Parameter Identification Using δ-Decisions for Hybrid Systems in Biology

Bing Liu, Soonho Kong, Sicun Gao, and Edmund M. Clarke

Computer Science Department, Carnegie Mellon University, PA 15213, USA

Abstract. Biological systems can possess multiple operational modes with specific nonlinear dynamics in each mode. Hybrid automata and its variants are often used to model and analyze the dynamics of such systems. Highly nonlinear hybrid models are difficult to analysis and usually have many parameters. An important problem is to identify parameter values using which the model can reach certain states of interests. We present a parameter identification framework using δ-complete decision procedures, which can solve satisfiability modulo theories (SMT) problems over the reals with a wide range of nonlinear functions, including ordinary differential equations (ODEs). We have demonstrated our method on two highly nonlinear hybrid systems: the prostate cancer progression model and the cardiac cellular action potential model. The results show that the parameter identification framework is convenient and efficient for performing model selection and personalized therapy optimization. We have also identified crucial parameter ranges related to cardiac disorders.

1 Introduction

The functioning of a biological system depends on its dynamics, i.e., the evolution of its constituent elements in space and time, as well as the interactions among these elements. Computational modeling and analysis methods are playing a crucial role in understanding the complex dynamics of biological systems [1]. In recent years, a variety of computational models have been developed, ranging from qualitative models that focus on the generic properties of biological systems [2, 3] to quantitative models that can simulate the time course of biological systems under various conditions [4, 5]. The choice of a modeling formalism depends on the goals of the modeling effort as well as the biological context.

One of the key aspects of biological systems is the differing behavior of the cell in various states. For example, different stages of the cell cycle are driven by the activation of different signaling pathways [6]. Hence in many settings, biological systems can possess multiple operational modes with specific nonlinear dynamics in each mode. Multiple variants of the formalism called hybrid automata [7] is often used in this context [8–13].

Hybrid automata are well-studied formalisms that are used to model the behavior of hybrid systems, which consist of discrete control computations in a

continuous environment. The state space of a hybrid automaton is defined by a finite set of continuous variables and modes. A system of differentials equations over the variables is associated with each mode. At any given time the automaton will reside in one of its modes and each variable will evolve according its differential equation in the mode. When the automaton satisfies a jump condition, it will switch to a new mode. As a result the system will start evolving according the differential equations associated with the new mode. It is worth to note that ODE models are special cases of hybrid automata and the techniques we develop here can be adapted to ODE models as well.

A hybrid automaton model of a biological system often involves many parameters such as the rate constants of the biochemical reactions, the initial conditions, and the threshold values in the jump conditions. Almost always, only a few rate constants will be available or can be measured experimentally. One needs to estimate the values of unknown rate constants by fitting the model to the experimental observations. Furthermore, it is also crucial to figure out what initial conditions or jump conditions may lead to a disorder or safety of the system, especially when studying hybrid systems for synthetic biology and clinical therapy [14]. All these questions can be answered by the *parameter synthesis* procedure, which aims to identify sets of parameters for which the system reach a given set of states. However, parameter synthesis for hybrid systems is difficult due to the interplay between the continuous and discrete components of the dynamics. The high expressive power of the mixed dynamics renders even simple reachability questions undecidable [7]. Various lines of work have explored ways to mitigate this problem [15–20].

In this paper, we propose a novel framework to tackle the parameter synthesis problem for nonlinear hybrid models in biology using δ-complete decision procedures. We describe the set of states of interested as a first-order logic formula and perform bounded model checking to determine reachability of these states. We then adapt an interval constrains propagation (ICP) based algorithm to explore the parameter spaces and identify the sets of resulting parameters. Note that determining the truth value of first-order sentences over the reals with nonlinear real functions is a well-known undefinable problem. Here we employ our recently developed δ-*decision* based framework to ask for answers that may have one-sided δ-bounded errors. That is, given a first-order sentence ϕ, we ask whether ϕ is false, or some δ-relaxation of ϕ is true, which is defined as a slight syntactic variation of ϕ. We have proved that the δ-complete decision procedures can solve SMT problems over the reals with arbitrary computable real functions [21] including solutions of Lipschitz-continuous ODEs [22].

We show the applicability of our method by carrying out two case studies. The first one involves a hybrid system built by [23], which aims to study the hormone therapy for prostate cancer. We show that our method is able to perform model selection by ruling out model candidates which hopelessly fit the experimental observation. We also used our method to optimize personalized treatment schemes for individual patients to achieve maximum therapeutic efficacy. In the second case study we analyzed a cardiac cell model developed by

[24] in order to investigate the cardiac disorders. We identified parameter ranges for which a cardiac cell may lose excitability. The results show that our method scales and can obtain biological insights that are consistent with experimental observations.

Turning to related work, a survey of modeling and analysis of biological systems using hybrid models can be found in [25]. Formal verification of hybrid systems is a well-established domain [26]. Analyzing the properties of biochemical networks using model checking techniques [27] is being actively pursued by a number of groups [28–32]. Of particular interest in our context are parameter synthesis methods which identify range of parameters for which some qualitative behavior is exhibited. The method presented in [33] can deal with parameter synthesis problem for piecewise affine linear systems. For nonlinear ODE systems, [34] described a more efficient way to explore the parameter space based on adaptive sampling and numerical simulation.

The rest of the paper is organized as follows. The next section introduces our δ-complete decision procedures over the reals. We formulated the parameter synthesis problem for hybrid automata in Section 2.2. In Section 2.3, we present techniques for synthesizing parameters using δ-complete decision procedures. In the subsequent section we present two case studies. In the final section, we summarize the paper and discuss future work.

2 Methods

2.1 $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-formulas and δ-decisions over the reals

We first briefly review our framework of δ-decision problems for first-order sentences over the reals with computable real functions. The notion of computable functions over the real numbers are developed in Computable Analysis [35]. In our recent work [21, 22], we developed a theory of decision problems over the reals with computable functions. It suffices to note that most common continuous real functions are computable, such as addition, multiplication, absolute value, min, max, exp, sin and solutions of Lipschitz-continuous ordinary differential equations. Compositions of computable functions are computable. In fact, the notion of computability of real functions directly corresponds to whether they can be numerically simulated. We write $\mathcal{F}$ to denote an arbitrary collection of symbols representing computable functions over $\mathbb{R}^n$ for various n. We consider the first-order formulas with a signature $\mathcal{L}_{\mathbb{R}_\mathcal{F}} = \langle 0, 1, \mathcal{F}, > \rangle$. Note that constants are seen as 0-ary functions in $\mathcal{F}$. $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-formulas are evaluated in the standard way over the corresponding structure $\mathbb{R}_\mathcal{F} = \langle \mathbb{R}, \mathcal{F}, > \rangle$. We use atomic formulas of the form $t(x_1, ..., x_n) > 0$ or $t(x_1, ..., x_n) \geq 0$, where $t(x_1, ..., x_n)$ are built up from functions in $\mathcal{F}$. To avoid extra preprocessing of formulas, we give an explicit definition of $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-formulas as follows.

