Precise Parameter Synthesis for Stochastic Biochemical Systems

Milan Češka[1,2], Frits Dannenberg[1], Marta Kwiatkowska[1], and Nicola Paoletti[1]

[1] Department of Computer Science, University of Oxford, UK
[2] Faculty of Informatics, Masaryk University, Czech Republic

Abstract. We consider the problem of synthesising rate parameters for stochastic biochemical networks so that a given CSL time-bounded property is guaranteed to hold, or, in the case of quantitative properties, the probability of satisfying the property is maximised/minimised. We develop algorithms based on the computation of lower and upper bounds of the probability, in conjunction with refinement and sampling, which yield answers that are precise to within an arbitrarily small tolerance value. Our methods are efficient and improve on existing approximate techniques that employ discretisation and refinement. We evaluate the usefulness of the methods by synthesising rates for two biologically motivated case studies, including the reliability analysis of a DNA walker.

1 Introduction

Biochemical reaction networks are a convenient formalism for modelling a multitude of biological systems, including molecular signalling pathways, logic gates built from DNA and DNA walker circuits. For low molecule counts, and under the well-mixed and fixed volume assumption, the prevailing approach is to model such networks using continuous-time Markov chains (CTMCs) [11]. Stochastic model checking [17], e.g. using `PRISM` [18], can then be employed to analyse the behaviour of the models against temporal logic properties expressed in CSL (Continuous Stochastic Logic) [2]. For example, one can establish the reliability and performance of DNA walker circuits by means of properties such as "what is the probability that the walker reaches the correct final anchorage within 10 min?". Since DNA circuits can implement biosensors and diagnostic systems, ensuring appropriate levels of reliability is crucial to guarantee the safety of deploying molecular devices in healthcare applications.

Stochastic model checking, however, assumes that the model is fully specified, including the kinetic rates. In view of experimental measurement error, these are rarely given precisely, but rather as intervals of values. The *parameter synthesis problem*, studied for CTMCs in [13], assumes a formula and a model whose rates are given as functions of model parameters, and aims to compute the parameter valuations that guarantee the satisfaction of the formula. This allows one, for example, to determine the ranges of parameter values for a given level of reliability and performance, which can provide important feedback to the designers of biosensors and similar molecular devices, and thus significantly extends the power of stochastic model checking.

In [13], the parameter synthesis problem was solved for CTMCs approximately, and only for probabilistic time-bounded reachability. In this paper, we address the parameter synthesis problem for stochastic biochemical reaction networks for the full time-bounded fragment of the (branching-time) logic CSL [2]. We formulate two variants: *threshold synthesis*, which inputs a CSL formula and a probability threshold and identifies the parameter valuations which meet the threshold, and *max synthesis*, where the maximum probability of satisfying the property and the maximizing set of parameter valuations are returned.

We develop efficient synthesis algorithms that yield answers with arbitrary precision. The algorithms exploit the recently published parameter exploration technique that computes safe approximations to the lower and upper bounds for the probability to satisfy a CSL property over a fixed parameter space [6]. In contrast to the exploration technique, our algorithms automatically derive the satisfying parameter regions through iterative decomposition of the parameter space based on refining the preliminary answer with additional decompositions up to a given problem-specific tolerance value. We also show that significant computational speed-up is achieved by enhancing the max synthesis algorithm by sampling the property at specific points in the parameter space. We demonstrate the usefulness of the method through two case studies: the SIR epidemic model [16], where we synthesize infection and recovery rates that maximize the probability of disease extinction, and the DNA walker circuit [9], where we derive the rates that ensure a predefined level of reliability.

Related work. Parameter synthesis has been studied for discrete-time Markovian models in [12, 7]. The approach applies to unbounded temporal properties and is based on constructing a rational function by performing state elimination [12]. For CTMCs and bounded reachability specifications, the problem can be reduced to the analysis of the polynomial function describing the reachability probability of a given target state [13]. The main limitation here is the high degree of the polynomials, which is determined by the number of uniformization steps. Therefore, in contrast to our method, only an approximate solution is obtained using discretization of parameter ranges. When considering linear-time specifications, specific restrictions can be placed on the rate function to result in a smooth satisfaction function (i.e. having derivatives of all orders). In that case, the function can be approximated using statistical methods which leverage the smoothness [5]. A concept similar to smoothness, uniform continuity, can be used to obtain an unbiased statistical estimator for the satisfaction function [14]. Both methods approximate parameter synthesis using confidence intervals. Inference of parameter values in probabilistic models from time-series measurements is a well studied area of research [1, 4], but different from the problem we consider. Interval CTMCs, where transition rates are given as intervals, have been employed to obtain a three-valued abstraction for CTMCs [15]. In contrast to parametric models we work with, the transition rates in interval CTMCs are chosen nondeterministically and schedulers are introduced to compute lower and upper probability bounds.

2 Background

We state preliminary definitions relevant to the study of Parametric Continuous Time Markov Chains [13, 6] that permit formal analysis of probabilistic models with uncertain parameters [20].

