2019

Parameter Estimation of Stochastic Models Against Probabilistic Temporal Logic Behavioral Specifications

Arfeen Khalid
University of Central Florida

PARAMETER ESTIMATION OF STOCHASTIC MODELS AGAINST PROBABILISTIC TEMPORAL LOGIC BEHAVIORAL SPECIFICATIONS

by

ARFEEN KHALID
M.S. Lahore University of Management Sciences, 2013
B.S. University of Engineering & Technology Lahore, 2010

A dissertation submitted in partial fulfilment of the requirements
for the degree of Doctor of Philosophy
in the Department of Computer Science
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Summer Term
2019

Major Professor: Sumit Kumar Jha

ABSTRACT

The inherent behavioral variability exhibited by stochastic systems makes it a challenging task for human experts to manually analyze them. Computational modeling of such systems helps in investigating and predicting the behaviors of their underlying processes but at the same time introduces the presence of several unknown parameters. A key challenge faced in this scenario is to determine the values of these unknown parameters against known behavioral specifications. The solutions that have been presented so far estimate the parameters of a given model against a single specification whereas a correct model is expected to satisfy all the behavioral specifications when instantiated with a single set of parameter values.

The main contribution of this thesis is computing a *quantitative tightness metric* describing how well a given stochastic model satisfies a known probabilistic behavioral specification and later employing that metric to guide a search algorithm in order to estimate all the unknown parameters present in the model such that the model satisfies multiple probabilistic temporal logic behavioral specifications *simultaneously*; thus, generating a single set of parameter values against multiple specifications. The first step of the presented solution uses a larger mean hypothesis test based statistical model checking technique to estimate the unknown parameters of the given stochastic model against a *single* probabilistic temporal logic behavioral specification and the second phase of this work extends it by using a multiple hypothesis testing based statistical model checking technique to estimate the parameters against *multiple* probabilistic behavioral specifications simultaneously. The benchmarks studied, analyzed and experimented on in this study are stochastic rule-based computational models of two biochemical receptors, FcϵRI and T-cell. Experimental results demonstrate successful parameter estimation of all the unknown parameters present in the two models against three probabilistic temporal logic behavioral specifications each.

To the people of my beloved country, Pakistan.

ACKNOWLEDGMENTS

All praises are due to God Almighty, the only worthy of worship, the most gracious and the most merciful. Firstly, I would like to express my sincere gratitude to my advisor, Dr. Sumit Kumar Jha for giving me the opportunity to learn and embark on this PhD journey. His guidance and support over the years has helped me make progress in academic research. Besides my advisor, I would like to thank the rest of my dissertation committee members Dr. Gary T. Leavens, Dr. Sharma V. Thankachan and Dr. Talat S. Rahman, for their precious services and valuable comments on my research work. Thanks to all the present and past colleagues of my research lab. I am also grateful to all my former teachers and professors, without which I would not have been here today. I gratefully acknowledge the financial support of National Science Foundation (NSF) through awards #1822976, #1422257 and #1438989 without which this research would not have been possible.

My special thanks goes to my husband Salman Anwar Khokhar and my brother Bilal Khalid who have continuously supported me during my good and bad times with their love, encouragement and optimism. I am also grateful to my friends Aisha Urooj Khan, Sehar Butt, Ramsha Dogar, Amna Mukhtar and Hiba Shahid for their continued support. Finally, but most importantly, I wish to thank my parents for giving me the upbringing, with their selfless dedication and sacrifices, that I have come this far.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

It has now been widely acknowledged that traditional deterministic mathematical models are not always sufficient to capture the dynamical behaviors of a stochastic system [33]. The characteristics not adequately captured by these traditional methods manifest themselves as apparent unpredictability within the system. Such random behaviors represent aspects of modeling stochastic systems that should not be disregarded and demand more refined modeling techniques. In this study, our focus and experimentation relies on a relatively complex discipline of stochastic systems known as biochemical networks.

Computational modeling of biochemical networks assists in understanding the functioning of a biochemical process as well as enables in silico experimentation. Various biological modeling languages [19, 24, 14, 23, 47, 35] can be used to describe stochastic biochemical systems, such as cell signaling pathways. In this study, our choice of a stochastic model is driven by a relatively recent technique known as rule-based modeling. Stochastic rule-based models serve as a natural and compact representation for biochemical reactions. The Gillespie stochastic simulation algorithm [20] and its variants [21] are then employed to predict the behavior of the biochemical system modeled by the stochastic model.

However, it is often not feasible to create a complete stochastic rule-based model from first principles. Instead, our knowledge of the biochemical system is used to obtain the set of chemical reactions or the core structure of the stochastic rule-based model. The lack of knowledge about the rate constants of biochemical reactions is readily modeled as unknown parameters in these models.

A primary challenge in the use of such a parameterized stochastic rule-based model for predicting the behavior of a biochemical system is the determination of the parameters of

the model from multiple experimental observations. Traditionally, parameter values of a stochastic model have been estimated against quantitative time series data. However, the focus of these efforts has been on discovering parameter values from a single probabilistic temporal logic specification. In practice, a biochemical model must satisfy multiple experimental observations made on the biological system being modeled [3]. Hence, it is important to estimate a single set of parameter values that causes a parameterized stochastic model to satisfy multiple probabilistic temporal logic specifications simultaneously.

Parameter estimation of computational models against probabilistic temporal logic specifications is usually comprised of two steps: (i) model checking that formally verifies a computational model against a known behavioral specification, and (ii) a search mechanism to explore the parameter space of the given model. Most of the model verification approaches [43, 25, 45, 41] employed for stochastic models focus on calculating a qualitative value that specifies whether a model satisfies a behavioral specification or not whereas we compute a quantitative measure that describes how well a parameterized rule-based stochastic biochemical model satisfies a given behavioral temporal logic specification. To encode the expected behavioral properties of a given stochastic model we use a probabilistic variant of the Signal Temporal Logic (STL) [16].

This dissertation presents the design of a new approach that utilizes a *quantitative tightness metric* to validate the model against multiple probabilistic temporal logic behavioral specifications *simultaneously* and to guide a simulated annealing based search in order to estimate a single set of parameter values for a parameterized stochastic model. The first step of the presented solution uses a larger mean hypothesis test based statistical model checking technique to estimate the parameters of the given stochastic model against a *single* probabilistic temporal logic behavioral specification; then, the second phase of this work uses a multiple hypothesis testing based statistical model checking technique to estimate the model parame-

ters against *multiple* probabilistic specifications simultaneously. In addition, the algorithms presented in this study are not limited to parameter estimation of only biological models but are applicable to models belonging to a wide range of areas.

We study two stochastic rule-based models of biochemical receptors, namely, FcϵRI and T-cell as our benchmarks to evaluate the usefulness of the presented method. Our experimental results demonstrate successful estimation of the unknown parameters present in the two models against three probabilistic temporal logic behavioral specifications each. We also report the final values of parameters found by our parameter estimation algorithm.

CHAPTER 2: BACKGROUND

In this section we build up some necessary background knowledge involving terms that are used frequently thorough out this document such as stochastic models, model checking, and rule-based modelling using BioNetGen software. We also formally define and explain probabilistic Signal Temporal Logic (STL) that is used to encode the behavioral specifications of our benchmark models.

2.1 Stochastic Models

Stochastic models are used to abstract those particular real world systems which are known to exhibit random or dynamic behaviors, i.e. behaviors that cannot be predicted without using sophisticated mathematical or statistical techniques. Unlike deterministic models that generate an output with absolute certainty, a parameterized stochastic model can produce a different output every time it is initialized with the same parameters values. One can say that stochastic models are an extension of deterministic models that are capable of embracing more complex variations in the dynamics of a system [15]. These complex variations are an integral part of various physical world processes. In other words, a single solution trajectory of a stochastic model only depicts a single realization of the system and does not represent the entire behavior of the model [2]. The presence of such uncertainty leads to generating several solution trajectories in order to learn the complete behavior of the model over a given time period.

Stochastic models can be used to represent dynamical behaviors of systems that belong to a variety of different areas such as biology, chemistry, physics, ecology, neuroscience,

economics, finance, signal processing, control systems and many others. Stochastic systems are often modelled as discrete-time Markov chains (DTMCs), continuous-time Markov chains (CTMCs), and stochastic differential equations (SDEs) [15]. Our methods presented in this study are applicable to both stochastic as well deterministic parameterized models. The Gillespie stochastic simulation algorithm [20] and its variants [21] are often employed to predict the behavior of the stochastic systems modeled by the stochastic model.

2.2 Model Checking

Model checking is a technique for automatically verifying a finite-state system against its correctness specifications. The correctness specifications represent certain known behaviors of the system and are generally encoded in some kind of formal propositional temporal logic such as Linear Temporal Logic (LTL), Signal Temporal Logic (STL), Computational Tree Logic (CTL), Continuous Stochastic Logic (CSL) or others [17, 34]. Given a computational model of a system and a formal specification, model checking determines whether the model satisfies the specification or not [4]. Model checking is an effective technique to uncover potential system errors and is applicable to both software and hardware systems. The process of model checking allows to incorporate the verification steps during the design phase of a system which reduces overall verification time, effort and cost of manufacturers. The main principle on which model checking relies involves an exhaustive search of the state space of the given model in order to check if a certain property holds for a state in that model or not.

Model checking techniques are often build upon numeric, symbolic or statistical methods. In this study, our main focus is using different flavors of statistical model checking that are based on using a quantitative measure of model satisfaction.

2.2.1 Statistical Model Checking

As the unknown parameters in a given stochastic model increase, the numerical and symbolic methods to determine the correctness of the model suffer from state space explosion problem; hence, become more computational intensive [48]. However, solutions based on statistical model checking (SMC) avoids this problem by using a combination of simulation and statistical methods to reason about the system's behavioral specifications expressed in some temporal logic. The central idea behind SMC is to obtain some simulation samples and then validate them against a behavioral specification to infer whether the model satisfies or violates the specification with some degree of confidence [36, 54, 49]. The validation phase generally uses hypothesis testing [53, 42] in order to generate a statistical evidence of specification validation. Although statistical methods do not guarantee accurate solutions but they are known to require far less memory and computational time when compared to numerical or symbolic methods [55].

2.3 Rule-based Modeling

Rule-based modeling is a relatively new formalism that allows to model biochemical systems by representing molecules as structured objects and molecular interactions as rules [19]. The interactions can be of various types including associations, dissociations, modifications to the internal state of a molecule as well as the production or consumption of molecular species [39]. In other words, a rule specifies how the states of reactants are modified to generate products in a biochemical reaction. Molecular interactions can result in a large number (hundreds to thousands) of possible molecular species. Conventional approaches, involving translating the model into ODEs or CTMCs, are unable to handle such combinatorial complexity. Rule-based modeling addresses this problem by expressing models with a high degree of modularity

and avoid the explicit enumeration of all possible molecular species or all the states of a system; hence, providing a succinct representation of the model [19]. The rules are then simulated to generate a reaction network comprised of all chemically distinct species and reactions [8]. Therefore, a rule-based model is a compact and generalized representation of conventional biochemical models.