$\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-Formulas Let $\mathcal{F}$ be a collection of Type 2 functions, which contains at least 0, unary negation -, addition +, and absolute value $|\cdot|$. We define:

$$t := x \mid f(t(\boldsymbol{x})), \text{ where } f \in \mathcal{F}, \text{ possibly constant;}$$
$$\varphi := t(\boldsymbol{x}) > 0 \mid t(\boldsymbol{x}) \geq 0 \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \exists x_i \varphi \mid \forall x_i \varphi.$$

In this setting $\neg\varphi$ is regarded as an inductively defined operation which replaces atomic formulas $t > 0$ with $-t \geq 0$, atomic formulas $t \geq 0$ with $-t > 0$, switches $\wedge$ and $\vee$, and switches $\forall$ and $\exists$. Implication $\varphi_1 \to \varphi_2$ is defined as $\neg\varphi_1 \vee \varphi_2$.

We define

$$\exists^{[u,v]} x.\varphi =_{df} \exists x.(u \leq x \wedge x \leq v \wedge \varphi),$$
$$\forall^{[u,v]} x.\varphi =_{df} \forall x.((u \leq x \wedge x \leq v) \to \varphi),$$

where u and v denote $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$ terms whose variables only contain free variables in φ, excluding x. It is easy to check that $\exists^{[u,v]} x.\varphi \leftrightarrow \neg\forall^{[u,v]} x.\neg\varphi$. We say a sentence is bounded if it only involves bounded quantifiers. A *bounded $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-sentence* is

$$Q_1^{[u_1,v_1]} x_1 \cdots Q_n^{[u_n,v_n]} x_n \ \psi(x_1, ..., x_n).$$

$Q_i^{[u_i,v_i]}$s are bounded quantifiers, and $\psi(x_1, ..., x_n)$ is a quantifier-free $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-formula. We write $\psi(x_1, ..., x_n)$ as $\psi[t_1(\boldsymbol{x}) > 0, ..., t_k(\boldsymbol{x}) > 0; t_{k+1}(\boldsymbol{x}) \geq 0, ..., t_m(\boldsymbol{x}) \geq 0]$ to emphasize that $\psi(\boldsymbol{x})$ is a Boolean combination of the atomic formulas shown.

δ-Variants Let $\delta \in \mathbb{Q}^+ \cup \{0\}$, and φ an $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-formula of the form

$$\varphi : \ Q_1^{I_1} x_1 \cdots Q_n^{I_n} x_n \ \psi[t_i(\boldsymbol{x}, \boldsymbol{y}) > 0; t_j(\boldsymbol{x}, \boldsymbol{y}) \geq 0],$$

where $i \in \{1, ...k\}$ and $j \in \{k+1, ..., m\}$. The *δ-weakening* φ^δ of φ is defined as the result of replacing each atom $t_i > 0$ by $t_i > -\delta$ and $t_j \geq 0$ by $t_j \geq -\delta$. That is,

$$\varphi^\delta : \ Q_1^{I_1} x_1 \cdots Q_n^{I_n} x_n \ \psi[t_i(\boldsymbol{x}, \boldsymbol{y}) > -\delta; t_j(\boldsymbol{x}, \boldsymbol{y}) \geq -\delta].$$

We then have the following main decidability result.

δ-Decidability Let $\delta \in \mathbb{Q}^+$ be arbitrary. There is an algorithm which, given any bounded φ, correctly returns one of the following two answers:

- "δ-True": φ^δ is true.
- "False": φ is false.

Note when the two cases overlap, either answer is correct.

We call this new decision problem the δ-decision problem for $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-sentences. If an algorithm solves the δ-decision problem correctly for a set S of $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-sentences, we say it is δ-complete for S. From δ-decidability, δ-complete decision procedures always exists for bounded $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-formulas. In practice, we have shown that the combination of the DPLL(T) framework and Interval Constraint Propagation (ICP) indeed gives us a δ-complete decision procedure. We implemented such procedures in our tool dReal [36], which solves formulas containing transcendental functions and ordinary differential equations. In what follows we will see how δ-complete decision procedures provide the engine for parameter synthesis of biological hybrid systems.

2.2 Parameterized $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-representations of hybrid automata

We now describe hybrid automata using $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-formulas, and define parameterization and perturbations on them. A hybrid system [7] is a tuple $H = \langle X, Q, \mathsf{flow}, \mathsf{guard}, \mathsf{reset}, \mathsf{inv}, \mathsf{init}\rangle$ where $X \subseteq \mathbb{R}^n$ specifies the range of the *continuous variables* $\boldsymbol{x}$ of the system. $Q = \{q_0, ..., q_m\}$ is a finite set of discrete *control modes*. $\mathsf{flow} \subseteq Q \times X \times \mathbb{R} \times X$ specifies the *continuous dynamics* for each mode. The flow predicate is usually defined either as explicit mappings from $\boldsymbol{a}_0$ and t to $\boldsymbol{a}_t$, or as solutions of systems of differential equations/inclusions that specify the derivative of $\boldsymbol{x}$ over time. $\mathsf{jump} \subseteq Q \times X \times Q \times X$ specifies the *jump conditions* between modes. $\mathsf{inv} \subseteq Q \times X$ defines the *invariant conditions* for the system to stay in a control mode. $\mathsf{init} \subseteq Q \times X$ defines the set of *initial configurations* of the system. Without loss of generality we always assume that q_0 is the only intial mode, and $\mathsf{init}_{q_0} \subseteq X$ denotes the initial values for the continuous variables.