A *Continuous Time Markov Chain* (CTMC) is a tuple $\mathcal{C} = (S, \pi_0, \mathbf{R})$ where S is a finite set of states, $\pi_0 : S \to \mathbb{R}_{\geq 0}$ is the initial distribution and $\mathbf{R} : S \times S \to \mathbb{R}_{\geq 0}$ is the rate matrix. A transition between states $s, s' \in S$ can occur only if $\mathbf{R}(s, s') > 0$ and in that case the probability of triggering the transition within t time units equals $1 - e^{-t\mathbf{R}(s,s')}$. The time spent in s, before a reaction occurs, is exponentially distributed with rate $E(s) = \sum_{s' \in S} \mathbf{R}(s, s')$, and when the transition occurs the probability of moving to state s' is given by $\frac{\mathbf{R}(s,s')}{E(s)}$. Let $\mathbf{E}$ be a $S \times S$ diagonal matrix such that $\mathbf{E}(s_i, s_i) = E(s_i)$, and define the generating matrix by setting $\mathbf{Q} = \mathbf{R} - \mathbf{E}$. Then a vector $\pi_t : S \to \mathbb{R}_{\geq 0}$ of the transient probabilities at time t is given by $\frac{d\pi_t}{dt} = \pi_t \mathbf{Q}$ such that $\pi_t = \pi_0 e^{\mathbf{Q}t}$. Using standard uniformisation the transient probability at time t is obtained as a sum of state distributions after i discrete-stochastic steps, weighted by the probability of observing i steps in a Poisson process. Let $\mathbf{P} = \mathbf{I} + \frac{1}{q}\mathbf{Q}$ be the uniformised matrix, where $q \geq \max\{E(s) - \mathbf{R}(s, s) \mid s \in S\}$ is called the uniformisation rate. The transient probabilities π_t are computed as $\pi_t = \pi_0 \sum_{i=0}^{k_\epsilon} \gamma_{i,qt} \mathbf{P}^i$ where $\gamma_{i,qt} = e^{-qt} \frac{(qt)^i}{i!}$ denotes the i-th Poisson probability for a process with rate qt, and k_ϵ satisfies the convergence bound $\sum_0^{k_\epsilon} \gamma_{i,qt} \geq 1 - \epsilon$ for some $\epsilon > 0$. The Poisson terms and summation bound can be efficiently computed using an algorithm due to Fox and Glynn [10].

We assume a set K of model parameters. The domain of each parameter $k \in K$ is given by a closed real interval describing the range of possible values, i.e, $[k^\perp, k^\top]$. The *parameter space* $\mathcal{P}$ induced by K is defined as the Cartesian product of the individual intervals: $\mathcal{P} = \bigtimes_{k \in K} [k^\perp, k^\top]$. A *parameter point* $p \in \mathcal{P}$ is an evaluation of each parameter k. Subsets of the parameter space are also referred to as *parameter regions* or *subspaces*. $\mathbb{R}[K]$ denotes the set of polynomials over the reals $\mathbb{R}$ with variables $k \in K$.

Parametric Continuous Time Markov Chains (pCTMCs) [13] extend the notion of CTMCs by allowing transition rates to depend on model parameters. Formally, a pCTMC over a set K of parameters is a triple $\mathcal{C} = (S, \pi_0, \mathbf{R})$ where s and π_0 are as above, and in this case $\mathbf{R} : S \times S \to \mathbb{R}[K]$ is the parametric rate matrix. Given a pCTMC $\mathcal{C}$ and a parameter space $\mathcal{P}$, we denote with $C_\mathcal{P}$ the (possibly uncountable) set $\{\mathcal{C}_p \mid p \in \mathcal{P}\}$ where $\mathcal{C}_p = (S, \pi, \mathbf{R}_p)$ is the instantiated CTMC obtained by replacing the parameters in $\mathbf{R}$ with their evaluation in p.

We consider the time-bounded fragment of CSL [2] to specify behavioural properties, with the following syntax. A state formula Φ is given as $\Phi ::= \mathsf{true} \mid a \mid \neg\Phi \mid \Phi \wedge \Phi \mid P_{\sim r}[\phi] \mid P_{=?}[\phi]$, where ϕ is a path formula whose syntax is $\phi ::= \mathsf{X}\ \Phi \mid \Phi\ \mathsf{U}^I\ \Phi$, a is an atomic proposition, $\sim \in \{<, \leq, \geq, >\}$, $r \in [0, 1]$ is a probability threshold and I is a bounded interval. Using $P_{=?}[\phi]$ we specify properties which evaluate to the probability that ϕ is satisfied. The synthesis methods presented in this paper can be directly adapted to the time-bounded fragment of CSL with the reward operator [17], but, for the sake of simplicity, here we present our methods only for the probabilistic operator P.

Let ϕ be a CSL path formula and $\mathcal{C}_{\mathcal{P}}$ be a pCTMC over a space $\mathcal{P}$. We denote with $\Lambda : \mathcal{P} \to [0, 1]$ the *satisfaction function* such that $\Lambda(p) = P_{=?}[\phi]$, that is, $\Lambda(p)$ is the probability of ϕ being satisfied over the CTMC $\mathcal{C}_p$. Note that the path formula ϕ may contain nested probabilistic operators, and therefore the satisfaction function is, in general, not continuous.

Biochemical reaction networks provide a convenient formalism for describing various biological processes as a system of well-mixed reactive species in a volume of fixed size. A CTMC semantics can be derived whose states hold the number of molecules for each species, and transitions correspond to reactions that consume and produce molecules. Bounds on species counts can be imposed to obtain a finite-state model. The rate matrix is defined as $\mathbf{R}(s_i, s_j) \overset{def}{=} \sum_{r \in \mathsf{reac}(s_i,s_j)} f_r(K, s_i)$ where $\mathsf{reac}(s_i, s_j)$ denotes all the reactions changing state s_i into s_j and f_r is the *stochastic rate function* of reaction r over parameters $k \in K$. In this paper, we assume *multivariate polynomial* rate functions that include, among others, mass-action kinetics where $k \in K$ are kinetic rate parameters.

3 Problem Definition

We consider pCTMC models of biochemical reaction networks that can be parametric in the rate constants and in the initial state. We introduce two parameter synthesis problems for this class of models: the *threshold synthesis* problem that, given a threshold $\sim r$ and a CSL path formula ϕ, asks for the parameter region where the probability of ϕ meets $\sim r$; and the *max synthesis* problem that determines the parameter region where the probability of the input formula attains its maximum, together with an estimation of that maximum. In the remainder of the paper, we omit the min synthesis problem that is defined and solved in a symmetric way to the max case.