2.4 BioNetGen Software

Biological Network Generator (BioNetGen) is an open source software tool that can be used to construct, visualize, simulate and analyze rule-based models. The tool is based on a formal language, known as BioNetGen language (BNGL), for specifying molecules and rules to model biological systems. A BNGL model file is composed of six primary blocks [7], namely, *(i) parameters:* include rate constants and values for initial concentrations of species and are responsible for governing the dynamics of the system, *(ii) molecule types:* define molecules, including components and allowed component states, *(iii) seed species:* describe the initial state of the system, *(iv) observables:* output functions of concentrations of species having particular attributes, *(v) reaction rules:* describe how molecules interact with each other and *(vi) actions:* provide various methods for generating and simulating the network. In our experiments, we use the stochastic simulation algorithm (SSA) implemented in the software to simulate our biochemical models. Each model simulation results in a time-stamped simulation trace capturing the behavior of the model when instantiated with a particular set of parameter values.

2.5 Signal Temporal Logic encoding behavioral specifications

The process of model verification requires that the acceptable behaviors of a model are represented as temporal logic formulas [17] so that computational methods can be applied to check model satisfiability against such behaviors. In this study, we use Signal Temporal Logic (STL) [44, 16] to encode the known behaviors of a stochastic biochemical model. We use a time bounded variant of STL where all temporal operators are associated with a lower and upper time-bound. The logical operators in STL formulas consist of $\wedge$ (and), $\vee$ (or), $\neg$ (negation), and temporal operators consist of **G** (global), **F** (future), and **U** (until).

Definition 1 (Signal Temporal Logic)**.** *A Signal Temporal Logic formula representing a model's known behavior is defined recursively by the following grammar:*

$$\phi := \mu \mid \neg\mu \mid \phi_1 \wedge \ \phi_2 \mid \phi_1 \vee \ \phi_2 \mid \mathrm{G}_{[t_1,t_2]}\ \phi \mid \mathrm{F}_{[t_1,t_2]}\ \phi \mid \phi_1\ \mathrm{U}_{[t_1,t_2]}\ \phi_2$$

where ϕ, ϕ_1 and ϕ_2 are STL formulas, $0 \leqslant t_1 < t_2 < \infty$ and μ is a predicate whose value is determined by the sign of a function of an underlying *signal* $\mathbf{x} \in \mathbb{R}$, i.e., $\mu \equiv \mu(\mathbf{x}) > 0$ [44, 27]. In this study, signal $\mathbf{x}$ is termed as a simulation trace or simulation trajectory. The validity of a formula ϕ with respect to signal $\mathbf{x}$ at time t is defined inductively as follows:

$$\begin{aligned}
(\mathbf{x},t) \vDash \mu \quad &\Leftrightarrow \quad \mu(x_t) > 0 \\
(\mathbf{x},t) \vDash \neg\mu \quad &\Leftrightarrow \quad \neg((\mathbf{x},t) \vDash \mu) \\
(\mathbf{x},t) \vDash \phi_1 \wedge \phi_2 \quad &\Leftrightarrow \quad (\mathbf{x},t) \vDash \phi_1 \wedge (\mathbf{x},t) \vDash \phi_2 \\
(\mathbf{x},t) \vDash \phi_1 \vee \phi_2 \quad &\Leftrightarrow \quad (\mathbf{x},t) \vDash \phi_1 \vee (\mathbf{x},t) \vDash \phi_2 \\
(\mathbf{x},t) \vDash \mathrm{G}_{[t_1,t_2]}\phi \quad &\Leftrightarrow \quad \forall t' \in [t+t_1, t+t_2] \text{ s.t.} (\mathbf{x},t') \vDash \phi \\
(\mathbf{x},t) \vDash \mathrm{F}_{[t_1,t_2]}\phi \quad &\Leftrightarrow \quad \exists t' \in [t+t_1, t+t_2] \text{ s.t.} (\mathbf{x},t') \vDash \phi \\
(\mathbf{x},t) \vDash \phi_1 \mathrm{U}_{[t_1,t_2]}\phi_2 \quad &\Leftrightarrow \quad \exists t' \in [t+t_1, t+t_2] \text{ s.t.} (\mathbf{x},t') \vDash \phi_2 \\
&\qquad\qquad \wedge \forall t'' \in [t,t'], (\mathbf{x},t'') \vDash \phi_1
\end{aligned}$$

A signal $\mathbf{x}$ satisfies ϕ, denoted by $\mathbf{x} \vDash \phi$, if $(\mathbf{x},0) \vDash \phi$. Informally, $\mathbf{x} \vDash \mathrm{G}_{[t_1,t_2]}\phi$ if ϕ holds at every time step between t_1 and t_2, and $\mathbf{x} \vDash \mathrm{F}_{[t_1,t_2]}\phi$ if ϕ holds at some time step between t_1 and t_2. Also, $\mathbf{x} \vDash \phi_1 \mathrm{U}_{[t_1,t_2]}\phi_2$ if ϕ_1 holds at every time step before ϕ_2 holds, and ϕ_2 holds at some time step between t_1 and t_2. Due to the stochastic nature of the given biochemical models, their behavioral specifications are also often probabilistic in nature. Therefore, we use a probabilistic variant of Signal Temporal Logic to encode model behaviors. Probabilistic STL can be further explained with the help of following example:

Example 1. *Consider the following probabilistic Signal Temporal Logic formula:*

$$P_{\geq 0.85}(\mathbf{G}_{[0,100]}((GProtein > 6000) \wedge \mathbf{F}_{[100,200]}(GProtein < 6000)))$$

It says that G protein should always have a high value i.e. greater than 6000 units during the first 100 (0-100) time units and should fall below 6000 at some point in time during the next 100 (100-200) time units with a probability of at least 0.85.

To verify the model against a given probabilistic STL property we use the algorithm implemented by a software framework Temporal Logic Extractor (TeLEx) [27]. Given a time-stamped simulation trace and a STL behavioral specification formula, TeLEx quantifies the degree of satisfiability by computing a tightness metric that describes how well a model satisfies the specification. TeLEx uses smooth functions, such as sigmoid and exponentials, to compute tight-satisfiability of STL formulas. It returns a positive value in case the STL formula is verified successfully; the larger the value the better it satisfies the behavior. If the model is not able to satisfy the STL formula, the algorithm returns a negative value describing how far off is the model from satisfying the specification; for further details and examples we refer the readers to [27]. In the coming chapters, we discuss how a TeLEx generated tightness metric is used in model checking and in guiding the search process of our parameter estimation algorithms.

CHAPTER 3: PARAMETER ESTIMATION AGAINST A SINGLE PROBABILISTIC TEMPORAL LOGIC SPECIFICATION

This chapter[1] describes a larger mean hypothesis testing based statistical model checking approach to estimate the unknown parameters of a stochastic model against a single probabilistic temporal logic behavioral specification. We first describe some of the related work that has been carried out in the recent past followed by the formal problem definition. We then explain the detailed methodology and our experimental results.

3.1 Related Work

One of the crucial steps in the process of parameter estimation is to check the model against behavioral temporal logic specifications. Statistical Model Checking (SMC) [36] is a popular method among many well-studied model checking techniques available in the literature. A detailed survey [57] on various SMC techniques indicates that Statistical Model Checking based on Wald's Sequential Probability Ratio Test (SPRT) [51] is widely studied [43, 25, 45] and used in practice. Another variant of SMC [29, 26] uses Bayesian Sequential Hypothesis Testing and improves performance by incorporating prior knowledge about the model being verified. Mancini et al. [41] propose a parallel SMC algorithm to yield model parameters of biological systems represented as Ordinary Differential Equations (ODEs).

Satisfiability Modulo Theory (SMT) has also been used to perform model checking of bio-

[1]The research reported in this chapter was published in 2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM) [30].

logical models [40]. It works by first extracting a collection of ODEs from a given model and then formulating these ODEs along with time-series data into a collection of SMT problems. Another approach [10] performs model checking based on an open source symbolic model checker, NuSMV [13], and formulates a given model using Binary Decision Diagrams (BDDs). These BDDs provide a compact way of representing boolean functions which in turn represent the different states of the given model.

A recent study [39] presents a SMC based parameter estimation framework for rule-based models formulated in BioNetGen. It verifies experimental data as well as qualitative properties of the given model. Wang et al. [52] extends the rule-based BioNetGen language to enable the specification of interactions among more than one cell. They employ SMC in order to analyze system properties and obtain interesting insights into the development of novel therapeutic strategies for pancreatic cancer. Techniques based on applying convolutional neural networks [56] in order to learn temporal formulas have also been recently proposed.

However, each of these methods is limited to verifying the model with a binary (yes/no) outcome whereas we compute a quantitative tightness metric describing how well the given model satisfies a behavioral specification.

3.1.1 Approaches using a quantitative measure of model satisfaction

Several attempts have been made to estimate parameters of stochastic systems using a quantitative measure of how well a model satisfies a known temporal logic behavioral specification. Rizk et al. [46] define a continuous degree of satisfaction of a temporal logic property which is then used as a fitness function in order to find kinetic parameters of a biochemical model. However, their continuous degree of satisfaction is limited to providing a quantitative measurement only in case of dissatisfaction and remains zero in case the model satisfies a

temporal logic property. Whereas we compute a tightness metric that provides a quantitative measure irrespective of whether or not the model satisfies a system property; thus, giving a more meaningful interpretation of model verification.

Bartocci et al. [5] uses Gaussian Process Upper Confidence Bound (GP-UCB) to compute a distribution of a quantitative satisfaction function specifying the degree of satisfaction of a property by the model. An average of this distribution is later used to guide the parameter search. Some other techniques [9, 12, 11] perform a global exploration of the parameter space of stochastic biochemical systems using probabilistic model checking, where for each parameter point they compute approximate upper and lower bounds of a landscape function that returns a quantitative value. This value is based on the probability that the model satisfies a given CSL (continuous stochastic logic) formula. Another software framework uses the Bayesian formalism for parameter estimation and model selection [37]. Euclidean distance (sum of squares) between the observed data and a simulated trajectory is computed; a parameter point is accepted if this distance is less than a threshold value.

However, each of these methods are limited to verifying only a single temporal logic specification of the model at one time whereas we also present, in the next chapter, a multiple hypothesis testing technique which enables us to validate the model against all given specifications simultaneously; hence, generating a single set of parameter values.

3.2 Problem Definition

Our approach aims to estimate the unknown parameters of a rule-based stochastic model using statistical model checking combined with a simulated annealing [32, 1] search so that the model satisfies a given Signal Temporal Logic (STL) specification with at least a user-

specified probability. We now formally define our problem.

Definition 2 (Single specification parameter estimation problem)**.** *Given a parametric stochastic rule-based model $\mathcal{M}(\rho)$ with unknown parameter set $\rho \in \mathbb{R}_{\geq 0}$, a desired specification ϕ in Signal Temporal Logic (STL), and a required probability $\theta \in (0,1)$, find a set of parameter values ρ_0 such that $\mathcal{M}(\rho_0) \vDash P_{\geq\theta}(\phi)$.*

Intuitively, we seek a set of parameter values ρ_0 so that the model satisfies ϕ with probability at least θ when it is instantiated with the set ρ_0.