$\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-representations of hybrid automata Let $H = \langle X, Q, \mathsf{flow}, \mathsf{jump}, \mathsf{inv}, \mathsf{init}\rangle$ be an n-dimensional hybrid automaton. Let $\mathcal{F}$ be a set of real functions, and $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$ the corresponding first-order language. We say that H has an $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-representation, if for every $q, q' \in Q$, there exists quantifier-free $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-formulas

$$\phi^q_{\mathsf{flow}}(\boldsymbol{x}, \boldsymbol{x}_0, t), \phi^{q\to q'}_{\mathsf{jump}}(\boldsymbol{x}, \boldsymbol{x}'), \phi^q_{\mathsf{inv}}(\boldsymbol{x}), \phi^q_{\mathsf{init}}(\boldsymbol{x})$$

such that for all $\boldsymbol{a}, \boldsymbol{a}' \in \mathbb{R}^n$, $t \in \mathbb{R}$:

- $\mathbb{R} \models \phi^q_{\mathsf{flow}}(\boldsymbol{a}, \boldsymbol{a}', t)$ iff $(q, \boldsymbol{a}, \boldsymbol{a}', t) \in \mathsf{flow}$.
- $\mathbb{R} \models \phi^{q\to q'}_{\mathsf{jump}}(\boldsymbol{a}, \boldsymbol{a}')$ iff $(q, q', \boldsymbol{a}, \boldsymbol{a}') \in \mathsf{jump}$.
- $\mathbb{R} \models \phi^q_{\mathsf{inv}}(\boldsymbol{a})$ iff $(q, \boldsymbol{a}) \in \mathsf{inv}$.
- $\mathbb{R} \models \phi^q_{\mathsf{init}}(\boldsymbol{a})$ iff $q = q_0$ and $\boldsymbol{a} \in \mathsf{init}_{q_0}$.

We can write $H = \langle X, Q, \phi_{\mathsf{flow}}, \phi_{\mathsf{jump}}, \phi_{\mathsf{inv}}, \phi_{\mathsf{init}}\rangle$ to emphasize that H is $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-represented. But from now on we simply write $\mathsf{flow}, \mathsf{jump}, \mathsf{inv}, \mathsf{init}$ to denote these logic formulas, so that we can use $H = \langle X, Q, \mathsf{flow}, \mathsf{jump}, \mathsf{inv}, \mathsf{init}\rangle$ directly to denote the $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-representation of H.

We say a hybrid automaton H has a *computable representation*, if H has an $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-representation, where $\mathcal{F}$ is an arbitrary set of computable functions. Combining continuous and discrete behaviors, the trajectories of hybrid systems are *piecewise continuous*. This motivates a two-dimensional structure of time, with which we can keep track of both the discrete changes and the duration of each continuous flow. A *hybrid time domain* T is a subset of $\mathbb{N} \times \mathbb{R}$ of the form $T_m = \{(i, t) : i < m \text{ and } t \in [t_i, t'_i] \text{ or } [t_i, +\infty)\}$, where $m \in \mathbb{N} \cup \{+\infty\}$, $\{t_i\}_{i=0}^m$ is an increasing sequence in $\mathbb{R}^+$, $t_0 = 0$, and $t'_i = t_{i+1}$. We write the set of all hybrid time domains as $\mathbb{H}$. Suppose $X \subseteq \mathbb{R}^n$ and T_m is a hybrid time domain. A *hybrid trajectory* is any continuous function $\xi : T_m \to X$. We write Ξ_X to denote the set of all possible hybrid trajectories from $\mathbb{H}$ to X. We can now define trajectories of a given hybrid automaton. The intuition behind the following definition is straightforward. The labeling function $\sigma^H_\xi(i)$ is used to map a step i to the corresponding discrete mode in H. In each mode, the system

flows continuously following the dynamics defined by $\mathsf{flow}(q, \boldsymbol{x}_0, t)$. Note that $(t - t_k)$ is the actual duration in the k-th mode. When a switch between two modes is performed, it is required that $\xi(k+1, t_{k+1})$ is updated from the exit value $\xi(k, t'_k)$ in the previous mode, following the jump conditions.

We say that $\xi : T_m \to X$ is *a trajectory of H of discrete depth m*, if there exists a *labeling function* $\sigma^H_\xi : \mathbb{N} \to Q$ such that:

- $\sigma^H_\xi(0) = q_0$ and $\mathbb{R}_\mathcal{F} \models \mathsf{init}_{q_0}(\xi(0,0))$.
- For any $(i,t) \in T_m$, $\mathbb{R}_\mathcal{F} \models \mathsf{inv}_{\sigma^H_\xi(i)}(\xi(i,t))$.
- When $i = 0$, $\mathbb{R}_\mathcal{F} \models \mathsf{flow}_{q_0}(\xi(0,0), \xi(0,t), t)$.
- When $i = k+1$, where $0 < k+1 < m$,

$$\mathbb{R}_\mathcal{F} \models \mathsf{flow}_{\sigma^H_\xi(k+1)}(\xi(k+1, t_{k+1}), \xi(k+1, t), (t - t_{k+1})) \text{ and}$$
$$\mathbb{R}_\mathcal{F} \models \mathsf{jump}_{(\sigma^H(k) \to \sigma^H(k+1))}(\xi(k, t'_k), \xi(k+1, t_{k+1})).$$

We write $[\![H]\!]$ to denote the set of all possible trajectories of H.

Reachability Properties Let $U \subseteq X \times Q$ be a subset of the state space of H. H reaches U if there exists $\xi \in [\![H]\!]$ such that there exists $t \in \mathbb{R}$ and $n \in \mathbb{N}$ satisfying

$$(\xi(t, n), \sigma^H_\xi(n)) \in U.$$

We say H is parameterized by $\boldsymbol{p} = (p_1, ..., p_m)$, if

$$H(\boldsymbol{p}) = \langle X, Q, \mathsf{flow}(\boldsymbol{p}), \mathsf{jump}(\boldsymbol{p}), \mathsf{inv}(\boldsymbol{p}), \mathsf{init}(\boldsymbol{p}) \rangle,$$

where $\boldsymbol{p}$ are among the free variables in the $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-representation of H. Thus, the parameter synthesis problem for reachability asks for an assignment for $\boldsymbol{a} \in \mathbb{R}^m$ such that $H(\boldsymbol{a})$ reaches U.

2.3 Synthesizing parameters with δ-decisions

We now show how to encode parameter synthesis problems for $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-represented hybrid systems using $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-formulas. Throughout the following two definitions, let $H = \langle X, Q, \mathsf{flow}, \mathsf{jump}, \mathsf{init} \rangle$ be an n-dimensional $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-represented hybrid system with $|Q| = m$, and unsafe an $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-formula that encodes a subset $U \subseteq X \times Q$. Let $k \in \mathbb{N}$ and $M \in \mathbb{R}$ be the bounds on steps and time respectively. Recall that $q_0 \in Q$ always denotes the starting mode.