In contrast to previous approaches that support only specific kinds of properties (e.g. reachability as in [13]), we consider the full time-bounded fragment of CSL with rewards, thus enabling generic and more expressive synthesis requirements. Moreover, the variants of the synthesis problem that we define correspond to qualitative and quantitative CSL formulas, which are of the form $P_{\geq r}[\phi]$ and $P_{=?}[\phi]$, respectively. Solutions to the threshold problem admit parameter points left undecided, while, in the max synthesis problem, the set of maximizing parameters is contained in the synthesis region. Our approach supports arbitrarily precise solutions through an input tolerance that limits the volume of the undecided region (in the threshold case) and of the synthesis region (in the max case). To the best of our knowledge, no other synthesis methods for CTMCs exist that provide guaranteed error bounds.

Problem 1 (Threshold Synthesis). Let $\mathcal{C}_{\mathcal{P}}$ be a pCTMC over a parameter space $\mathcal{P}$, $\Phi = P_{\geq r}[\phi]$ with $r \in [0, 1]$ be a CSL formula and $\varepsilon > 0$ a volume tolerance. The *threshold synthesis* problem is finding a partition $\{T, U, F\}$ of $\mathcal{P}$, such that:

1. $\forall p \in T.\ \Lambda(p) \geq r$; and
2. $\forall p \in F.\ \Lambda(p) < r$; and
3. $\mathrm{vol}(U)/\mathrm{vol}(\mathcal{P}) \leq \varepsilon$

where Λ is the satisfaction function of ϕ on $\mathcal{C}_{\mathcal{P}}$; and $\mathrm{vol}(A) = \int_A 1 d\mu$ is the volume of A.

Problem 2 (Max Synthesis). Let $\mathcal{C}_{\mathcal{P}}$ be a pCTMC over a parameter space $\mathcal{P}$, $\Phi = P_{=?}[\phi]$ be a CSL formula and $\epsilon > 0$ a probability tolerance. The *max synthesis* problem is finding a partition $\{T, F\}$ of $\mathcal{P}$ and probability bounds $\Lambda^{\perp}$, $\Lambda^{\top}$ such that:

1. $\Lambda^{\perp} - \Lambda^{\top} \leq \epsilon$;
2. $\forall p \in T.\ \Lambda^{\perp} \leq \Lambda(p) \leq \Lambda^{\top}$; and
3. $\exists p \in T.\ \forall p' \in F.\ \Lambda(p) > \Lambda(p')$.

where Λ is the satisfaction function of ϕ on $\mathcal{C}_{\mathcal{P}}$.

Note that we need to consider a probability tolerance to control the inaccuracy of the max probability, and in turn of region T. Indeed, constraining only the volume of T gives no guarantees on the precision of the maximizing region.

4 Computing Lower and Upper Probability Bounds

This section presents a generalization of the parameter exploration procedure originally introduced in [6]. The procedure takes a pCTMC $\mathcal{C}_{\mathcal{P}}$ and a CSL path formula ϕ, and provides safe under- and over-approximations for the minimal and maximal probability that $\mathcal{C}_{\mathcal{P}}$ satisfies ϕ, that is, lower and upper bounds $\Lambda_{\min}$ and $\Lambda_{\max}$ satisfying $\Lambda_{\min} \leq \min_{p\in\mathcal{P}} \Lambda(p)$ and $\Lambda_{\max} \geq \max_{p\in\mathcal{P}} \Lambda(p)$. The accuracy of these approximations is improved by partitioning the parameter space $\mathcal{P}$ into subspaces and re-computing the corresponding bounds, which forms the basis of the synthesis algorithms that we discus in the next section. For now we focus on obtaining approximations $\Lambda_{\min}, \Lambda_{\max}$ for a fixed parameter space $\mathcal{P}$. The model-checking problem for any time-bounded CSL formula reduces to the computation of transient probabilities [3], and a similar reduction is applicable to the computation of lower and upper bounds. Following [6], to correctly handle nested probabilistic operators, under- and over-approximations of the satisfying sets of states in the nested formula are computed.

We now re-state the transient probabilities as given by standard uniformisation and include the dependency on the model parameters in our notation, so that $\pi_{t,p} = \pi_0 \sum_{i=0}^{k_\epsilon} \gamma_{i,qt} \mathbf{P}_p^i = \sum_{i=0}^{k_\epsilon} \gamma_{i,qt} \tau_{i,p}$ where $\mathbf{P}_p$ is the uniformised rate matrix obtained from the rate matrix $\mathbf{R}_p$ and $\tau_{k,p} = \pi_0 \mathbf{P}_p^k$ is the probability evolution in the discretized process. Observe that, if some functions $\pi_i^{\min}$ and $\pi_i^{\max}$ can be obtained such that for any step i,

$$\tau_i^{\min} \leq \min_{p\in\mathcal{P}} \tau_{i,p} \text{ and } \tau_i^{\max} \geq \max_{p\in\mathcal{P}} \tau_{i,p} \quad (1)$$

then robust approximations $\pi_t^{\min} = \sum_{i=0}^{k_\epsilon} \gamma_{i,qt} \tau_i^{\min}$ and $\pi_t^{\max} = \sum_{i=0}^{k_\epsilon} \gamma_{i,qt} \tau_i^{\max}$ provide the bounds $\Lambda_{\min}$ and $\Lambda_{\max}$ that we seek. As usual, the vector ordering in Equation 1 holds element-wise. If we assume that some functions $f_k^{\min}, f_k^{\max}$ exist such that $f_k^{\min}(\tau_i^{\min}) = \tau_{i+k}^{\min}$ and $f_k^{\max}(\tau_i^{\max}) = \tau_{i+k}^{\max}$ then recursively the terms in Equation 1 for all i are obtained, given that the first k terms for $\tau_i^{\min}, \tau_i^{\max}$ are known. We now note that the functions