3.3 Methodology

Our proposed methodology employs a new statistical model checking technique based on the hypothesis test proposed by Hayre et al. [22] that sequentially selects the larger of the means of two normal distributions. Due to the stochastic nature of the given model, model simulations using the same set of parameter values exhibit varying behaviors. The hypothesis test generates several BioNetGen simulation traces before reaching a decision of whether the model satisfies a behavioral property or not. Each of these simulation traces is then verified against a behavioral property, using TeLEx [27], which returns a quantitative tightness metric specifying how well the model satisfies the behavioral property. Hence, a distribution of TeLEx quantitative tightness metrics is formed. We consider two such distributions obtained by sequentially simulating the model with two different sets of parameter values; then, we make a comparison between them using the hypothesis test [22] to select the distribution with the larger mean. The greater the TeLEx tightness metric, the better the STL specification is satisfied by the model; hence, we look for the distribution with the larger mean. Once the hypothesis test selects a distribution, simulated annealing guided by the mean of the selected distribution explores the parameter space of the given model in order to estimate

the values of the unknown model parameters so that the model satisfies ϕ with probability at least θ.

In the following two subsections, we first describe our larger mean hypothesis test based statistical model checking approach and then explain how this approach is infused with the search algorithm in order to estimate model parameters.

3.3.1 Larger Mean Hypothesis Test

To understand the larger mean hypothesis test, consider $x_1, x_2, ...$ and $y_1, y_2, ...$ be two independent sequences of independent TeLEx tightness metrics obtained by simulating the model with two randomly generated sets of parameter values. The actual mean and variance of these distributions are unknown to us as we are generating them sequentially. Suppose that the model is simulated and verified m number of times for the first set of parameter values and n number of times for the second set of parameter values and $m \geqslant 1$, $n \geqslant 1$, then the estimated means for the two sequences can be calculated as,

$$\bar{x}_m = \frac{(x_1 + ... + x_m)}{m}, \quad \bar{y}_n = \frac{(y_1 + ... + y_n)}{n} \tag{3.1}$$

Setting the difference between the estimated means as $\delta_{mn} = \bar{x}_m - \bar{y}_n$, we perform the following sequential hypothesis test to select the distribution with the larger mean:

$$\begin{aligned} H_1 &: \delta_{mn} > 0 \\ H_2 &: \delta_{mn} < 0 \end{aligned} \tag{3.2}$$

Rejection of H_2 implies that $\bar{x}_m > \bar{y}_n$; hence, we select the set of parameter values associated with the first distribution. Similarly, rejection of H_1 implies that $\bar{y}_n > \bar{x}_m$, resulting in the selection of the set of parameter values associated with the second distribution. We then

calculate a ratio Z_{mn} as:

$$Z_{mn} = \frac{mn\delta_{mn}}{ms_n^2 + nr_m^2} \tag{3.3}$$

where,

$$\begin{aligned} r_m^2 &= \frac{1}{m-1}\sum_{i=1}^{m}(x_i - \bar{x}_m)^2 \\ s_n^2 &= \frac{1}{n-1}\sum_{j=1}^{n}(y_j - \bar{y}_n)^2 \end{aligned} \tag{3.4}$$

and apply the following termination rule when performing the hypothesis test:

$$\begin{aligned} &\text{If } Z_{mn} \geqslant b_{mn}, \text{ stop and reject } H_2; \\ &\text{If } Z_{mn} \leqslant -b_{mn}, \text{ stop and reject } H_1; \\ &\text{If } -b_{mn} < Z_{mn} < b_{mn}, \text{ continue sampling.} \end{aligned} \tag{3.5}$$

The stopping boundary b_{mn} is defined by any sequence that decreases to zero as m and $n \to \infty$. In our experiments, we choose $b_{mn} = b(m+n+6)/(m+n-6)$, with $b = 6$ as suggested by [22].

If the first distribution has a larger mean, then H_2 is rejected; otherwise, H_1 is rejected. If the means are fairly close, the test continues on drawing more samples i.e. it performs more model simulations until it has reached a decision.

The test requires a minimum number, say k, of BioNetGen simulations to be generated for both sets of parameter values before the actual test can begin so that the values r_m^2 and s_n^2 can be calculated without any errors. We take $k \geqslant 2$ which implies that at least two simulation traces would be generated for each set before the larger mean hypothesis test begins calculating Z_{mn}.

3.3.2 Search Algorithm

We apply simulated annealing [32, 1] to efficiently explore the parameter space of the given model in order to find a set of parameter values that satisfies the given Signal Temporal Logic (STL) specification with a given probability. It is an iterative method that uses the selected larger mean, generated from the hypothesis test, as its guide to search the given model's parameter space.

We explain the behavior of the search process with the help of Algorithm 1. The algorithm starts with a random set of parameter values ρ_1 in the parameter space of the given model (line 1). It then randomly perturbs ρ_1 to find one of its neighboring set of parameter values ρ_2 (line 2). These two sets are then passed on to our statistical model checking function in line 3 that performs larger mean hypothesis test (described in the previous subsection) to determine which of these two sets creates distribution with the larger mean indicating better model satisfaction of the given specification ϕ.

The hypothesis test function returns three values; (i) set of parameter values which resulted in the distribution with the larger mean ρ_{new}, (ii) the mean μ_{new} of the selected distribution and (iii) the acceptance probability θ_{acc} with which the selected set of parameter values satisfies ϕ. θ_{acc} is the ratio of successful simulation traces to total number of traces generated during the model checking phase until the hypothesis test makes its decision and stops. We then compare the acceptance probability θ_{acc} with the required probability θ in line 4 and return ρ_{new} if the model satisfies ϕ with probability greater than or equal to the required probability i.e $\theta_{acc} \geqslant \theta$. In case this does not happen, we start our main simulated annealing loop which runs from line 6 through line 20. Essentially, each new iteration of the loop explores a neighboring set of parameter values of the currently selected set ρ_{new} and selects the set that has the larger mean. However, in order to avoid a local optimum solution, the

Algorithm 1 Parameter Estimation Algorithm Against A Single Specification

Require:

$\mathcal{M}(\rho)$ Parameterized stochastic model
ϕ STL specification
θ required probability
t_i initial temperature
t_f final temperature
c cooling rate

Ensure:

ρ_0 such that $\mathcal{M}(\rho_0) \vDash P_{\geq\theta}(\phi)$

1: $\rho_1 \leftarrow rand()$
2: $\rho_2 \leftarrow$ FindANeighbour(ρ_1)
3: $\rho_{new}, \mu_{new}, \theta_{acc} \leftarrow$ LargerMean($\mathcal{M}$, ρ_1, ρ_2, ϕ)
4: **if** $\theta_{acc} \geqslant \theta$ **then return** ρ_{new} **end if**
5: $t \leftarrow t_i$
6: **while** $t \geqslant t_f$ **do**
7: $\mu_{old} \leftarrow \mu_{new}$
8: $\rho_{old} \leftarrow \rho_{new}$
9: $\rho_1 \leftarrow \rho_{new}$
10: $\rho_2 \leftarrow$ FindANeighbour(ρ_1)
11: $\rho_{new}, \mu_{new}, \theta_{acc} \leftarrow$ LargerMean($\mathcal{M}$, ρ_1, ρ_2, ϕ)
12: **if** $\theta_{acc} \geqslant \theta$ **then return** ρ_{new} **else**
13: **if** $\mu_{new} \leqslant \mu_{old}$ **then**
14: **if** $rand(0, 1) > e^{-(\mu_{new} - \mu_{old})/t}$ **then**
15: $\rho_{new} \leftarrow \rho_{old}$
16: **end if**
17: **end if**
18: **end if**
19: $t \leftarrow c * t$
20: **end while**
21: **return** "No parameter set found!"

algorithm also sometime selects a set of parameter values with the smaller mean but with a very low probability.

The main search loop starts the annealing process at a very high temperature t_i (line 5) and slowly cools it down after every iteration by multiplying it with a cooling factor c (line 19). Lines 7 and 8 store the previously selected set of parameter values and mean to ρ_{old} and μ_{old}

respectively, so that new values can be generated using larger mean hypothesis test (lines 9-11). Once again the acceptance probability θ_{acc} of the newly selected set ρ_{new} is compared with the required probability θ and ρ_{new} is returned in case $\theta_{acc} \geqslant \theta$ (line 12). If $\theta_{acc} < \theta$, the means of the two distributions are compared (line 13). If $\mu_{new} > \mu_{old}$, we already have a better set of parameter values ρ_{new} in our hands and we move onto the next iteration and start exploring its neighbors.

Lines 14-16 show that how simulated annealing avoids local optimum solutions by sometimes accepting sets of parameter values with lower means. At the beginning of annealing process, we have higher temperatures and tend to take risks by sometimes accepting worse values and focusing on exploring more space but as we move towards lower temperatures, this risk factor is reduced as demonstrated in line 14. This notion of slow cooling is interpreted as a slow decrease in the probability of accepting worse solutions as the parameter space is explored. Finally, if we are unable to find any set of parameter values that satisfy ϕ with probability at least θ after the final temperature is reached, we return this information as a message in line 21.

3.4 Experimentation

We study a rule-based T-cell receptor model, written in BNGL, as the benchmark to evaluate our single specification parameter estimation algorithm. The model is simulated using Gillespie's stochastic simulation algorithm (SSA) implemented in the BioNetGen software. We estimate the values of all 29 unknown parameters present in the model such that the model satisfies a given Signal Temporal Logic (STL) property with at least the required threshold probability. First, we briefly describe the T-cell receptor model and its properties and then explain our experimental results.

3.4.1 Benchmark: T-cell

Human and animal blood is composed of several different components, one of which is the white blood cells (or leukocyte) that are responsible for defending the body against intruders such as viruses, bacteria, and parasites. Lymphocytes are a kind of white blood cells that are known to specifically target an invading pathogen and are critical for initiating an adaptive immune response. A T-cell is a type of lymphocyte that is responsible for a variety of cell-mediated adaptive responses and therefore is an important part of the immune system. The cell surface receptors of a T-cell, known as T-cell receptors (TCRs), are able to detect the presence of foreign substances such as antigens by binding to specific polypeptide fragments displayed by a protein called the major histocompatibility complex (MHC). MHC recognizes compatible proteins and thus helps the immune system to determine compatibility between tissues of different individuals. T-cells avoid autoimmunity by (i) responding strongly to the presence of small quantities of antigen while simultaneously (ii) not responding at all to the large quantities of endogenous (host) peptide-MHC (pMHC). Thus, maintaining a fine balance between these two types of responses induces an element of uncertainty to the cell behavior. This means that when a computational model of a T-cell receptor is simulated with the same parameter values, it might generate different trajectories against each simulation. This uncertainty or stochasticity makes it an interesting and fitting choice for our parameter estimation problem.