$\mathsf{Reach}^k_{H,q'}(\boldsymbol{x}^t_k)$ defines the states that H can reach, if after k steps of discrete changes it is in mode q'. From there, if H makes a jump from mode q' to q, then the states have the make a discrete change following $\mathsf{jump}_{q' \to q}(\boldsymbol{x}^t_k, \boldsymbol{x}_{k+1})$. As last, in mode q', any state $\boldsymbol{x}^t_{k+1}$ that H can reach should satisfy the flow conditions $\mathsf{flow}_q(\boldsymbol{x}^t_{k+1}, \boldsymbol{x}_{k+1}, t)$ in mode q. Note that after each discrete jump, a new time variable t_k is introduced and independent from the previous ones. The

(k, M)-reachability encoding of H and U, $\mathsf{Reach}^{k,M}(H, U)$, is defined as:

$$
\begin{aligned}
&\exists \boldsymbol{a} \exists^X \boldsymbol{x}_0 \exists^X \boldsymbol{x}_0^t \cdots \exists^X \boldsymbol{x}_k \exists^X \boldsymbol{x}_k^t \exists^{[0,M]} t_0 \cdots \exists^{[0,M]} t_k \\
&\Big(\mathsf{init}_{q_0}(\boldsymbol{x}_0) \wedge \mathsf{flow}_{q_0}(\boldsymbol{a}, \boldsymbol{x}_0, \boldsymbol{x}_0^t, t_0) \\
&\wedge \forall^{[0,t_0]} t \forall^X \boldsymbol{x}\ (\mathsf{flow}_{q_0}(\boldsymbol{a}, \boldsymbol{x}_0, \boldsymbol{x}, t) \rightarrow \mathsf{inv}_{q_0}(\boldsymbol{a}, \boldsymbol{x})) \\
&\wedge \bigvee_{i=0}^{k-1} \Big(\bigvee_{q,q' \in Q} \Big(\mathsf{jump}_{q \rightarrow q'}(\boldsymbol{a}, \boldsymbol{x}_i^t, \boldsymbol{x}_{i+1}) \wedge \mathsf{flow}_{q'}(\boldsymbol{a}, \boldsymbol{x}_{i+1}, \boldsymbol{x}_{i+1}^t, t_{i+1}) \\
&\wedge \forall^{[0,t_0]} t \forall^X \boldsymbol{x}\ (\mathsf{flow}_{q'}(\boldsymbol{a}, \boldsymbol{x}_{i+1}, \boldsymbol{x}, t) \rightarrow \mathsf{inv}_{q_0}(\boldsymbol{a}, \boldsymbol{x}))) \Big) \Big) \\
&\wedge\ \mathsf{unsafe}(\boldsymbol{a}, \boldsymbol{x}_k^t) \Big).
\end{aligned}
$$

H reaches U in k steps of discrete jumps with time duration less than M for each state, if and only if, $\mathsf{Reach}^{k,M}(H, U)$ is true.

3 Case Studies

We have developed an open-source tool dReach using OCaml to perform δ-complete reachability analysis for hybrid systems. dReach is built upon our SMT solver dReal [36] that implements a δ-complete decision procedure. All the experiments reported below were done using a machine with two Intel Xeon E5-2650 2.00GHz processors and 64GB RAM.

3.1 Hormone therapy for prostate cancer

Prostate cancer is the second leading cause of cancer-related deaths among men in United States [37]. Hormone therapy in the form of androgen deprivation has been a cornerstone of the management of advanced prostate cancer for several decades. However, controversy remains regarding its optimum application [38]. Continuous androgen suppression (CAS) therapy has many side effects including anemia, osteoporosis, impotence, etc. Further, most patients experience a relapse after a median duration of 18-24 months of CAS treatment, due to the proliferation of androgen-independent (AI) cancer cells.

In order to reduce side effects of CAS and to delay the time to relapse, intermittent androgen suppression (IAS) was proposed aiming to limit the duration of androgen-poor conditions and avoid emergence of AI cells [39]. In details, IAS therapy switches between on-treatment and off-treatment modes by monitoring the serum level of a tumor marker called prostate-specific antigen (PSA): (i) when the PSA level decreases and reaches a lower threshold value r_0, androgen suppression is suspended; (ii) when the PSA level increases and reaches a upper threshold value r_1, androgen suppression is resumed by the administration of medical agents.

Recent clinical phase II and III trials confirm that IAS has significant advantages in terms of quality of life and cost. However, with respect to time to relapse

and cancer-specific survival, the clinical trials suggest that to what extent IAS is superior to CAS depends on the individual patient and the on- and off-treatment scheme [40–42]. Thus, a crucial unsolved problem is how to design a personalized treatment scheme for each individual to achieve maximum therapeutic efficacy.

To answer this question, mathematical models have been developed to study the dynamics of prostate cancer under androgen suppression [43, 23, **?**,**?**]. Recently, attempts have been made to computationally classify patients and obtain the optimal treatment scheme [44, 45]. However, these results relied on simplifying nonlinear hybrid dynamical systems to more manageable versions such as piecewise linear models [44] and piecewise affine systems [45]. In this section, we show that our δ-decision based parameter synthesis approach can help to design personalized treatment scheme based on nonlinear hybrid systems with arbitrary computable real functions. Here we focus on the hybrid model presented by [23], which describes the growth of a prostate tumor as the dynamics of a mixed population of androgen-dependent (AD) and androgen-independent (AI) cells.

The model has two modes which are shown in Figure 1. $x(t)$, $y(t)$, and $z(t)$ represent the population of AD cells, the population of AI cells, and the serum androgen concentration, respectively. The growth dynamics of AD and AI cells are governed by their proliferation rate, apoptosis rate and mutation rate from AD to AI phenotype, depending on androgen concentration $z(t)$. The PSA level v (ng ml^{-1}) is defined as $v(t) = x(t) + y(t)$. The treatment is suspended or restarted according to the value of v and dv/dt. In mode 2 (off-treatment), the androgen concentration is maintained at the normal level z_0 by homeostasis. In mode 1 (on-treatment), the androgen is cleared at a rate $1/\tau$. Table 1 lists the values of model parameters.

