins of cell-to-cell variability in TRAIL-induced apoptosis," *Nature*, vol. 459, no. 7245, pp. 428–432, 2009.