As described earlier, in order to defend the immune system, T-cell must be able to detect foreign peptides while entirely ignoring self peptides [38]. We test this behavior by observing one of the model's key output i.e. the fraction of doubly phosphorylated extracellular-signal-regulated kinase ($ppERK$) which is also taken as a measure of T-cell activation. T-cell activation is critical for the initiation and regulation of the immune response since the activated T-

cells can scan the intracellular environment in order to target and destroy infected cells. For $(ppERK/totERK) < 0.10$, the cell is considered inactive and for $(ppERK/totERK) > 0.5$, the cell is considered active. Here, $totERK$ is the total concentration of ERK present in the T-cell. We estimate the parameter values of the T-cell model such that it satisfies the following three probabilistic STL properties:

Property 1: $P_{\geq 0.99}(\mathbf{G}_{[0,300]}(ppERK/totERK < 0.95))$

As mentioned before, cell activity inside a T-cell receptor can be detected by observing the level of $ppERK$. This property verifies that the fraction $ppERK/totERK$ always remains below 0.95 during the first 300 seconds of the simulation.

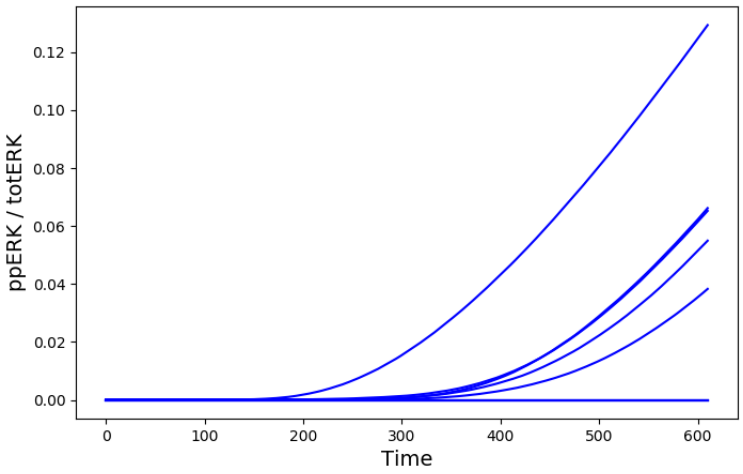

Figure 3.1: BioNetGen traces validating Property 1 of T-cell receptor

We also provide a visual representation of a series of BioNetGen traces successfully validating this property in Figure 3.1. It can be seen that the concentration of $ppERK$ to total concentration of ERK in the model always stays below the provided threshold (0.95).

Property 2:

$$P_{\geq 0.99}(\mathbf{F}_{[0,300]}((ppERK/totERK < 0.1) \wedge \mathbf{F}_{[300,600]}(ppERK/totERK > 0.5)))$$

Cell activation behavior can be observed by investigating the fraction of cells transiently activated as a result of simulation by a small number of agonist pMHC [38]. This property verifies if the system is able to achieve an active state after 300 seconds of the simulation provided that it was inactive at the start of the simulation.

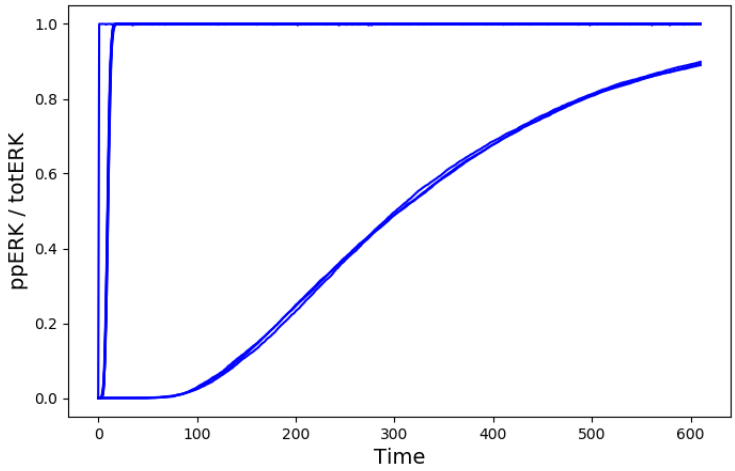

Figure 3.2: BioNetGen traces validating Property 2 of T-cell receptor

The traces shown in Figure 3.2 represent cell activation of the receptor. One can easily verify that the cell shows its inactive state during the first 0-300 seconds and eventually becomes active during the next 300-600 seconds.

Property 3:

$$P_{\geq 0.20}(\mathbf{F}_{[0,300]}((ppERK/totERK > 0.5) \wedge \mathbf{F}_{[300,600]}(ppERK/totERK < 0.1)))$$

Lipniacki et al. [38] explains how antagonist pMHC entirely inhibits the cell activation which was achieved earlier by agonist pMHC (Property 2). Third property monitors the deactiva-

tion behavior of T-cells which is a result of negative feedback activation mediated by protein tyrosine phosphatases (SHP1). It validates if the cell can switch from an active state during the first half of the simulation to an inactive state during the second half of the simulation. This cell deactivation behavior can be clearly observed from the traces shown in Figure 3.3.

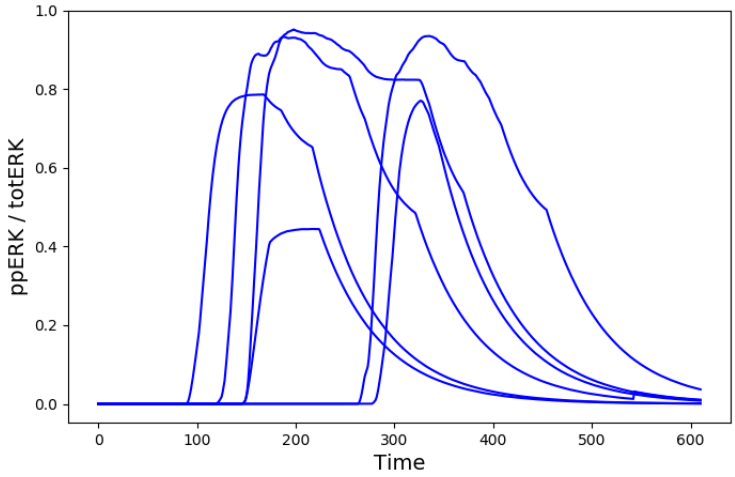

Figure 3.3: BioNetGen traces validating Property 3 of T-cell receptor

3.4.2 Experimental Results

We apply our single specification parameter estimation algorithm to the T-cell receptor rule-based model. This subsection shows how our technique successfully estimates all 29 unknown parameters of the model against three probabilistic STL properties described in the previous subsection.

We also compare our approach to two of the earlier statistical model checking based methods and demonstrate that our approach works up to *20 times* faster than a sequential probability ratio test (SPRT) model checking approach [25] and up to *4 times* faster than a Bayesian sequential model checking (BSMC) approach [26]. Table 3.1 shows the time taken by SPRT,

Table 3.1: Execution time of our approach compared to two earlier statistical model checking methods to estimate the parameters of T-cell model.

Property	SPRT [25] (hrs)	BSMC [26] (hrs)	Our Approach (hrs)	Prior best/ Our approach
First property	1.42	0.28	0.06	4.66
Second property	8.54	1.71	0.35	4.88
Third property	-	16.73	4.24	3.94

BSMC and our approach to estimate the unknown parameters of the T-cell receptor model against three probabilistic STL properties. It is clear that our method outperforms the two competing approaches. For the third property, SPRT was not able to find any set of parameter values before the annealing process cooled down.

Table 3.2 shows the actual estimated sets of parameter values against all three specifications obtained by our parameter estimation algorithm. The first two parameters N1 and N2 refer to agonist pMHC and antagonist pMHC respectively. Parameters 3-23 are the rate constants that govern the dynamics of reaction rules. Parameters 24-29 are the molecule types present in the model that are composed of components that can bind to each other, both within a molecule and between molecules.

Table 3.2: Estimated Parameters for the T-cell receptor model against three probabilistic STL behavioral specifications.

Param No.	Param Name	Param Value (Property 1)	Param Value (Property 2)	Param Value (Property 3)
1	pMHC(p∼ag)	4107.39	47.36	3244.07
2	pMHC(p∼en)	7031.39	13.87	9684.76
3	b1	8.14e-05	1.040e-07	9.55e-07
4	b2	0.00082	0.0013	0.00048
5	d1	611.01	0.0017	25.32
6	d2	0.061	3905.01	33.01
7	lb	0.007	6.67e-10	5.38e-10
8	ly1	5.63e-09	6.18e-06	0.017
9	ly2	0.00014	741.60	2.97
10	ls1	13.25	0.048	41.67
11	ls2	2.07e-06	2.15e-06	0.0037
12	tp	4.87e-05	2.43e-05	0.014
13	s0	0.011	2.84e-06	0.0100
14	s1	0.00065	0.01	0.0017
15	s2	0.026	0.63	0.51
16	s3	29.18	0.0020	0.108
17	z0	2.43e-10	1.39e-09	2.72e-10
18	z1	0.00087	0.015	1.18e-05
19	z2	0.00045	0.0031	0.0089
20	m1	0.0039	0.00060	0.00758
21	m2	7.37e-06	3.91e-06	0.018
22	e1	2.10e-09	4.46e-07	0.032
23	e2	3.97e-05	4.86e-06	0.0166
24	TCR	1466.51	3104.78	336.41
25	LCK	247625.33	1335.25	357017.49
26	ZAP	57509571.87	1055023836.6	3527381.64
27	MEK	674999.76	15873.66	203.43
28	ERK	2628592.63	9293.71	1154607.34
29	SHP	11774.56	1853.70	62.26

CHAPTER 4: PARAMETER ESTIMATION AGAINST MULTIPLE PROBABILISTIC TEMPORAL LOGIC SPECIFICATIONS

This chapter[1] extends the single specification parameter estimation problem presented in the previous chapter to estimating the unknown parameters of stochastic models against multiple probabilistic temporal logic behavioral specifications. We first formally define the multiple specification parameter estimation problem and then explain the proposed algorithm and our experimental results.

4.1 Problem Definition

We are interested in finding a set of parameter values $\rho_0 \in \mathbb{R}^n$ so that the model $\mathcal{M}$, when instantiated with set ρ_0, satisfies each specification ϕ_i with probability greater than or equal to its corresponding required probability r_i. We now formally define our problem.

Definition 3 (Multiple specification parameter estimation problem)**.** *Given a parameterized stochastic rule-based model $\mathcal{M}(\rho)$ with parameter set $\rho \in \mathbb{R}^n$, a set of desired specifications $\boldsymbol{\phi} = \{\phi_1, ..., \phi_k\}$ in Signal Temporal Logic, and a corresponding set of required probabilities $\boldsymbol{r} = \{r_1, ..., r_k\}$ where $r_i \in [0,1]$, find a set of parameter values ρ_0 such that the following holds:*

$$\mathcal{M}(\rho_0) \vDash P_{\geq r_1}(\phi_1) \wedge P_{\geq r_2}(\phi_2) \wedge \cdots \wedge P_{\geq r_k}(\phi_k)$$

[1] A preliminary version of the research reported in this chapter was published in 8th IEEE International Conference on Computational Advances in Bio and medical Sciences (ICCABS) 2018 [31].