Table 1. Prostate cancer model parameter values

Parameter	Bone metastasis	Lymph node metastasis
α_x	0.0204 d^{-1}	0.0168 d^{-1}
α_y	0.0242 d^{-1}	0.0277 d^{-1}
β_x	0.0076 d^{-1}	0.0085 d^{-1}
β_y	0.0168 d^{-1}	0.0222 d^{-1}
k_1	0.0 nM	0.0 nM
k_2	2.0	2.0
k_3	8.0 nM	8.0 nM
k_4	0.5	0.5
m_1	0.00005 d^{-1}	0.00005 d^{-1}
z_0	20.0 nM	20.0 nM
τ	62.5 d	62.5 d

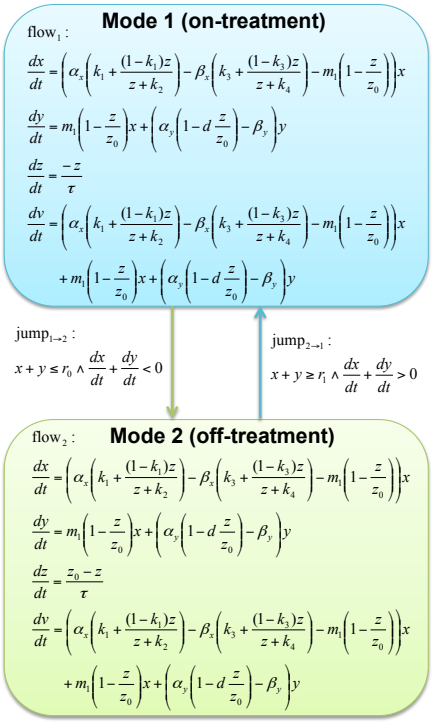

Fig. 1. The prostate cancer treatment model.

Model selection Based on different assumptions of the proliferation dynamics of AI cells, the above model has three variations, denoted as H_1, H_2, and H_3, which are discriminated by the value of d, i.e.:

- H_1: AI cells grow at the constant rate independent of the androgen level ($d = 0$)
- H_2: AI cells do not grow when the androgen level is normal ($d = 1 - \beta_2/\alpha_2$)
- H_3: AI cells decrease when the androgen level is normal ($d = 1$)

In order to perform model selection using δ-decision procedures, we specified the cancer relapse as a state with "$v > 30$", since the PSA level v reflects the total number of tumor cells. We then checked whether each of the model candidates can reach a relapse state within a bounded time of 1000 days. Here the treatment scheme threshold parameters were fixed as $r_0 = 4$ (ng ml^{-1}) and $r_1 = 10$ (ng ml^{-1}). The range of the initial concentration of androgen was given as $[10, 20]$ (nM).

Given the invariant $v \in [0, 30]$, H_1 and H_2 are unable to reach a state with $t = 1000$. In other words, H_1 and H_2 will always lead to cancer relapse state no

matter which initial androgen concentration was chosen. This is conflict with the clinical observations by [40, 41]. In contrast, H_3 is able to avoid the relapse state and reproduce the experimental observation (see Figure 2). Thus, we completed the model selection process by ruling out H_1 and H_2 and choose H_3 for further analysis.

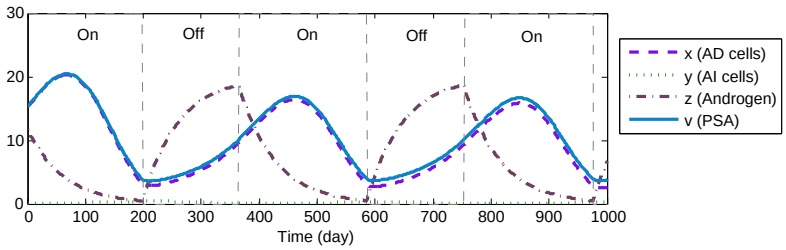

Fig. 2. Simulated time profiles of H_3 model.

Personalized therapy design We next apply our parameter synthesis method to selecting suitable therapy and designing personalized treatment scheme for individual patients. Note that the parameter values in Table 1 were estimated from clinical data of hundreds of patients [41]. Among patients, the values of some parameters vary, which causes the variability in responsiveness to hormone therapy. We select a set of "personalized parameters" including α_y (the proliferation rate of AI cells), β_y (the apoptosis rate of AI cells), m_1 (the mutation rate from AD to AI cells), and $z(0)$ (the initial androgen level). The values of these parameters can be either experimentally measured [46] or computationally determined from PSA time serials data [47].

Figure 3(a-c) illustrates the PSA dynamics of 3 patients with different personalized parameters under the same IAS treatment scheme ($r_0 = 4$, $r_1 = 10$). IAS prevents the relapse for Patient#1 and delays the relapse for Patient#2, but does not help Patient#3. Figure 3(d) shows that, by modifying the IAS scheduling parameters r_0 and r_1, the relapse of Patient#3 can be avoided or delayed. Thus, we can formulate the personalized therapy design problem as a parameter synthesis procedure: (i) fill in parameter values of a patient to H_3; (ii) set the ranges of scheduling parameters as $r_0 \in [0, 8)$ (nM) and $r_1 \in [8, 15]$; (iii) check if H_3 can reach a state with $t = 1000$ given the invariant $v \in [0, 30]$ (i.e. no relapse). If the δ-decision procedure returns False, it means that androgen suppression therapy is not suitable for the patient. Otherwise, a treatment scheme containing feasible values of r_0 and r_1 will be returned, which could prevent or delay the relapse of the patient. Note that if $r_0 = 0$ is returned, it refers to the CAS scheme.

We tested our method on real patients data collected by [41]. The values of α_y, β_y, m_1, and $z(0)$ for each selected patient were estimated by fitting the

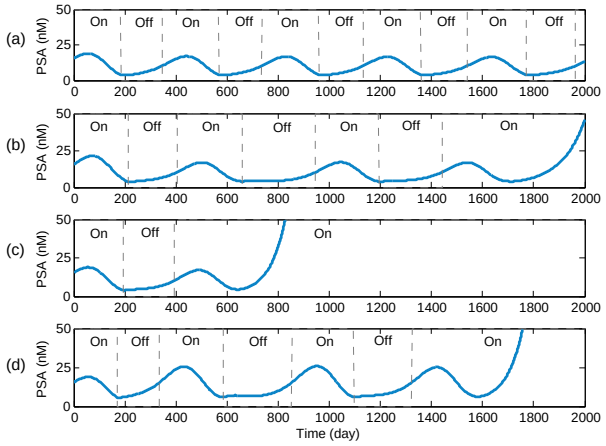

Fig. 3. Simulated PSA profiles of patients with different parameters. (a) Patient#1: $\alpha_y = 0.0242$, $\beta_y = 0.0168$, $m_1 = 0.00005$, $z(0) = 12$, $r_0 = 4$, $r_1 = 10$ (b) Patient#2: $\alpha_y = 0.24$, $\beta_y = 0.13$, $z(0) = 13$, $m_1 = 0.0001$, $r_0 = 4$, $r_1 = 10$ (c) Patient#3: $\alpha_y = 0.35$, $\beta_y = 0.187$, $m_1 = 0.00005$, $z(0) = 10$, $r_0 = 4$, $r_1 = 10$ (d) Patient#3: $\alpha_y = 0.035$, $\beta_y = 0.187$, $m_1 = 0.00005$, $z(0) = 10$, $r_0 = 6$, $r_1 = 15$.

model to the PSA time serials data under the first 1.5 cycles of IAS therapy (data available at http://www.nicholasbruchovsky.com/clinicalResearch.html). Table 2 summarized the suggested treatment scheme for selected patients. The in silico validation results are shown in Supplementary Materials.