$$f_k^{\min}(\tau_i^{\min}) = \min_{p \in \mathcal{P}} \ \tau_i^{\min} \mathbf{P}_p^k \text{ and } f_k^{\max}(\tau_i^{\max}) = \max_{p \in \mathcal{P}} \ \tau_i^{\max} \mathbf{P}_p^k \qquad (2)$$

can be under- and over-approximated using analytical methods when the parametric rate matrix $\mathbf{R}_p$ employs low-degree multivariate polynomial expressions. Provided that $\mathbf{R}_p(s_i, s_j)$ is a polynomial of at most degree d over the parameter space, the degree of $\tau_{k,p}(s) = \pi_0 \mathbf{P}_p^k(s)$ is at most kd.

An analytical treatment for the case $k = 1$ and $d = 1$ is given in [6]. Here, we derive an effective method to obtain approximations using $k = 1$ for multivariate polynomials where each variable has degree at most 1; full details are included in [21]. More advanced methods can be used, provided that the under- and over-approximations for Equation 2 are sound. Note that the solution $\pi_{t,p}(s)$ itself can be expressed as a polynomial of degree at most $k_\epsilon d$. A direct attempt to bound the polynomial expression of $\pi_{t,p}(s)$ is difficult due to the large number of uniformisation steps, k_ϵ, and previous approaches in parameter synthesis have provided an approximate solution by sampling the value of $\pi_{t,p}$ over a grid in $\mathcal{P}$ [13], rather than bounding the polynomial itself as in our approach. The computational complexity depends on the chosen rate function and the bounding method for the functions in Equation 2, but for our settings it has the same asymptotic complexity as standard uniformisation. Two approximation errors are introduced when we compute $\pi_t^{\max}$ (or $\pi_t^{\min}$).

Firstly, the probabilities $\tau_i^{\max}, \tau_{i+k}^{\max}, \tau_{i+2k}^{\max}, \ldots$ are locally maximized, so that different parameter evaluations are allowed at each step and for each state. Secondly, the error of over-approximating $f_k^{\max}(\tau_i^{\max})$ accumulates in $\tau_i^{\max}$ at every iteration.

5 Refinement-based Parameter Synthesis

We present algorithms to solve Problems 1 and 2, based on the computation of probability bounds introduced in Section 4 and iterative parameter space refinement. In the max synthesis case we employ parameter sampling to enhance the synthesis procedure.

Threshold Synthesis. Algorithm 1 describes the method to solve the threshold synthesis problem with input formula $\Phi = P_{\geq r}[\phi]$. The idea, also illustrated in Figure 1, is to iteratively refine the undecided parameter subspace U (line 3) until the termination condition is met (line 14). At each step, we obtain a partition R of U. For each subspace $\mathcal{R} \in R$, the algorithm computes bounds $\Lambda_{\min}^{\mathcal{R}}$ and $\Lambda_{\max}^{\mathcal{R}}$ on the maximal and minimal probability that $\mathcal{C}_{\mathcal{R}}$ satisfies ϕ (line 5). We then evaluate if $\Lambda_{\min}^{\mathcal{R}}$ is above the threshold r, in which case the satisfaction

Algorithm 1 Threshold Synthesis

```
Require: pCTMC $\mathcal{C}_\mathcal{P}$ over parameter space $\mathcal{P}$, CSL formula
    $\Phi = P_{\geq r}[\phi]$ and volume tolerance $\varepsilon > 0$
Ensure: $T$, $U$ and $F$ as in Problem 1
 1: $T \leftarrow \emptyset$, $F \leftarrow \emptyset$, $U \leftarrow \mathcal{P}$
 2: repeat
 3:     $R \leftarrow \mathsf{decompose}(U)$, $U \leftarrow \emptyset$
 4:     for each $\mathcal{R} \in R$ do
 5:         $(\Lambda^{\mathcal{R}}_{\min}, \Lambda^{\mathcal{R}}_{\max}) \leftarrow \mathsf{computeBounds}(\mathcal{C}_\mathcal{R}, \phi)$
 6:         if $\Lambda^{\mathcal{R}}_{\min} \geq r$ then
 7:             $T \leftarrow T \cup \mathcal{R}$
 8:         else if $\Lambda^{\mathcal{R}}_{\max} < r$ then
 9:             $F \leftarrow F \cup \mathcal{R}$
10:         else
11:             $U \leftarrow U \cup \mathcal{R}$
12: until $\mathrm{vol}(U)/\mathrm{vol}(\mathcal{P}) \leq \varepsilon$        ▷ where $\mathrm{vol}(A) = \int_A 1 d\mu$
```

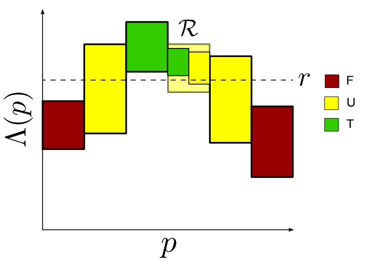


Fig. 1. Refinement in threshold synthesis with $\geq r$. Parameter values are on the x-axis, probabilities on the y-axis. Each box describes a parameter region (width), and its probability bounds (height). The refinement of $\mathcal{R}$ yields regions in T and in U.

of Φ is guaranteed for the whole region $\mathcal{R}$ and thus it is added to T. Otherwise, the algorithm tests whether $\mathcal{R}$ can be added to the set F by checking if $\Lambda^{\mathcal{R}}_{\max}$ is below the threshold r. If $\mathcal{R}$ is neither in T nor in F, it forms an undecided subspace that is added to the set U. The algorithm terminates when the volume of the undecided subspace is negligible with respect to the volume of the entire parameter space, i.e. $\mathrm{vol}(U)/\mathrm{vol}(\mathcal{P}) \leq \varepsilon$, where ε is the input tolerance. Otherwise, the algorithm continues to the next iteration where U is further refined.