4.2 Methodology

Our solution to the problem of estimating parameters from multiple specifications is illustrated in Figure 4.1. Given an initial set of parameter values ρ_{init}, we first simulate the given model $\mathcal{M}(\rho_{init})$ using BioNetGen to generate a time-stamped simulation trace. The simulation trace along with the given set of k STL specifications $\boldsymbol{\phi} = \{\phi_1, ..., \phi_k\}$ is then fed to TeLEx which returns k quantitative tightness metrics describing the distance between the model $\mathcal{M}(\rho_{init})$ and each of the k given specifications. TeLEx quantitative tightness metrics and the set of required probabilities $\boldsymbol{r} = \{r_1, ..., r_k\}$ are then passed onto multiple hypothesis testing (MHT) that decides whether the model instantiated with ρ_{init} satisfies the given probabilistic STL specifications or not.

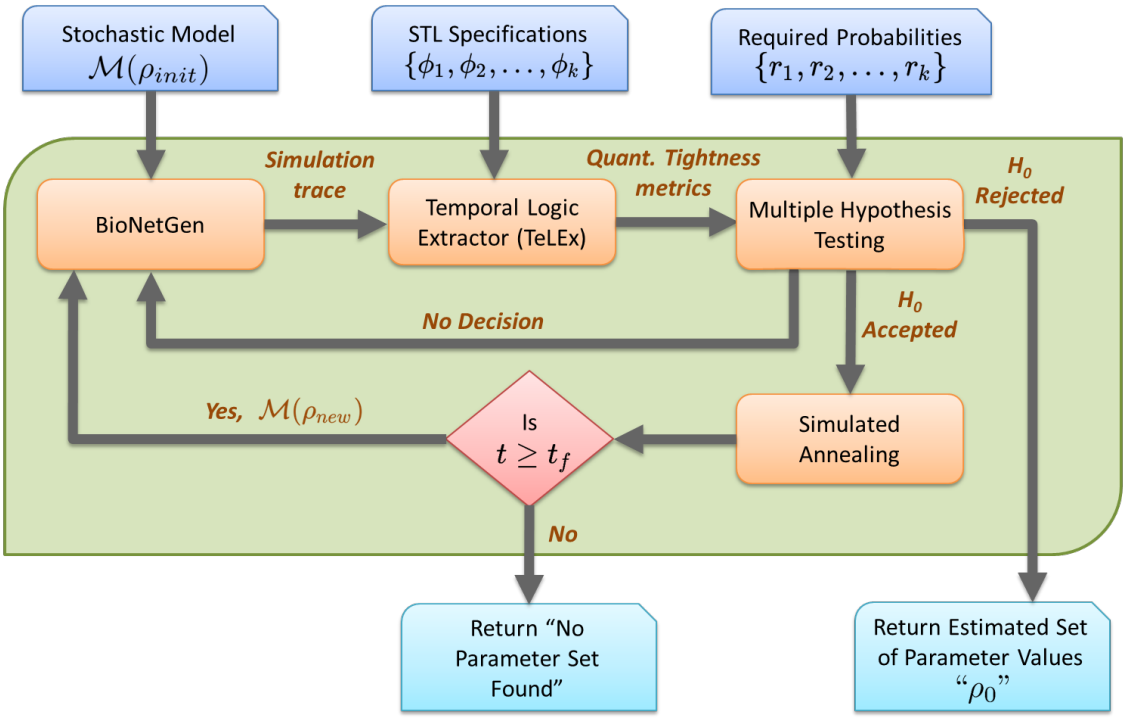

Figure 4.1: Multiple specification parameter estimation system using multiple hypothesis testing based statistical model checking.

The hypothesis test continues on drawing samples, i.e. continues on generating model simulations, until it has made a decision. If the test declares that $\mathcal{M}(\rho_{init})$ satisfies all the given specifications with greater than or equal to their corresponding required probabilities, then ρ_{init} is returned as the estimated set of parameter values. Otherwise a search algorithm, guided by the mean of quantitative tightness metrics, explores the parameter space of the given model in order to find a new set of parameter values and repeats the above process.

We explain the approach implemented in this study by dividing it into two phases. The first phase employs multiple hypothesis testing (MHT) based statistical model checking to check a given model against multiple STL behavioral specifications. The second phase of our method implements a simulated annealing based search algorithm to explore model's parameter space and finds a single set of parameter values which satisfies all the given probabilistic behavioral specifications simultaneously.

4.2.1 Multiple hypothesis testing based statistical model checking

As the problem (Definition 3) states, given a set $\boldsymbol{\phi}$ of k STL behavioral specifications we are required to generate a single set of parameter values such that the model satisfies each specification ϕ_i with probability at least r_i. In order to achieve this, the given model is first simulated using BioNetGen and the simulation trace is verified using TeLEx against each given specification. Essentially, TeLEx verification function is called k times and each call returns a quantitative tightness metric against the respective specification ϕ_i.

Due to the stochastic nature of the given model, model simulations with the same parameter values show varying behaviors which results in varying TeLEx tightness metrics. Therefore a multiple hypothesis testing technique, proposed by Bartroff et al. [6], is employed that generates several model simulations before deciding model satisfiability. The process of gen-

erating multiple simulations results in forming a separate distribution of TeLEx tightness metrics against each specification. Hence, k distributions are generated where each distribution corresponds to the tightness metrics computed by TeLEx against each of the k STL specifications.

We define the mean of each distribution i as $\theta^{(i)}$. In order to decide whether the given model satisfies a specification, we aim to test the hypothesis that the mean $\theta^{(i)}$ of each distribution is less than a threshold value $\theta'^{(i)}$ [51, Section 5.4]. If the mean $\theta^{(i)}$ of distribution i is greater than its corresponding threshold $\theta'^{(i)}$, the model satisfies ϕ_i since a greater mean of TeLEx tightness metrics indicates better model satisfiability. However if $\theta^{(i)} < \theta'^{(i)}$, it is considered that the model does not satisfy the corresponding specification ϕ_i.

Instead of testing the hypothesis against a single value $\theta'^{(i)}$, the problem is relaxed by introducing two thresholds $\theta_0^{(i)}$ and $\theta_1^{(i)}$ such that $\theta_0^{(i)} < \theta'^{(i)} < \theta_1^{(i)}$. We vary the two thresholds using a common factor, δ i.e. $\theta_0^{(i)} = \theta'^{(i)} - \delta$ and $\theta_1^{(i)} = \theta'^{(i)} + \delta$ where $0 < \delta < 1$ [54]. The region $(\theta_0^{(i)}, \theta_1^{(i)})$ is defined as the *indifference region*. As the indifference region becomes smaller, we reduce the risk of making a wrong decision but this comes at the cost of drawing more samples in order for the hypothesis test to make a decision. Hence for each distribution i, we test the following hypothesis:

$$\begin{aligned} H_0^{(i)} &: \theta^{(i)} = \theta_0^{(i)} \\ H_1^{(i)} &: \theta^{(i)} = \theta_1^{(i)} \end{aligned} \tag{4.1}$$

with type I (false negative) and type II (false positive) family-wise error rate (FWER) bounds as α and β respectively. Suppose $x_j^{(i)}$ is the tightness metric returned by TeLEx after verifying j^{th} simulation trace against i^{th} STL specification ϕ_i. Then for the n^{th} model simulation, the hypothesis test calculates the following log-likelihood ratio against each specification ϕ_i:

$$Z_i = \log \frac{e^{-(1/2\sigma_i^2)\sum_{j=1}^{n}(x_j^{(i)}-\theta_1^{(i)})^2}}{e^{-(1/2\sigma_i^2)\sum_{j=1}^{n}(x_j^{(i)}-\theta_0^{(i)})^2}} \tag{4.2}$$

The test samples sequentially until $Z_i \leq A_i$ or $Z_i \geq B_i$ where A_i and B_i are the stopping boundaries and are defined as functions of FWER bounds α and β. For each distribution i, the stopping boundaries are defined as:

$$A_i = log\left(\frac{\beta}{(1-\alpha_i)(k-i+1)}\right), \quad B_i = log\left(\frac{(1-\beta_i)(k-i+1)}{\alpha}\right) \tag{4.3}$$

where,

$$\alpha_i = \frac{(k-i+1-\beta)\alpha}{(k-i+1)(k-\beta)}, \quad \beta_i = \frac{(k-i+1-\alpha)\beta}{(k-i+1)(k-\alpha)} \tag{4.4}$$

Note that the hypothesis test continues on taking samples until each Z_i crosses one of its corresponding stopping boundaries A_i or B_i.

Rejection of the null hypothesis $H_0^{(i)}$ indicates that the mean of the distribution i is greater than threshold $\theta'^{(i)}$ and model satisfies the corresponding behavioral specification ϕ_i whereas, acceptance of a null hypothesis $H_0^{(i)}$ means that the model with its current set of parameter values is not able to satisfy ϕ_i. In case all $H_0^{(i)}$ are rejected, the system further checks whether each specification ϕ_i is satisfied with probability at least r_i. If yes, it returns the current set of parameter values as the solution otherwise, it explores other parameter values using our search algorithm. We describe the behavior of the search algorithm in the following subsection.

4.2.2 Parameter search algorithm

To explore the parameter space of the given model, we employ the same search algorithm simulated annealing from the previous chapter (see Section 3.3.2). As mentioned earlier, simulated annealing [32, 1] is a probabilistic technique for approximating the global optimum of

a given function. The algorithm (refer to Algorithm 2) starts the search at a high temperature value, t_i, provided as input. It then checks model satisfiability with a random set of initial parameter values, ρ_{init}, against given specifications using our multiple hypothesis test (MHT) based statistical model checking approach (described in the previous subsection). ρ_{init} is returned as the estimated set of parameter values in case the hypothesis test returns "accept". If the hypothesis test rejects ρ_{init}, the algorithm checks model satisfiability with a randomly generated neighbouring set of parameter values, ρ_{new}. Every time a set of parameter values is rejected, the algorithm starts a new iteration by finding a new neighboring set and repeating the model checking step. The objective function that drives the search algorithm is defined as the *mean of unsuccessful simulations* exhibiting negative TeLEx tightness metrics. For each new iteration, mean (μ_{new}) of the current iteration is compared with mean (μ_{old}) of the previous iteration and the set of parameter values that is maximizing the mean function is accepted. However, in order to avoid local optima the algorithm also sometimes, with a very small probability, accepts bad solutions. The search process continues until it finds a required set of parameter values or the given temperature cools down to a predefined value, t_f.

4.3 Experimentation

We study two rule-based receptor models FcϵRI and T-cell for experimental purposes. We show that the presented approach is successful in estimating 26 parameters of a FcϵRI model and 29 parameters of a T-cell model satisfying three probabilistic STL properties each.