Table 2. Personalized hormone therapy scheme for selected patients

Patient ID	α_y	β_y	m_1	$z(0)$	Suggested scheme
#8	0.025	0.021	3.0E-5	8.23	$r_0 = 5.0$, $r_1 = 11.2$
#10	0.019	0.009	5.9E-5	9.44	$r_0 = 4.1$, $r_1 = 9.4$
#45	0.012	0.041	1.0E-5	12.61	$r_0 = 3.8$, $r_1 = 12.2$
#97	0.031	0.015	2.3E-5	10.61	–

3.2 Parameter identification for cardiac disorders

Mathematical modeling the dynamics of cardiac cells is important in understanding the mechanisms of cardiac disorders. [24] has developed an extremely versatile electrical model for cardiac cells, referred as minimum resistor model (MRM), which reproduces experimentally measured characteristics of human ventricular cell dynamics. Identifying the parameter ranges for which the MRM accurately reproduces cardiac abnormalities will benefit the development of the

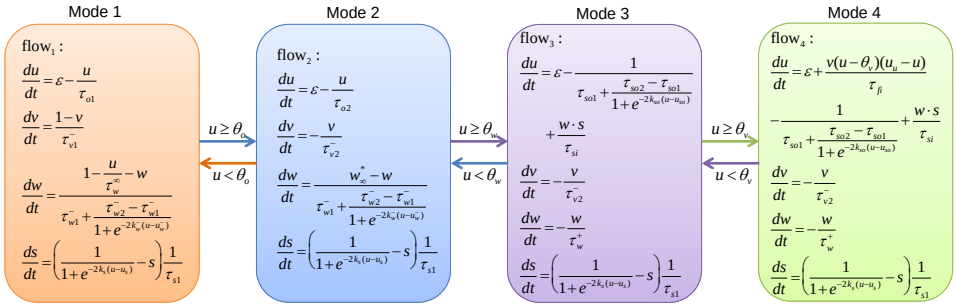

Fig. 4. The minimal resistor model of cardiac cells.

treatment of cardiac disorders. For instance, improper functioning of the cardiac cell ionic channels may cause the cells to lose excitability. Unexcitable cells can induce ventricular tachycardia or fibrillation by blocking propagating electrical waves. In order to identify parameter ranges for which cardiac cells loss excitability, [48] linearized and transformed MRM into a piecewise-multiaffine system called MHA so that parameters synthesis process can be performed using the method proposed in [33]. However, due the the simplification, MRM and MHA have different sets of parameters. In this section, we show how we identify MRM parameter ranges for cardiac disorders without the help of linear approximation.

MRM contains 4 state variables and 26 parameters. An action potential (AP) is a change in the cell's transmembrane potential u, as a response to an external stimulus (current) ϵ. The flow of total currents is controlled by a fast channel gate v and two slow gates w and s. In Mode 1, gates v and w are open and gate s

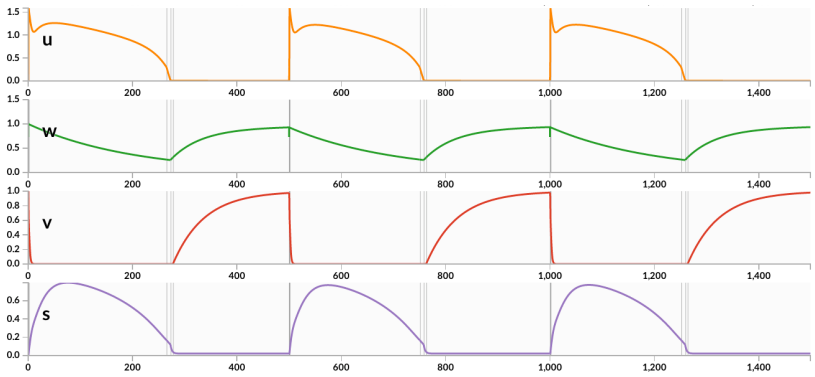

Fig. 5. The simulated time profile of MRM.

is closed. The transmembrane potassium current causes the decay of u. The cell

is resting and waiting for stimulation. We assume external stimulus ϵ equals to 1 and lasts for 1 millisecond. The stimulation causes u increase which may trigger $\mathsf{jump}_{1\to 2} : u \geq \theta_o$. In Mode 2, v starts closing. The decay rate of u changes. The systems will jump to Mode 3 if $u \geq \theta_w$. In Mode 3, w is also closing. u is governed by the potassium current and the calcium current. When $u \geq \theta_v$, Mode 4 can be reached which means a successful AP initiation. In Mode 4, u reaches its peak due to the fast opening of sodium channel. The cardiac muscle contracts and u starts decreasing. Figure 5 shows a witness trace computed by dReal when Mode 4 is reachable. The stimulus ϵ was reset every 500 milliseconds.

When the system can not reach a state in Mode 4, the cardiac cell loses the excitability, which might lead to tachycardia and fibrillation. Starting with Mode 1, we then synthesized parameters using which the system will never go into Mode 4. We obtained the following results (see Supplementary Materials for more details):

$$\tau_{o1} \in (0, 0.006) \vee \tau_{o2} \in (0, 0.13) \vee 6.2 \cdot \tau_{so1} + \tau_{so2} \geq 9.9$$

The results suggest that when $\tau_{o1} \in (0, 0.006)$, the system will always stay in Mode 1. When $\tau_{o2} \in (0, 0.13)$, a state in Mode 3 can not be reached. Furthermore, whether the system can jump from Mode 3 to Mode 4 depends on the interplay between τ_{so1} and τ_{so2}. Figure 6 visualizes these results by showing the simulated time profiles with different parameter values.

