Generating Box Invariants*

Ashish Tiwari

SRI International,
333 Ravenswood Ave,
Menlo Park, CA, U.S.A
tiwari@csl.sri.coma
Tel/Fax:+1.650.859.4774/2844

Abstract. Box invariant sets are box-shaped positively invariant sets. We show that box invariants are computable for a large class of nonlinear and hybrid systems. The technique for computing these invariants is based on nonlinear constraint solving. This paper also shows that the class of multiaffine systems, which has been used successfully for modeling and analyzing regulatory and biochemical reaction networks, can be generalized to the class of componentwise monotone and componentwise quasi monotone systems without losing any of its nice properties.

Introduction

A *positively invariant* set is a subset of the state space of a dynamical system with the property that, if the system state is in this set at some time, then it will stay in this set in the future [1].[1] A rectangular box, $Box(\boldsymbol{l}, \boldsymbol{u})$, specified using two diagonally opposite points $\boldsymbol{l}$ and $\boldsymbol{u}$ in $\Re^n$, where $\boldsymbol{l} < \boldsymbol{u}$ (interpreted componentwise), and its vertices and faces are defined as follows.[2]

$$\begin{aligned}
Box(\boldsymbol{l}, \boldsymbol{u}) &= \{\boldsymbol{x} \in \Re^n \mid \boldsymbol{l}_i \leq \boldsymbol{x}_i \leq \boldsymbol{u}_i, \text{ for all } i\} \\
Vert(\boldsymbol{l}, \boldsymbol{u}) &= \{\boldsymbol{x} \in \Re^n \mid \boldsymbol{x}_i = \boldsymbol{l}_i \text{ or } \boldsymbol{x}_i = \boldsymbol{u}_i, \text{ for all } i\} \\
Faces(\boldsymbol{l}, \boldsymbol{u}) &= \bigcup_{j=1}^{n} (L^j(\boldsymbol{l}, \boldsymbol{u}) \cup U^j(\boldsymbol{l}, \boldsymbol{u})) \\
L^j(\boldsymbol{l}, \boldsymbol{u}) &= \{\boldsymbol{x} \in Box(\boldsymbol{l}, \boldsymbol{u}) \mid \boldsymbol{x}_j = \boldsymbol{l}_j\} \\
U^j(\boldsymbol{l}, \boldsymbol{u}) &= \{\boldsymbol{x} \in Box(\boldsymbol{l}, \boldsymbol{u}) \mid \boldsymbol{x}_j = \boldsymbol{u}_j\}
\end{aligned}$$

* Research supported in part by the National Science Foundation under grant CNS-0720721.

1 A positively invariant set, as defined above, is called an *inductive* property in computer science terminology. An invariant, in computer science, is a subset of the state space that is a superset of the set of all reachable states.

2 With some additional work, most of the results described in this paper can be adapted to the case when we consider boxes that are unbounded in certain coordinates (that is, we have $-\infty$ and $+\infty$ as bounds on some components) or degenerate in some coordinates (that is, certain components are fixed to have a constant value). In this paper, we do not consider these other cases to keep the presentation simple.

We are interested in the case when $Box(\boldsymbol{l}, \boldsymbol{u})$ is a positively invariant set [2, 3]. We say a hybrid system is box invariant if there exists a box that is also a positively invariant set. This is formally defined below.

Definition 1. *A hybrid system HS is said to be box invariant if there exists a finite rectangular box, $Box(\boldsymbol{l}, \boldsymbol{u})$, such that*
(a) for each mode q with continuous dynamics $\dot{\boldsymbol{x}} = \boldsymbol{f}^q(\boldsymbol{x})$ and invariant $Inv(q)$, for any point $\boldsymbol{y} \in Faces(\boldsymbol{l}, \boldsymbol{u})$, it is the case that, for all j, $\boldsymbol{f}^q_j(\boldsymbol{y}) \geq 0$ whenever $\boldsymbol{y} \in L^j(\boldsymbol{l}, \boldsymbol{u}) \cap Inv(q)$ and $\boldsymbol{f}^q_j(\boldsymbol{y}) \leq 0$ whenever $\boldsymbol{y} \in U^j(\boldsymbol{l}, \boldsymbol{u}) \cap Inv(q)$, and
(b) for each discrete transition from mode q to q' — with guard $G(q, q') \subseteq \Re^n$ and a reset map R that resets state $\boldsymbol{y}$ to some state in $R(q, q', \boldsymbol{y}) \subseteq \Re^n$ — and for each point $\boldsymbol{y}$ in $Box(\boldsymbol{l}, \boldsymbol{y})$ that satisfies the invariant, $\boldsymbol{y} \in Inv(q)$, and the guard, $\boldsymbol{y} \in G(q, q')$, if $\boldsymbol{y'} \in R(q, q', \boldsymbol{y})$ then $\boldsymbol{y'} \in Box(\boldsymbol{l}, \boldsymbol{u})$.