Algorithm 2 Parameter Estimation Algorithm Against Multiple Specifications

Require:

$\mathcal{M}(\rho)$	Parameterized stochastic model
$\{\phi_1, \phi_2, ..., \phi_k\}$	STL specifications
$\{r_1, r_2, ..., r_k\}$	required probabilities
t_i	initial temperature
t_f	final temperature
c	cooling rate

Ensure:

ρ_0 such that $\mathcal{M}(\rho_0) \vDash P_{\geq r_1}(\phi_1) \wedge P_{\geq r_2}(\phi_2) \wedge .... \wedge P_{\geq r_k}(\phi_k)$

```
t ← t_i
ρ_init ← rand( )
result, μ_init ← MHT(M(ρ_init), {φ_1, .., φ_k}, {r_1, .., r_k})
if result = "acc" then return ρ_init end if
ρ_new ← ρ_init
μ_new ← μ_init
while t ⩾ t_f do
    ρ_old ← ρ_new
    μ_old ← μ_new
    ρ_new ← FindANeighbour(ρ_old)
    result, μ_new ← MHT(M(ρ_new), {φ_1, .., φ_k}, {r_1, .., r_k})
    if result = "acc" then return ρ_new
    else
        if μ_new ⩽ μ_old then
            if rand(0, 1) > e^(−(μ_new − μ_old)/t) then
                ρ_new ← ρ_old
            end if
        end if
    end if
    t ← c ∗ t
end while
return "No parameter set found!"
```

4.3.1 Benchmark 1: FcϵRI

FcϵRI is the high affinity receptor for immunoglobulin E (IgE) and is a member of the multichain immune recognition receptors (MIRR) family. It is responsible for controlling the activation of two different types of white blood cells known as human mast cells and basophils

which are a crucial part of the immune and neuroimmune system in living organisms. To be more specific, FcϵRI plays an important role in wound healing and immediate allergic reactions. It also participates in antigen presentation of immediate hypersensitivity reactions involving IgE-mediated release of histamine and some other mediators. Some of these allergic reactions can result in serious consequences and FcϵRI is of vital importance [50] in providing physiological protection against such reactions and maintaining the allergic response by controlling the secretion of allergic mediators and induction of cytokine gene transcription.

We perform our experiments on a rule-based model of FcϵRI developed by Faeder et al. [18] which aims to examine the function of multiple components in the phosphorylation and activation of Spleen tyrosine kinase, Syk, whose inhibition aids in treating autoimmune diseases. The authors of [18] model FcϵRI in its tetrameric form, which includes an α-chain that binds IgE and three Immunoreceptor Tyrosine-based Activation Motif (ITAM)-containing subunits, a β-chain, and two disulfide-linked γ-chains. The model exhibits four major reactions namely association, dissociation, phosphorylation, and dephosphorylation. All experiments in [18] are performed to stimulate and observe rat basophilic leukemia (RBL) cells using covalently cross-linked IgE dimers.

We translate three important model behaviors described in the results section of [18] to the following probabilistic Signal Temporal Logic behavioral properties:

Property 1:

$$P_{\geq 0.95}(\mathbf{F}_{[0,1500]}((RecDim/RecTot > 0.5) \wedge \mathbf{G}_{[1500,3000]}(RecDim/RecTot \geqslant 0.5)))$$

The first property investigates the kinetics of receptor aggregation and tyrosine phosphorylation in RBL cells when they are stimulated with covalently linked dimers of IgE. This property monitors one of the features of dimer-induced receptor (RecDim) phosphorylation

where the percentage of RecDim reaches half of its maximum value at around half an hour of the simulation and continues on increasing till one hour. In other words, this property verifies if the percentage of RecDim is observed to persist in the later half of the simulation.

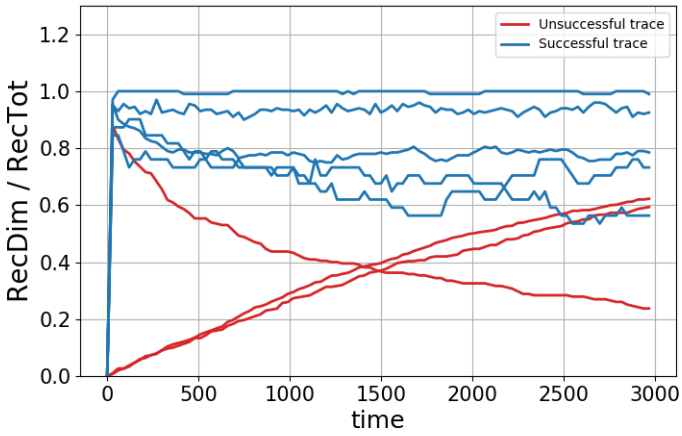

Figure 4.2: Successful BioNetGen simulation trajectories (in blue) of FcϵRI model satisfying Property 1. Unsuccessful trajectories are shown in red.

A visual representation of a series of BioNetGen traces successfully validating this property is provided in Figure 4.2. For comparison purposes, this figure also shows some of the unsuccessful traces obtained during the model verification phase.

Property 2:

$$P_{\geq 0.80}(\mathbf{F}_{[0,1500]}((LynRecPbeta/LynTot \geqslant 0.8) \wedge \mathbf{G}_{[1500,3000]}(LynRecPbeta/LynTot \geqslant 0.8)))$$

Simulating the FcϵRI model shows a rapid redistribution of Tyrosine-protein kinase Lyn. This redistribution becomes even more tightly associated with the receptor via binding of Lyn's SH2 domain to the phosphorylated β ITAM (LynRecPbeta). The second property verifies if a large percentage (say 80%) of the available Lyn is bound through its SH2 domain to β when the receptor aggregation reaches its maximum.

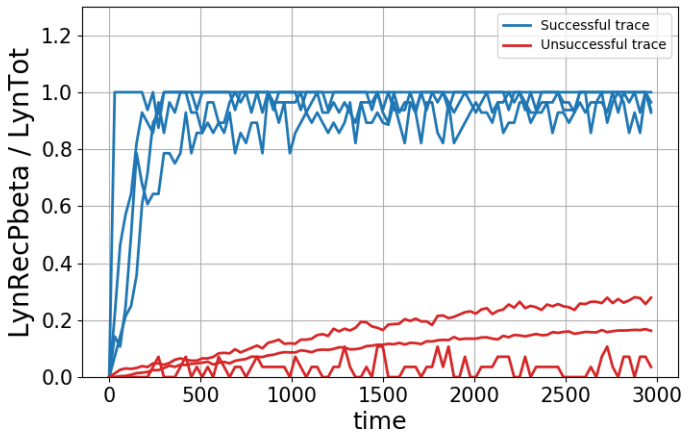

Figure 4.3: Successful BioNetGen simulation trajectories (in blue) of FcϵRI model satisfying Property 2. Unsuccessful trajectories are shown in red.

The traces shown in Figure 4.3 clearly represent the behavior described in this property. It can be observed that the concentration of LynRecPbeta reaches 80% of the total Lyn value at least once during the first half of the simulation (0-1500 seconds) and constantly maintains that percentage ($\geqslant$ 80%) during the second half of the simulation (1500-3000 seconds).

Property 3: $P_{\geq 0.99}(\mathbf{G}_{[1200,3000]}(RecPgamma/RecPbeta \geqslant 2.0))$

Another observation made by the authors in [18] is that Syk, which binds to the γ ITAM, is present in much higher concentration than Lyn, which binds to the β ITAM. This happens because a phosphorylated γ ITAM tyrosine has a longer lifetime than a β phosphotyrosine. Our third property verifies this behavior by showing that the level of γ phosphorylation (RecPgamma) exceeds that of β phosphorylation (RecPbeta) by $\sim$2-fold after simulating the model for some time and persists this behavior afterwards.

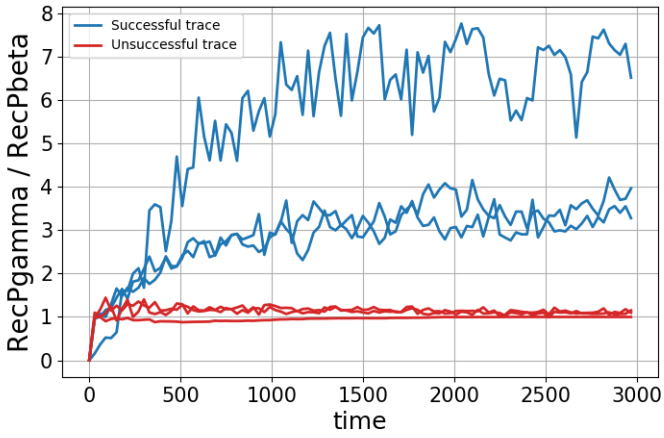

Figure 4.4: Successful BioNetGen simulation trajectories (in blue) of FcϵRI model satisfying Property 3. Unsuccessful trajectories are shown in red.

Figure 4.4 illustrates the successful as well as unsuccessful traces obtained from BioNetGen verifying the above STL property of FcϵRI model. It shows how the concentration of RecPgamma is persistently greater (by a factor of 2) than that of RecPbeta during the later part of the simulation.

Our experimental results estimating 26 parameters of FcϵRI model are shown in Table 4.1. We record the time taken by our method to find a single set of parameter values for the model against different combinations of three probabilistic STL specifications. We run Algorithm 2 for 20 times and each run starts the search from a random set of parameter values. We then record the number of successful runs generating the set of estimated parameter values and the average time taken by the algorithm over successful runs.

Table 4.2 shows the estimated parameter values of FcϵRI satisfying all three specifications obtained by our algorithm when $\alpha = 0.001$, $\beta = 0.001$, $\delta = 0.01$ and $\boldsymbol{r} = \{0.95, 0.80, 0.99\}$.

Table 4.1: Performance of our method to estimate parameters of FcϵRI model against three specifications with required probability $\boldsymbol{r} = \{0.95, 0.80, 0.99\}$, $\alpha = 0.005$, $\beta = 0.2$

Properties	Total # of runs	# of succ runs	Avg. time over succ runs (hrs)
Property 1	20	10	0.2005
Property 2	20	7	0.6608
Property 3	20	1	2.0171
Property 1,2	20	4	0.3137
Property 2,3	20	1	9.2775
Property 1,3	20	3	2.0816
Property 1,2,3	20	3	1.9162

As mentioned before there are two types of parameters in rule-based models: initial concentration of species and rate constants. The first four parameters in Table 4.2 represent the molecules present in FcϵRI model and the values represent their initial concentration at the start of each simulation. Parameters 5-26 are the rate constants which measure the rate at which a certain reaction rule proceeds. For instance, parameter 17 (pLb) and 19 (pLg) act as the rate constants for receptor transphosphorylation of β and γ ITAMs respectively by constitutive Lyn. The result of this transphosphorylation is being observed in Property 3 of FcϵRI model. Each of the rate constants is able to regulate one or more reactions. For example, parameter 25 (dm) serves as a rate constant for two reactions i.e. receptor dephosphorylation of β as well as γ ITAMs.