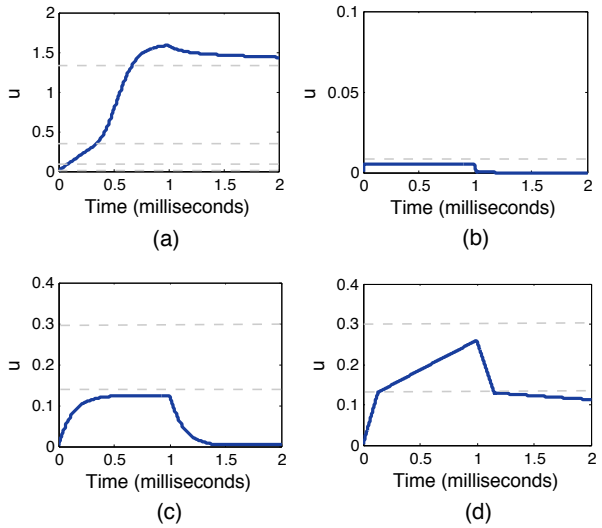

Fig. 6. Simulation results with different parameter settings. (a) Original parameters (b) $\tau_{o1} = 0.0055$ (c) $\tau_{o2} = 0.125$ (d) $\tau_{so1} = 1.2$, $\tau_{so2} = 1.0$

4 Conclusion

Hybrid automata are well-studied formalisms for modeling the behavior of biological systems. In this article, we have presented a framework using δ-complete decision procedures for the parameter identification of hybrid biological systems. We have used the $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-formulas to describe parameterized hybrid automata and encode parameter synthesis problems. Determining the satisfiability of $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-formulas with nonlinear real functions is undecidable. To overcome this, we have employed the δ-decision procedures to perform bounded model checking, and developed an interval constrains propagation based algorithm to obtain the resulting parameters.

We have demonstrated the applicability of our method with the help of two hybrid biological models. In the prostate cancer case study, our method successfully ruled out model candidates which hopelessly fit the experimental observation. We also designed personalized treatment schemes for individual patients. By investigating a highly nonlinear model of the cardiac cell, we have identified critical parameters that can induce cardiac disorders.

Our δ-decisions based parameter synthesis method has the potential to be applied to model classes such as hybrid functional Petri nets models [49]. We plan to explore this in our future work. Another interesting direction will be applying our method to tackling the parameter estimation problem, which is currently one of the most important challenges in systems biology. By properly encoding the noisy web-lab experimental data using logic formulas, bounded model checking can be utilized to estimate the unknown parameter values. In this connection, a model checking based parameter estimation framework presented in [50] promises to offer helpful pointers.

References

1. Liu, B., Thiagarajan, P.: Modeling and analysis of biopathways dynamics. Journal of Bioinformatics and Computational Biology **10**(4) (2012) 1231001
2. Helikar, T., Konvalina, J., Heidel, J., Rogers, J.A.: Emergent decision-making in biological signal transduction networks. Proceedings of the National Academy of Sciences **105**(6) (2008) 1913–1918
3. Gong, H., Zuliani, P., Wang, Q., Clarke, E.: Formal analysis for logical models of pancreatic cancer. In: Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on, IEEE (2011) 4855–4860
4. Karr, J.R., Sanghvi, J.C., Macklin, D.N., Gutschow, M.V., Jacobs, J.M., Bolival, B., Assad-Garcia, N., Glass, J.I., Covert, M.W.: A whole-cell computational model predicts phenotype from genotype. Cell **150**(2) (2012) 389–401
5. Liu, B., Zhang, J., Tan, P.Y., Hsu, D., Blom, A.M., Leong, B., Sethi, S., Ho, B., Ding, J.L., Thiagarajan, P.: A computational and experimental study of the regulatory mechanisms of the complement system. PLoS Computational Biology **7**(1) (2011) e1001059
6. Alberts, B., Johnson, A., Lewis, J., Raff, M., Roberts, K., Walter., P.: Molecular Biology of the Cell. 4 edn. New Tork: Garland Science (2002)

7. Henzinger, T.A.: The theory of hybrid automata. In: Proceedings of the 11th Annual IEEE Symposium on Logic in Computer Science. LICS '96, Washington, DC, USA, IEEE Computer Society (1996) 278–292
8. Ghosh, R., Tomlin, C.: Symbolic reachable set computation of piecewise affine hybrid automata and its application to biological modelling: Delta-notch protein signalling. Systems Biology **1**(1) (2004) 170–183
9. Ye, P., Entcheva, E., Smolka, S., Grosu, R.: Modelling excitable cells using cycle-linear hybrid automata. Systems Biology, IET **2**(1) (2008) 24–32
10. Aihara, K., Suzuki, H.: Theory of hybrid dynamical systems and its applications to biological and medical systems. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences **368**(1930) (2010) 4893–4914
11. Antoniotti, M., Mishra, B., Piazza, C., Policriti, A., Simeoni, M.: Modeling cellular behavior with hybrid automata: Bisimulation and collapsing. In: Computational Methods in Systems Biology, Springer (2003) 57–74
12. Lincoln, P., Tiwari, A.: Symbolic systems biology: Hybrid modeling and analysis of biological networks. In: Hybrid Systems: Computation and Control. Springer (2004) 660–672
13. Baldazzi, V., Monteiro, P.T., Page, M., Ropers, D., Geiselmann, J., De Jong, H.: Qualitative analysis of genetic regulatory networks in bacteria. In: Understanding the Dynamics of Biological Systems. Springer (2011) 111–130
14. Tanaka, G., Hirata, Y., Goldenberg, S.L., Bruchovsky, N., Aihara, K.: Mathematical modelling of prostate cancer growth and its application to hormone therapy. Philosophical Transactions of The Royal Society A: Mathematical, Physical and Engineering Sciences **368** (2010) 5029–5044
15. Girard, A., Le Guernic, C., Maler, O.: Efficient computation of reachable sets of linear time-invariant systems with inputs. In: HSCC'06. (2006) 257–271
16. Frehse, G.: Phaver: Algorithmic verification of hybrid systems past hytech. In: HSCC'05. (2005) 258–273
17. Clarke, E., Fehnker, A., Han, Z., Krogh, B., Stursberg, O., Theobald, M.: Verification of hybrid systems based on counterexample-guided abstraction refinement. In: TACAS'03. (2003) 192–207
18. Alur, R., Henzinger, T.A., Lafferriere, G., Pappas, G.J.: Discrete abstractions of hybrid systems. P. IEEE **88**(7) (2000) 971–984
19. Agrawal, M., Stephan, F., Thiagarajan, P., Yang, S.: Behavioural approximations for restricted linear differential hybrid automata. In: HSCC'06. (2006) 4–18
20. Henzinger, T.A., Kopke, P.W.: Discrete-time control for rectangular hybrid automata. Theor. Comput. Sci. **221**(1) (1999) 369–392
21. Gao, S., Avigad, J., Clarke, E.M.: Delta-complete decision procedures for satisfiability over the reals. In: IJCAR'12, Manchester, UK. (2012) 286–300
22. Gao, S., Avigad, J., Clarke, E.M.: Delta-decidability over the reals. In: LICS'12, Dubrovnik, Croatic. (2012) 305–314
23. Ideta, A.M., Tanaka, G., Takeuchi, T., Aihara, K.: A mathematical model of intermittent androgen suppression for prostate cancer. J. Nonlinear Sci. **18** (2008) 593–614
24. Bueno-Orovio, A., Cherry, E.M., Fenton, F.H.: Minimal model for human ventricular action potentials in tissue. Journal of Theoretical Biology **253** (2008) 544–560
25. Bortolussi, L., Policriti, A.: Hybrid systems and biology. In Bernardo, M., Degano, P., Zavattaro, G., eds.: Formal Methods for Computational Systems Biology. Volume 5016 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2008) 424–448