Since, for a finite formula ϕ, only a finite number of refinement steps is needed to meet the desired tolerance, the algorithm always terminates. The initial decomposition of the parameter space is guided by a prior sampling of probability values. For more details see [21].

Max Synthesis. Algorithm 2 is used to solve the max synthesis problem, which returns the set T containing the parameter valuations that maximize $\Phi = P_{=?}[\phi]$ and the set F not yielding the maximum value of Φ. Let R be a partition of T. For each subspace $\mathcal{R} \in R$, the algorithm computes bounds $\Lambda^{\mathcal{R}}_{\min}$ and $\Lambda^{\mathcal{R}}_{\max}$ on the maximal and minimal probability that $\mathcal{C}_\mathcal{R}$ satisfies Φ (line 5). The algorithm then rules out subspaces that are guaranteed to be included in F, by deriving an under-approximation (MLB) to the maximum satisfaction probability (line 7). If $\Lambda^{\mathcal{R}}_{\max}$ is below the under-approximation, the subspace $\mathcal{R}$ can be safely added to the set F (line 9). Otherwise, it is added to the set T. The bound MLB is derived as follows. In the naive approach, the algorithm uses the maximum over the least bounds in the partition of T, that is, $\mathsf{MLB} = \max\{\Lambda^{\mathcal{R}}_{\min} \mid \mathcal{R} \in R\}$. Let $\overline{\mathcal{R}}$ be the region with highest lower bound. The sampling-based approach improves on this by sampling a set of parameters $\{p_1, p_2, \ldots\} \subseteq \overline{\mathcal{R}}$ and taking the highest value of $\Lambda(p)$, that is, $\mathsf{MLB} = \max\{\Lambda(p_i) \mid p_i \in \{p_1, p_2, \ldots\}\}$. Although regular CSL model checking is nearly as expensive as the computation of the bounds for a pCTMC, the bound obtained by the sampling method excludes more boxes (see Fig. 2), which in turn leads to fewer refinements in the next iteration. In

Algorithm 2 Max Synthesis

```
Require: pCTMC $\mathcal{C}_\mathcal{P}$ over parameter space $\mathcal{P}$, CSL
    formula $\Phi = P_{=?}[\phi]$ and probability tolerance $\epsilon > 0$
Ensure: $\Lambda^\perp$, $\Lambda^\top$, $T$ and $F$ as in Problem 2
 1: $F \leftarrow \emptyset$, $T \leftarrow \mathcal{P}$
 2: repeat
 3:     $R \leftarrow \mathsf{decompose}(T)$, $T \leftarrow \emptyset$
 4:     for each $\mathcal{R} \in R$ do
 5:         $(\Lambda^\mathcal{R}_{\min}, \Lambda^\mathcal{R}_{\max}) \leftarrow \mathsf{computeBounds}(\mathcal{C}_\mathcal{R}, \phi)$
 6:     $\mathsf{MLB} \leftarrow \mathsf{getMaximalLowerBound}(R)$
 7:     for each $\mathcal{R} \in R$ do
 8:         if $\Lambda^\mathcal{R}_{\max} < \mathsf{MLB}$ then
 9:             $F \leftarrow F \cup \mathcal{R}$
10:         else
11:             $T \leftarrow T \cup \mathcal{R}$
12:     $\Lambda^\perp \leftarrow \min\{\Lambda^\mathcal{R}_{\min} \mid \mathcal{R} \in T\}$
13:     $\Lambda^\top \leftarrow \max\{\Lambda^\mathcal{R}_{\max} \mid \mathcal{R} \in T\}$
14: until $\Lambda^\top - \Lambda^\perp < \epsilon$
```

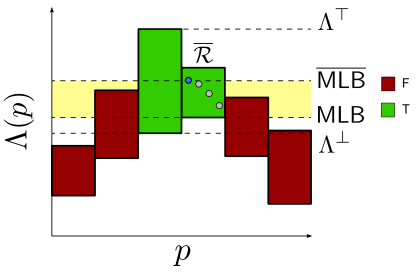


Fig. 2. Refinement in max synthesis. The two outermost regions (in red) cannot contain the maximum, as their upper bound is below the maximum lower bound (MLB) found at region $\overline{\mathcal{R}}$. The maximum lower bound is improved by sampling few points $p \in \overline{\mathcal{R}}$ and taking the highest value ($\overline{\mathsf{MLB}}$) of the satisfaction function $\Lambda(p)$. The yellow area highlights the improvement.

this case we perform additional checks, not discussed here, for detecting and discarding regions containing points of jump discontinuity that might prevent the algorithm from reaching the target accuracy and thus from terminating.

The overall time complexity of the synthesis algorithms is directly determined by the number of subspaces that need to be analysed to obtain the desired precision. This number depends on the number of unknown parameters, the shape of the satisfaction function and the type of synthesis. In practice, the algorithms scale exponentially in the number of parameters and linearly in the volume of the parameter space.

6 Results

We demonstrate the applicability and efficiency of the developed algorithms on two case studies.