4.3.2 Benchmark 2: T-cell

Our second benchmark is a rule-based model of T-cell receptor, also known as T-lymphocyte. It is a type of white blood cell that is a vital part of the immune system. It detects the presence of toxins or other foreign substances, known as antigens, with the help of T-cell receptors (TCRs). More details related to T-cell receptor can be found in Section 3.4.1. In our experiments, we observe the model's primary output i.e. the fraction of doubly

Table 4.2: Estimated set of parameter values obtained by our approach for FcϵRI satisfying three specifications with $\boldsymbol{r} = \{0.95, 0.80, 0.99\}$, $\alpha = 0.005$, $\beta = 0.2$

Param No.	Param Name	Param Value
1	Lig_tot	84097.58
2	Rec_tot	203.843
3	Lyn_tot	23.868
4	Syk_tot	40001.02
5	kp1	3.58e-07
6	km1	0.000493
7	kp2	22.514
8	km2	0.00177
9	kpL	3.927
10	kmL	13002.84
11	kpLs	2.705
12	kmLs	0.0231
13	kpS	0.000568
14	kmS	0.250
15	kpSs	2.869
16	kmSs	0.00602
17	pLb	247.822
18	pLbs	10005.41
19	pLg	0.00157
20	pLgs	1197.121
21	pLS	0.247
22	pLSs	4.713
23	pSS	10226.75
24	pSSs	421.061
25	dm	0.0599
26	dc	2000.160

phosphorylated ERK (ppERK). This fraction (ppERK/totERK) is also taken as a measure of T-cell activation. If $ppERK/totERK < 0.10$, the cell is considered to be inactive and if $ppERK/totERK > 0.50$, the T-cell is considered to be in its active state. All three of our properties closely observe the value of this fraction and determine if the cell successfully achieves an active state after exhibiting an inactive state (and vice versa) within a defined period of time. We estimate the parameter values of the model such that it satisfies the following three properties:

Property 1: $P_{\geq 0.85}(\mathbf{G}_{[0,300]}(ppERK/totERK < 0.95))$

As mentioned earlier, cell activity is measured by changing the quantity of ppERK. The first property verifies if the fraction of ppERK always stays below a given threshold value (0.95) during the first 300 time units of the simulation.

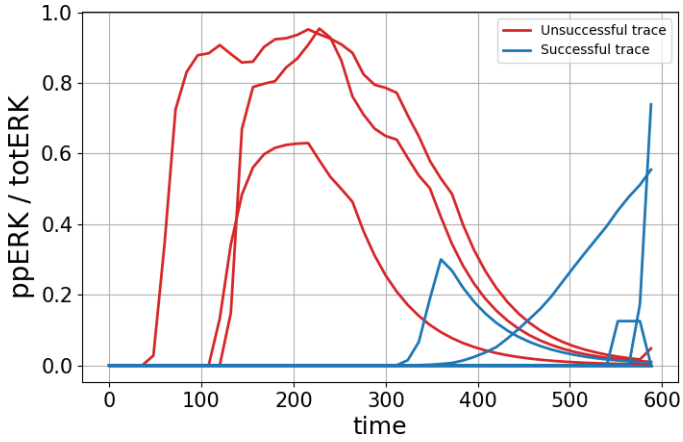

Figure 4.5: Successful BioNetGen simulation trajectories (in blue) of T-cell model satisfying Property 1. Unsuccessful trajectories are shown in red.

A visual representation of a series of BioNetGen traces successfully validating this property is provided in Figure 4.5. For comparison purposes, this figure also shows some of the unsuccessful traces obtained during the model verification phase.

Property 2:

$$P_{\geq 0.80}(\mathbf{F}_{[0,300]}((ppERK/totERK < 0.1) \wedge \mathbf{F}_{[300,600]}(ppERK/totERK > 0.5)))$$

Model simulations have shown that a small number of agonist peptides are sufficient for cell activation. Our second property verifies this behavior by determining if T-cell is able to

achieve the active state in the later half of its simulation provided that it was inactive at some point in time during the first half.

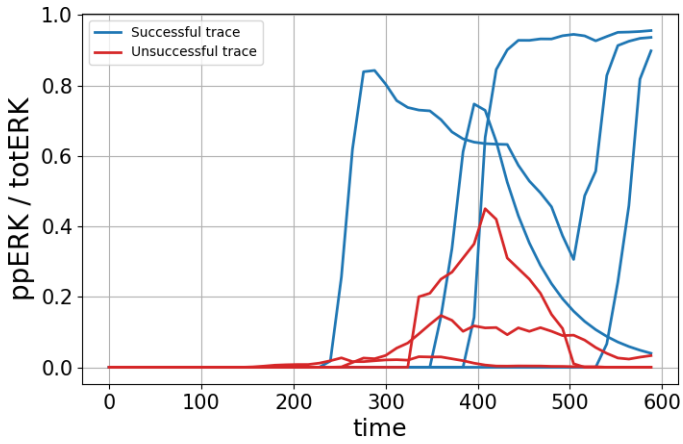

Figure 4.6: Successful BioNetGen simulation trajectories (in blue) of T-cell model satisfying Property 2. Unsuccessful trajectories are shown in red.

The traces shown in Figure 4.6 clearly show the cell activation behavior described by the above property.

Property 3:

$$P_{\geq 0.70}(\mathbf{F}_{[0,1000]}((ppERK/totERK > 0.5) \wedge \mathbf{F}_{[1000,2000]}(ppERK/totERK < 0.1)))$$

It is known that cell activation achieved by agonist peptides (Property 2) can be almost completely inhibited by antagonist peptides [38]. The third property monitors the deactivation behavior of T-cells which is a result of pSHP mediated negative feedback. This property verifies if the system can go from an active state during first 300 seconds of the simulation to an inactive state during the next 300 seconds.

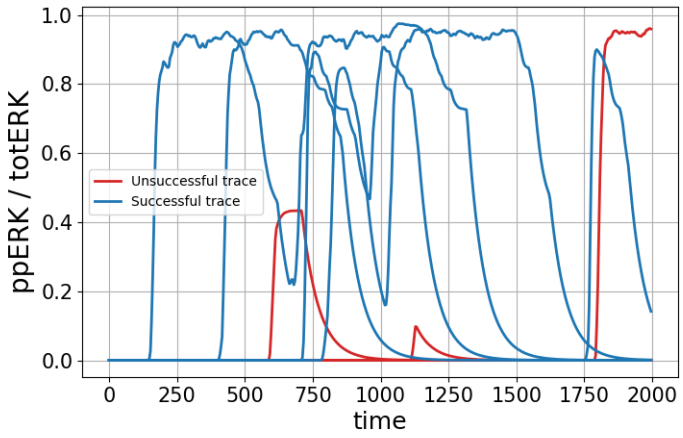

Figure 4.7: Successful BioNetGen simulation trajectories (in blue) of T-cell model satisfying Property 3. Unsuccessful trajectories are shown in red.

Figure 4.7 illustrates the successful as well as unsuccessful traces of T-cell model obtained from BioNetGen against the above STL property. It can be observed from this figure that the traces in blue correctly represents the deactivation behavior of a T-cell receptor. Our experimental results to estimate 29 parameters of a rule-based T-cell receptor model against three STL properties with their respective required probabilities are shown in Table 4.3.

Table 4.3: Performance of our method to estimate parameters of T-cell model against three specifications with required probability $\boldsymbol{r} = \{0.85, 0.80, 0.70\}$, $\alpha = 0.005$, $\beta = 0.2$

Properties	Total # of runs	# of succ runs	Avg. time over succ runs (hrs)
Property 1	20	13	0.2851
Property 2	20	9	0.1042
Property 3	20	1	3.9671
Property 1,2	20	3	0.3913
Property 2,3	40	2	15.229
Property 1,3	40	4	8.2544
Property 1,2,3	40	2	13.112

Table 4.4 shows the estimated parameter values of the same model satisfying all three specifications obtained by our approach when $\alpha = 0.001$, $\beta = 0.001$, $\delta = 0.01$ and $\boldsymbol{r} = \{0.85, 0.80, 0.70\}$. In Table 4.4, the first eight parameters represent the molecules present in T-cell model with their respective initial concentrations. Parameters 9-29 are the rate constants which quantify the rate of reaction rules. All three properties validated for this model involve the monitoring of doubly phosphorylated ERK (ppERK). The rate constants involved in the phosphorylation and dephosphorylation of ERK are parameter 28 (e1) and 29 (e2) respectively.

Table 4.4: Estimated set of parameter values obtained by our approach for T-cell model satisfying three specifications with $\boldsymbol{r} = \{0.85, 0.80, 0.70\}$, $\alpha = 0.005$, $\beta = 0.2$

Param No.	Param Name	Param Value
1	pMHC(p∼ag)	10000.17
2	pMHC(p∼en)	98882.07
3	TCR	200.541
4	LCK	6928.496
5	ZAP	10168.30
6	MEK	27998888.31
7	ERK	348.856
8	SHP	54303826.56
9	b1	0.00155
10	b2	0.000631
11	d1	25.074
12	d2	33.137
13	lb	0.000291
14	ly1	0.00166
15	ly2	27.0005
16	ls1	0.0840
17	ls2	0.000178
18	tp	17.234
19	s0	0.000243
20	s1	0.0100
21	s2	0.000693
22	s3	5.434
23	z0	0.0002
24	z1	0.0104
25	z2	3.976
26	m1	0.00510
27	m2	1.800
28	e1	0.00661
29	e2	2.0035

CHAPTER 5: CONCLUSION

In this study, we have proposed a new method for estimating the unknown parameters of a stochastic model such that the model satisfies multiple probabilistic temporal logic behavioral specifications simultaneously. The first step of our proposed method computes a quantitative tightness metric describing how well a model satisfies a given probabilistic specification and later uses that metric in a larger mean hypothesis test based statistical model checking technique combined with a simulated annealing search to estimate all the unknown parameters of that model.

The second step of our method employs the same quantitative tightness metric to validate the given stochastic model, using a multiple hypothesis testing based statistical model checking approach combined with simulated annealing search, in order to find a single set of parameter values for the model so that the model satisfies all the given probabilistic temporal logic behavioral specifications simultaneously. Our experimental results demonstrate that our method was successful in estimating 29 and 26 parameters of two stochastic rule-based biochemical models FcϵRI and T-cell respectively against three probabilistic temporal logic behavioral specifications each.

5.1 Future Work

We plan to pursue multiple directions for future work. Since our current method requires several sequential model simulations to be generated in order to check model satisfiability; therefore, we plan to take advantage of the latest parallel computing frameworks and execute these simulations in parallel by implementing the presented algorithm on GPUs. The curse

of dimensionality is one of the biggest challenges faced when designing solutions to such problems. To address this, we intend to investigate dimensionality reduction techniques [28] which would help the search process to perform more efficiently and generate useful results faster.

We are also interested in exploring and designing solutions based on deep learning techniques that can improve the search process. Another interesting future direction is to construct a neural network that is able to learn the given stochastic model from its time series simulation traces. Later this neural network could be used to generate hundreds of simulation traces in parallel on GPUs eliminating the need to actually simulate the model. This could eventually help expedite the parameter estimation process.

REFERENCES

[1] E. Aarts, J. Korst, and W. Michiels. Simulated annealing. In *Search methodologies*, pages 187–210. Springer, 2005.

[2] L. J. Allen. *An introduction to stochastic processes with applications to biology*. Chapman and Hall/CRC, 2010.

[3] M. Antoniotti, A. Policriti, N. Ugel, and B. Mishra. Model building and model checking for biochemical processes. *Cell biochemistry and biophysics*, 38(3):271, 2003.

[4] C. Baier and J.-P. Katoen. *Principles of model checking*. MIT press, 2008.