26. Alur, R.: Formal verification of hybrid systems. In: EMSOFT11, Taipei, Taiwan. (2011) 1–6
27. Clarke, E.M., Grumberg, O., Peled, D.A.: Model Checking. MIT Press (1999)
28. Clarke, E.M., Faeder, J.R., Langmead, C.J., Harris, L.A., Jha, S.K., Legay, A.: Statistical model checking in BioLab: Applications to the automated analysis of T-Cell receptor signaling pathway. In: CMSB'08. (2008) 231–250
29. Chabrier-Rivier, N., Chiaverini, M., Danos, V., Fages, F., Schchter, V.: Modeling and querying biomolecular interaction networks. Theor. Comput. Sci. **325**(1) (sep 2004) 25–44
30. Kwiatkowska, M., Norman, G., Parker, D.: Using probabilistic model checking in systems biology. SIGMETRICS Perform. Eval. Rev. **35**(4) (2008) 14–21
31. Li, C., Nagasaki, M., Koh, C.H., Miyano, S.: Online model checking approach based parameter estimation to a neuronal fate decision simulation model in Caenorhabditis elegans with hybrid functional Petri net with extension. Mol. Biosyst. **11(Suppl 7)**(S10) (2010) 1–13
32. Liu, B., Hagiescu, A., Palaniappan, S.K., Chattopadhyay, B., Cui, Z., Wong, W., Thiagarajan, P.S.: Approximate probabilistic analysis of biopathway dynamics. Bioinformatics **28**(11) (2012) 1508–1516
33. Batt, G., belta, C., Weiss, R.: Temporal logic analysis of gene networks under parameter uncertainty. IEEE T. Automat. Contr. **53** (2008) 215–229
34. Donze, A., Clermont, G., Langmead, C.J.: Parameter synthesis in nonlinear dynamical systems: Application to systems biology. J Comput. Biol. **17**(3) (2010) 325–336
35. Weihrauch, K.: Computable Analysis: An Introduction. Springer (2000)
36. Gao, S., Kong, S., Clarke, E.M.: dReal: An SMT solver for nonlinear theories of reals. In: CADE'13, Lake Placid, USA. (2013) 208–214
37. Siegel, R., Naishadham, D., Jemal, A.: Cancer statistics, 2013. CA. Cancer J. Clin. **63** (2013) 11–30
38. Buchan, N.C., Goldenberg, S.L.: Intermittent androgen suppression for prostate cancer. Nat. Rev. Urol. **7** (2010) 552–560
39. Bruchovsky, N., Goldenberg, S.L., Rennie, P.S., E., G.M.: Theoretical considerations and initial clinical results of intermittent hormone treatment of patients with advanced prostatic carcinoma. Urologe A **34** (1995) 389–392
40. Bruchovsky, N., Klotz, L., Crook, J., Malone, S., Ludgte, C., Morris, W., Gleave, M.E., Goldenberg, S.L., Rennie, P.S.: Final results of the canadian prospective phase II trial of intermittent androgen suppression for men in biochemical recurrence after radiotherapy for locally advanced prostate cancer. Cancer **107** (2006) 389–395
41. Bruchovsky, N., Klotz, L., Crook, J., Malone, S., Ludgte, C., Morris, W., Gleave, M.E., Goldenberg, S.L., Rennie, P.S.: Locally advanced prostate cancerbiochemical results from a prospective phase II study of intermittent androgen suppression for men with evidence of prostate-specific antigen recurrence after radiotherapy. Cancer **109** (2007) 858–867
42. Hamilton, G., Theyer, G.: 13. In: Advances in Prostate Cancer. INTECH (2013) 305–330
43. Jackson, T.L.: A mathematical model of prostate tumor growth and androgen-independent relapse. Discrete Contin. Dyn. Syst. Ser. B **4** (2004) 187–201
44. Hirata, Y., di Bernordo, M., Bruchovsky, N., Aihara, K.: Hybrid optimal scheduling for intermittent androgen suppression of prostate cancer. Chaos **20**(4) (2010) 045125

45. Suzuki, T., Bruchovsky, N., K., A.: Piecewise affine systems modelling for optimizing hormone therapy of prostate cancer. Philos. Trans. A Math. Phys. Eng. Sci. **368**(1930) (2010) 5045–5059
46. Berges, R., Vukanovic, J., Epstein, J., CarMichel, M., Cisek, L., Johnson, D., Veltri, R., Walsh, P., Isaacs, J.: Implication of cell kinetic changes during the progression of human prostatic cancer. Clin. Cancer Res. **1**(5) (1995) 473–480
47. Hirata, Y., Bruchovsky, N., Aihara, K.: Development of a mathematical model that predicts the outcome of hormone therapy for prostate cancer. J. Theor. Biol. **264** (2010) 517–527
48. Grosu, R., Batt, G., Fenton, F.H., Gilmm, J., Guernic, C.L., Smolka, S.A., Bartocci, E.: From cardiac cells to genetic regulatory networks. In: Computer Aided Verification, Springer (2011) 1–15
49. Matsuno, H., Tanaka, Y., Aoshima, H., Doi, A., Matsui, M., Miyano, S.: Biopathways representation and simulation on hybrid functional petri net. In Silico Biol. **3**(3) (2003) 389–404
50. Palaniappan, S., Gyori, B., Liu, B., Hsu, D., Thiagarajan, P.S.: Statistical model checking based calibration and analysis of bio-pathway models. In: CMSB'13, Klosterneuburg, Austria. (2013) 120–134