6.1 Epidemic model

The SIR model [16] describes the epidemic dynamics in a closed population of susceptible (S), infected (I) and recovered (R) individuals. In the model, a susceptible individual is infected after a contact with an infected individual with rate k_i. Infected individuals recover with rate k_r, after which they are immune to the infection. We can describe this process with the following biochemical reaction model with mass action kinetics: $i : S + I \xrightarrow{k_i} I + I, r : I \xrightarrow{k_r} R$. We represent the model as a pCTMC with k_i and k_r as parameters, and initial populations $S = 95$, $I = 5$, $R = 0$. We consider the time-bounded CSL formula

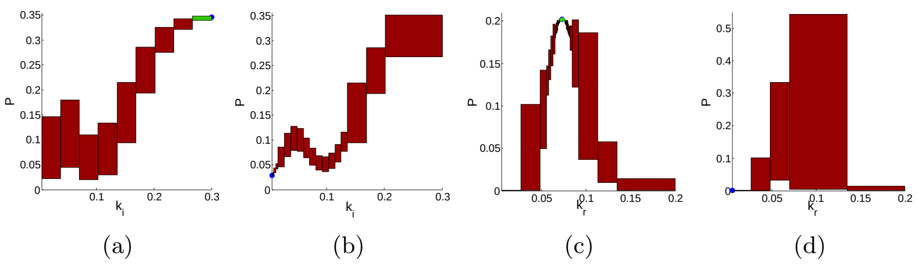


Fig. 3. Solution to max (a,c) and min (b,d) synthesis using sampling-based refinement for $P_{=?}[(I > 0)U^{[100,120]}(I = 0)]$. Probability tolerance $\epsilon = 1\%$ (a,c) and $\epsilon = 0.1\%$ (b,d).

Problem	k_i	k_r	Runtime	Subspaces	Λ^* [%]	T
1. Max	$[0.005, 0.3]$	0.05	16.5 s	9	33.94	$[0.267, 0.3]$
2. Min	$[0.005, 0.3]$	0.05	49.5 s	21	2.91	$[0.005, 0.0054]$
3. Max	0.12	$[0.005, 0.2]$	99.7 s	57	19.94	$[0.071, 0.076]$
4. Min	0.12	$[0.005, 0.2]$	10.4 s	5	0.005	$[0.005, 0.026]$
5. Max	$[0.005, 0.3]$	$[0.005, 0.2]$	3.6 h	5817	35.01	$[0.217, 0.272] \times [0.053, 0.059]$
6. Max	$[0.005, 0.3]$	$[0.005, 0.2]$	6.2 h	10249	34.77	$[0.209, 0.29] \times [0.051, 0.061]$

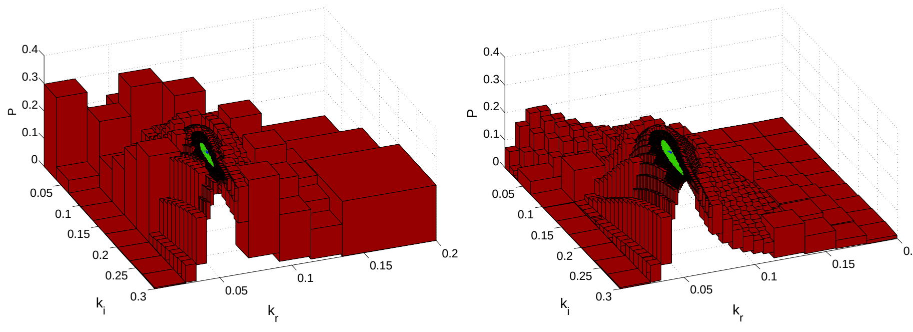
Table 1. The computation of the synthesis problems for $P_{=?}[(I > 0)U^{[100,120]}(I = 0)]$ using probability tolerance $\epsilon = 1\%$ (problems 1,3,5,6) and $\epsilon = 0.1\%$ (problems 2,4). The sampling-based refinement is used apart from problem 5. The minimal bounding box of T is reported in problems 5 and 6. Λ^* denotes $\Lambda^\perp$ (problems 1,3,5,6) and $\Lambda^\top$ (problems 2,4).

$\Phi = P_{=?}[(I > 0)U^{[100,120]}(I = 0)]$, which asks for the probability that the infection lasts for at least 100 time units, and dies out before 120 time units. Model parameters and the property are taken from [5], where the authors estimate the satisfaction function for Φ following a Bayesian approach[3].

Figure 3 and Table 1 (problems 1-4) illustrate the solutions using sampling-based refinement for max and min synthesis problems over one-dimensional parameter spaces. We report that, in order to meet the desired probability tolerance, problems 2 (Fig. 3b) and 3 (Fig. 3c) require a high number of refinement steps due to two local extrema close to the minimizing region and due to a bell-shaped Λ with the maximizing region at the top, respectively. Our precise results for problem 1 (Fig. 3a) improve on the estimation in [5], where in the equivalent experiment the max probability is imprecisely registered at smaller k_i values.

We also compare the solutions to the max synthesis problem over the two-dimensional parameter space obtained by applying Alg. 2 with sampling-based (Fig. 4a, problem 5 in Table 1) and naive (Fig. 4b, problem 6 in Table 1) refinement. In the former case, a more precise T region is obtained (with a volume 2.04 smaller than in the naive approach), thus giving a more accurate approximation of the max probability. Sampling also allows ruling out earlier those parameter regions that are outside the final solution, thus avoiding unnecessary decompo-

[3] In [5], a linear-time specification equivalent to Φ is given.



(a) (b)

Fig. 4. Solutions to max synthesis with sampling-based refinement (a) and without sampling (b) for $P_{=?}[(I > 0)U^{[100,120]}(I = 0)]$ using probability tolerance $\epsilon = 1\%$.

sitions and improving the runtime (1.72 times slower without sampling). This is visible by the coarser approximations of probabilities in the F region.