[5] E. Bartocci, L. Bortolussi, L. Nenzi, and G. Sanguinetti. On the robustness of temporal properties for stochastic models. *arXiv preprint arXiv:1309.0866*, 2013.

[6] J. Bartroff and J. Song. Sequential tests of multiple hypotheses controlling type I and II familywise error rates. *Journal of statistical planning and inference*, 153:100–114, 2014.

[7] M. L. Blinov, J. R. Faeder, and W. S. Hlavacek. Rule-based modeling of biological systems using BioNetGen modeling language.

[8] M. L. Blinov, J. R. Faeder, B. Goldstein, and W. S. Hlavacek. BioNetGen: software for rule-based modeling of signal transduction based on the interactions of molecular domains. *Bioinformatics*, 20(17):3289–3291, 2004.

[9] L. Brim, M. Češka, S. Dražan, and D. Šafránek. Exploring parameter space of stochastic biochemical systems using quantitative model checking. In *Proceedings of the 25th international conference on Computer Aided Verification*, pages 107–123. Springer-Verlag, 2013.

[10] L. Calzone, N. Chabrier-Rivier, F. Fages, and S. Soliman. Machine learning biochemical networks from temporal logic properties. *Lecture Notes in Computer Science*, 4220:68–94, 2006.

[11] M. Češka, P. Pilař, N. Paoletti, L. Brim, and M. Kwiatkowska. PRISM-PSY: precise GPU-accelerated parameter synthesis for stochastic systems. In *International Conference on Tools and Algorithms for the Construction and Analysis of Systems*, pages 367–384. Springer, 2016.

[12] M. Češka, F. Dannenberg, N. Paoletti, M. Kwiatkowska, and L. Brim. Precise parameter synthesis for stochastic biochemical systems. *Acta Informatica*, 54(6):589–623, 2017.

[13] A. Cimatti, E. Clarke, E. Giunchiglia, F. Giunchiglia, M. Pistore, M. Roveri, R. Sebastiani, and A. Tacchella. NuSMV 2: An opensource tool for symbolic model checking. In *International Conference on Computer Aided Verification*, pages 359–364. Springer, 2002.

[14] F. Ciocchetta, A. Duguid, S. Gilmore, M. L. Guerriero, and J. Hillston. The Bio-PEPA tool suite. In *Quantitative Evaluation of Systems, 2009. QEST'09. Sixth International Conference on the*, pages 309–310. IEEE, 2009.

[15] S. Ditlevsen and A. Samson. Introduction to stochastic models in biology. In *Stochastic biomathematical models*, pages 3–35. Springer, 2013.

[16] A. Donzé, T. Ferrere, and O. Maler. Efficient robust monitoring for STL. In *International Conference on Computer Aided Verification*, pages 264–279. Springer, 2013.

[17] E. A. Emerson et al. Temporal and modal logic. *Handbook of Theoretical Computer Science, Volume B: Formal Models and Sematics (B)*, 995(1072):5, 1990.

[18] J. R. Faeder, W. S. Hlavacek, I. Reischl, M. L. Blinov, H. Metzger, A. Redondo, C. Wofsy, and B. Goldstein. Investigation of early events in FcεRI-mediated signaling using a detailed mathematical model. *The Journal of Immunology*, 170(7):3769–3781, 2003.

[19] J. R. Faeder, M. L. Blinov, and W. S. Hlavacek. Rule-based modeling of biochemical systems with BioNetGen. *Systems biology*, pages 113–167, 2009.

[20] D. T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *The journal of physical chemistry*, 81(25):2340–2361, 1977.

[21] D. T. Gillespie, M. Roh, and L. R. Petzold. Refining the weighted stochastic simulation algorithm. *The Journal of chemical physics*, 130(17):174103, 2009.

[22] L. S. Hayre and J. C. Gittins. Sequential selection of the larger of two normal means. *Journal of the American Statistical Association*, 76(375):696–700, 1981.

[23] S. Hoops, S. Sahle, R. Gauges, C. Lee, J. Pahle, N. Simus, M. Singhal, L. Xu, P. Mendes, and U. Kummer. COPASI–a COmplex PAthway SImulator. *Bioinformatics*, 22(24): 3067–3074, 2006.

[24] M. Hucka, A. Finney, H. M. Sauro, H. Bolouri, J. C. Doyle, H. Kitano, A. P. Arkin, B. J. Bornstein, D. Bray, A. Cornish-Bowden, et al. The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. *Bioinformatics*, 19(4):524–531, 2003.

[25] F. Hussain, S. K. Jha, S. Jha, and C. J. Langmead. Parameter discovery in stochastic biological models using simulated annealing and statistical model checking. *International Journal of Bioinformatics Research and Applications 2*, 10(4-5):519–539, 2014.

[26] F. Hussain, C. J. Langmead, Q. Mi, J. Dutta-Moscato, Y. Vodovotz, and S. K. Jha. Automated parameter estimation for biological models using Bayesian statistical model checking. *BMC bioinformatics*, 16(17):S8, 2015.

[27] S. Jha, A. Tiwari, S. A. Seshia, T. Sahai, and N. Shankar. TeLEx: Passive STL learning using only positive examples. In *International Conference on Runtime Verification*, pages 208–224. Springer, 2017.

[28] S. K. Jha and C. J. Langmead. Stochastic computational model parameter synthesis system, Jan. 31 2017. US Patent 9,558,300.

[29] 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*, volume 5688, pages 218–234. Springer, 2009.

[30] A. Khalid and S. K. Jha. Calibration of rule-based stochastic biochemical models using statistical model checking. In *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 179–184. IEEE, 2018. doi: 10.1109/BIBM.2018.8621292.

[31] A. Khalid and S. K. Jha. Parameter estimation of stochastic biochemical models using multiple hypothesis testing. In *2018 IEEE 8th International Conference on Computational Advances in Bio and Medical Sciences (ICCABS)*, pages 1–1. IEEE, 2018. doi: 10.1109/ICCABS.2018.8541912.

[32] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *science*, 220(4598):671–680, 1983.

[33] H. Kitano. Systems biology: a brief overview. *Science*, 295(5560):1662–1664, 2002.

[34] M. Kwiatkowska, G. Norman, and D. Parker. Stochastic model checking. In *International School on Formal Methods for the Design of Computer, Communication and Software Systems*, pages 220–270. Springer, 2007.

[35] M. Kwiatkowska, G. Norman, and D. Parker. PRISM 4.0: Verification of probabilistic real-time systems. In G. Gopalakrishnan and S. Qadeer, editors, *Proc. 23rd International Conference on Computer Aided Verification (CAV'11)*, volume 6806 of *LNCS*, pages 585–591. Springer, 2011.

[36] A. Legay, B. Delahaye, and S. Bensalem. Statistical model checking: An overview. In *International Conference on Runtime Verification*, pages 122–135. Springer, 2010.

[37] J. Liepe, P. Kirk, S. Filippi, T. Toni, C. P. Barnes, and M. P. Stumpf. A framework for parameter estimation and model selection from experimental data in systems biology using approximate Bayesian computation. *Nature protocols*, 9(2):439–456, 2014.

[38] T. Lipniacki, B. Hat, J. R. Faeder, and W. S. Hlavacek. Stochastic effects and bistability in T cell receptor signaling. *Journal of theoretical Biology*, 254(1):110–122, 2008.

[39] B. Liu and J. R. Faeder. Parameter estimation of rule-based models using statistical model checking. In *Bioinformatics and Biomedicine (BIBM), 2016 IEEE International Conference on*, pages 1453–1459. IEEE, 2016.

[40] C. Madsen, F. Shmarov, and P. Zuliani. BioPSy: an SMT-based tool for guaranteed parameter set synthesis of biological models. In *International Conference on Computational Methods in Systems Biology*, pages 182–194. Springer, 2015.

[41] T. Mancini, E. Tronci, I. Salvo, F. Mari, A. Massini, and I. Melatti. Computing biological model parameters by parallel statistical model checking. In *International Conference on Bioinformatics and Biomedical Engineering*, pages 542–554. Springer, 2015.

[42] M. Nikulin. Statistical hypotheses, encyclopedia of mathematics. URL `http://www.encyclopediaofmath.org/index.php?title=Statistical_hypotheses,_verification_of&oldid=12498`.

[43] 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 *International Conference on Computational Methods in Systems Biology*, pages 120–134. Springer, 2013.

[44] V. Raman, A. Donzé, D. Sadigh, R. M. Murray, and S. A. Seshia. Reactive synthesis from signal temporal logic specifications. In *Proceedings of the 18th International Conference on Hybrid Systems: Computation and Control*, pages 239–248. ACM, 2015.

[45] R. Ramanathan, Y. Zhang, J. Zhou, B. M. Gyori, W.-F. Wong, and P. Thiagarajan. Parallelized parameter estimation of biological pathway models. In *HSB*, pages 37–57, 2015.

[46] 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 *International Conference on Computational Methods in Systems Biology*, pages 251–268. Springer, 2008.

[47] H. Schmidt and M. Jirstrand. Systems Biology Toolbox for MATLAB: a computational platform for research in systems biology. *Bioinformatics*, 22(4):514–515, 2005.

[48] K. Sen, M. Viswanathan, and G. Agha. On statistical model checking of stochastic systems. In *International Conference on Computer Aided Verification*, pages 266–280. Springer, 2005.

[49] K. Sen, M. Viswanathan, and G. Agha. Vesta: A statistical model-checker and analyzer for probabilistic systems. In *Quantitative Evaluation of Systems, 2005. Second International Conference on the*, pages 251–252. IEEE, 2005.

[50] H. Turner and J.-P. Kinet. Signalling through the high-affinity IgE receptor FcεRI. *Nature*, 402(6760supp):24, 1999.

[51] A. Wald. Sequential tests of statistical hypotheses. *The Annals of Mathematical Statistics*, 16(2):117–186, 1945.

[52] Q. Wang, N. Miskov-Zivanov, B. Liu, J. R. Faeder, M. Lotze, and E. M. Clarke. Formal modeling and analysis of pancreatic cancer microenvironment. In *International Conference on Computational Methods in Systems Biology*, pages 289–305. Springer, 2016.

[53] H. L. Younes and R. G. Simmons. Probabilistic verification of discrete event systems using acceptance sampling. In *International Conference on Computer Aided Verification*, pages 223–235. Springer, 2002.

[54] H. L. Younes and R. G. Simmons. Statistical probabilistic model checking with a focus on time-bounded properties. *Information and Computation*, 204(9):1368–1409, 2006.

[55] H. L. Younes, M. Kwiatkowska, G. Norman, and D. Parker. Numerical vs. statistical probabilistic model checking. *International Journal on Software Tools for Technology Transfer*, 8(3):216–228, 2006.

[56] J. Zhou, R. Ramanathan, W.-F. Wong, and P. Thiagarajan. Automated property synthesis of ODEs based bio-pathways models. In *International Conference on Computational Methods in Systems Biology*, pages 265–282. Springer, 2017.

[57] P. Zuliani. Statistical model checking for biological applications. *International Journal on Software Tools for Technology Transfer*, 17(4):527–536, 2015.