6.2 DNA walkers

We revisit models of a DNA walker, a man-made molecular motor that traverses a track of anchorages and can take directions at junctions in the track [22], which can be used to create circuits that evaluate Boolean functions. `PRISM` models of the walker stepping behaviour were developed previously [9] based on rate estimates in the experimental work. The walker model is modified here to allow uncertainty in the stepping rate, and we consider its behaviour over a single-junction circuit. Given a distance d between the walker-anchorage complex and an uncut anchorage, and d_a being the distance between consecutive anchorages, the stepping rate k is defined as: $k = k_s$ when $d \leq 1.5d_a$, $k = c \cdot k_s/50$ when $1.5d_a < d \leq 2.5d_a$, $k = c \cdot k_s/100$ when $2.5d_a < d \leq 24\text{nm}$ and $k = 0$, otherwise.

The base stepping rate $k_s \in [0.005, 0.020]$ is now defined as an interval, as opposed to the original value of 0.009. We have also added factor c for steps between anchorages that are not directly adjacent, but we will assume $c = 1$ for now. The base stepping rate may depend on buffer conditions and temperature, and we want to verify the robustness of the walker with respect to the uncertainty in the value of k_s.

We compute the minimal probability of the walker making it onto the correct final anchorage (min synthesis for the property $P_{=?}[F^{[T,T]}$ finish-correct]) and the maximum probability of the walker making it onto the incorrect anchorage (max synthesis for the property $P_{=?}[F^{[T,T]}$ finish-incorrect). We list the probabilities at $T = 15, 30, 45, 200$ minutes in Table 2. For time $T = 30, 45, 200$, we note that the walker is robust, as the minimal guaranteed probability for correct outcome is greater than the maximum possible probability for incorrect outcome. For time $T = 15$ this is not the case. We also consider a property

			Runtime		Subspaces	
Time bound	Min. correct	Max. incorrect	$\emptyset$	Sampling	$\emptyset$	Sampling
$T = 15$	1.68%	5.94%	0.55 s	0.51 s	22	11
$T = 30$	14.86%	10.15%	1.43 s	1.35 s	35	15
$T = 45$	33.10%	12.25%	3.53 s	2.14 s	61	21
$T = 200$	79.21%	16.47%	213.57s	88.97 s	909	329

Table 2. The computation of min-synthesis for $P_{=?}[F^{[T,T]}$ finish-correct] and max-synthesis for $P_{=?}[F^{[T,T]}$ finish-incorrect] using $k_s \in [0.005, 0.020], c = 1$ and probability tolerance $\epsilon = 1\%$. The runtime and subspaces are listed for the first property.

that provides bounds on the *ratio* between the walker finishing on the correct versus the incorrect anchorage. The rates $c \cdot k_s/50$ and $c \cdot k_s/100$ correspond to the walker stepping onto anchorages that are not directly adjacent, which affects the probability for the walker to end up on the unintended final anchorage. For higher values of c, we expect the walker to end up in the unintended final anchorage more often. Now we add uncertainty on the value of c, so that $c \in [0.25, 4]$, and define the performance related property $P_{\geq 0.4}[F^{[30,30]}$ finish-correct$] \wedge P_{\leq 0.08}[F^{[30,30]}$ finish-incorrect], that is, the probability of the walker to make it onto the correct anchorage is at least 40% by time $T = 30$ min, while the probability for it to make it onto the incorrect anchorage is no greater than 8%. In other words, we require a correct signal of at least 40% and a correct-to-incorrect ratio of at least 5 by time $T = 30$ min. We define a similar property at time $T = 200$ min, this time requiring a signal of at least 80%: $P_{\geq 0.8}[F^{[200,200]}$ finish-correct$] \wedge P_{\leq 0.16}[F^{[200,200]}$ finish-incorrect]. The synthesized ranges of k_s and c where the properties hold are shown in Fig. 5.

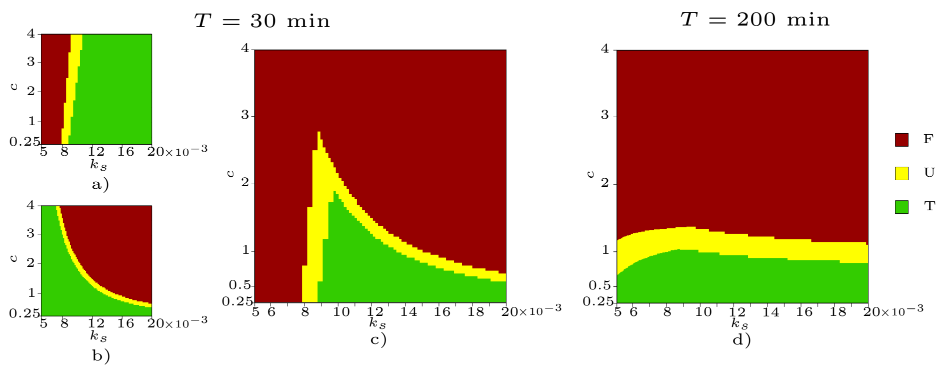


Fig. 5. The computation and results of the threshold synthesis for different formulae, using volume tolerance $\varepsilon = 10\%$. a) $\Phi_1 = P_{\geq 0.4}[F^{[30,30]}$ finish-correct], runtime 443.5 s, 2692 subspaces. b) $\Phi_2 = P_{\leq 0.08}[F^{[30,30]}$ finish-incorrect], runtime 132.3 s, 807 subspaces. c) $\Phi_1 \wedge \Phi_2$. d) $P_{\geq 0.8}[F^{[200,200]}$ finish-correct$] \wedge P_{\leq 0.16}[F^{[200,200]}$ finish-incorrect], runtime 12,3 h, 47229 subspaces.

7 Conclusion

We have developed efficient algorithms for synthesising rate parameters for biochemical networks so that a given reliability or performance requirement, expressed as a time-bounded CSL formula, is guaranteed to be satisfied. The techniques are based on the computation of lower and upper probability bounds of [6] in conjunction with region refinement and sampling. The high computational costs observed in our case studies can be reduced by parallel processing of individual subspaces, or by utilizing advanced uniformisation techniques [19, 8]. We plan to include the synthesis algorithms in the `param` module of the `PRISM` model checker [7, 18].

Acknowledgements

We would like to thank Luboš Brim for helpful input and fruitful discussions. This work has been partially supported by the ERC Advanced Grant VERIWARE, Microsoft Studentship (Frits Dannenberg) and the Ministry of Education, Youth, and Sport project No. CZ.1.07/2.3.00/30.0009 - Employment of Newly Graduated Doctors of Science for Scientific Excellence (Milan Češka).

References

1. A. Andreychenko, L. Mikeev, D. Spieler, and V. Wolf. Parameter Identification for Markov Models of Biochemical Reactions. In *Computer Aided Verification (CAV)*, LNCS, pages 83–98. Springer Berlin Heidelberg, 2011.
2. A. Aziz, K. Sanwal, V. Singhal, and R. Brayton. Verifying Continuous Time Markov Chains. In *Computer Aided Verification (CAV)*, volume 1102 of *LNCS*, pages 269–276. Springer Berlin Heidelberg, 1996.
3. C. Baier, B. Haverkort, H. Hermanns, and J. Katoen. Model-Checking Algorithms for Continuous-Time Markov Chains. *IEEE Trans. on Soft. Eng.*, 29(6):524–541, 2003.
4. L. Bortolussi and G. Sanguinetti. Learning and designing stochastic processes from logical constraints. In *Quantitative Evaluation of Systems (QEST)*, volume 8054 of *LNCS*, pages 89–105. Springer Berlin Heidelberg, 2013.
5. L. Bortolussi and G. Sanguinetti. Smoothed model checking for uncertain continuous time markov chains. *CoRR ArXiv*, 1402.1450, 2014.
6. L. Brim, M. Češka, S. Dražan, and D. Šafránek. Exploring parameter space of stochastic biochemical systems using quantitative model checking. In *Computer Aided Verification (CAV)*, volume 8044 of *LNCS*, pages 107–123. Springer Berlin Heidelberg, 2013.
7. T. Chen, E. M. Hahn, T. Han, M. Kwiatkowska, H. Qu, and L. Zhang. Model repair for Markov decision processes. In *Theoretical Aspects of Software Engineering (TASE)*, pages 85–92. IEEE, 2013.
8. F. Dannenberg, E. M. Hahn, and M. Kwiatkowska. Computing cumulative rewards using fast adaptive uniformisation. In *CMSB*, volume 8130 of *LNCS*, pages 33–49. Springer, 2013.

9. F. Dannenberg, M. Kwiatkowska, C. Thachuk, and A. Turberfield. DNA walker circuits: Computational potential, design, and verification. *Natural Computing*, 2014. To appear.
10. B. L. Fox and P. W. Glynn. Computing Poisson Probabilities. *CACM*, 31(4):440–445, 1988.
11. D. T. Gillespie. Exact Stochastic Simulation of Coupled Chemical Reactions. *Journal of Physical Chemistry*, 81(25):2340–2381, 1977.
12. E. M. Hahn, H. Hermanns, and L. Zhang. Probabilistic reachability for parametric Markov models. *International Journal on Software Tools for Technology Transfer (STTT)*, 13(1):3–19, 2011.
13. T. Han, J. Katoen, and A. Mereacre. Approximate parameter synthesis for probabilistic time-bounded reachability. In *Real-Time Systems Symposium (RTSS)*, pages 173–182. IEEE, 2008.
14. S. K. Jha and C. J. Langmead. Synthesis and infeasibility analysis for stochastic models of biochemical systems using statistical model checking and abstraction refinement. *Theor. Comput. Sci.*, 412(21):2162–2187, 2011.
15. J.-P. Katoen, D. Klink, M. Leucker, and V. Wolf. Three-valued abstraction for continuous-time markov chains. In *Computer Aided Verification (CAV)*, volume 4590 of *LNCS*, pages 311–324. Springer Berlin Heidelberg, 2007.
16. W. O. Kermack and A. G. McKendrick. Contributions to the mathematical theory of epidemics. ii. the problem of endemicity. *Proceedings of the Royal society of London. Series A*, 138(834):55–83, 1932.
17. M. Kwiatkowska, G. Norman, and D. Parker. Stochastic Model Checking. In *SFM 2007*, volume 4486 of *LNCS*, pages 220–270. Springer Berlin Heidelberg, 2007.
18. M. Kwiatkowska, G. Norman, and D. Parker. PRISM 4.0: Verification of Probabilistic Real-time Systems. In *CAV 2011*, volume 6806 of *LNCS*, pages 585–591. Springer Berlin Heidelberg, 2011.
19. M. Mateescu, V. Wolf, F. Didier, and T. A. Henzinger. Fast Adaptive Uniformization of the Chemical Master Equation. *IET Systems Biology*, 4(6):441–452, 2010.
20. K. Sen, M. Viswanathan, and G. Agha. Model-checking markov chains in the presence of uncertainties. In *Tools and Algorithms for the Construction and Analysis of Systems (TACAS)*, volume 3920 of *LNCS*, pages 394–410. Springer Berlin Heidelberg, 2006.
21. M. Češka, F. Dannenberg, M. Kwiatkowska, and N. Paoletti. Precise parameter synthesis for stochastic biochemical systems. Technical Report CS-RR-14-08, Department of Computer Science, University of Oxford, 2014.
22. S. F. J. Wickham, J. Bath, Y. Katsuda, M. Endo, K. Hidaka, H. Sugiyama, and A. J. Turberfield. A DNA-based molecular motor that can navigate a network of tracks. *Nature nanotechnology*, 7:169–73, 2012.