Note that the above definition requires that (a) each of the constituent continuous dynamical system in the hybrid system be box invariant with respect to the *same* box, and (b) starting from any point in the box, any reset (on a discrete transition) can only result in a point that is also in the same box. The motivation for our interest in a *single global* positively invariant box is that the hybrid systems of interest to us are approximations of high-dimensional continuous dynamical systems and we are interested in positively invariant sets for the original system.

The above definition can be extended to systems with inputs $\boldsymbol{u}$ by treating $\boldsymbol{u}$ as state variables whose derivative is 0. It corresponds to requiring that the state variables be bounded assuming the inputs are bounded.

It is easy to see that a box invariant set, as defined above, is indeed a positively invariant set for the hybrid system. Although we have defined the concept in its generality, all examples in this paper are restricted to either continuous dynamical systems or hybrid systems with identity resets. Note that Condition (b) of Definition 1 becomes trivial when we have only identity reset maps.

Related Work Computational results on box invariance of linear systems [3] and some preliminary results for nonlinear and hybrid systems [2] have been presented before. This paper develops these results further and identifies the classes of monotone, quasi-monotone, and uniformly quasi-monotone systems on which box invariants computation can be reduced to constraint solving. Sankaranarayanan et. al. [8] used constraint solving to search for invariants of a given *form*. Our work here is a specialization to box-shaped invariants, and develops the necessary and sufficient approaches for this special case. The "barrier certificates" proposed by Prajna, Jadbabaie and Pappas [9–12] are also essentially inductive invariants. Tiwari [13] generated linear inductive invariants for linear systems and techniques for computing inductive invariants for nonlinear systems were suggested by Tiwari and Khanna [14]. Rodriguez-Carbonell and Tiwari [15] showed that the best (strongest) possible polynomial equational invariant was computable for hybrid systems with linear dynamics in each mode. Pappas et al. have also considered the problem of computing invariants, but only for linear systems, using interesting techniques [16, 17]. In contrast to all these works, the work in this paper is focused on a very simple form of invariant. Our goal here is to maintain efficiency and scalability, while compromising on the generality of the form of invariants.

Specialized forms of the notion of box invariance have been studied previously in the literature in the form of componentwise asymptotic stability [4, 5] and Lyapunov stability under the infinity vector norms [6, 7], but this paper differs in two significant ways: it considers the *computational* aspects of box invariance and focuses on *nonlinear* systems.

Polynomial Hybrid Systems

In polynomial hybrid systems, the dynamics are specified using polynomials over the state variables and the guards, invariants, and resets are specified using semi-algebraic sets. For such systems, the conditions in Definition 1 can be written as a formula in the first-order theory of reals

$$\exists \boldsymbol{l}, \boldsymbol{u}.\forall q.\forall \boldsymbol{x}. \bigwedge_{1\leq j\leq n} ((\boldsymbol{x} \in L^j \wedge \boldsymbol{x} \in \mathit{Inv}(q) \Rightarrow \boldsymbol{p}_j^q(\boldsymbol{x}) \geq 0) \;\wedge \\ (\boldsymbol{x} \in U^j \wedge \boldsymbol{x} \in \mathit{Inv}(q) \Rightarrow \boldsymbol{p}_j^q(\boldsymbol{x}) \leq 0)), \qquad (1)$$

where $\boldsymbol{p}^q$ specifies the dynamics in mode q. In the presence of resets, we need additional formulas to express that resets do not take the dynamics out of the box. This is also expressible in the first-order theory of reals (assuming that the invariants, guards, and resets are specified using polynomials.) Since the first-order theory of reals is decidable [18, 19], the following result follows.

Theorem 1. *Box invariance of polynomial hybrid systems is decidable.* □

While this is a useful theoretical result, it is not very practical due to the high complexity of the decision procedure for real-closed fields. We specialize the above result to a subclass of polynomial systems that is a generalization of the class of multiaffine systems.

Monotone Systems

A function $f : \Re \mapsto \Re$ is *monotonically increasing* if $f(x) \leq f(x')$ whenever $x < x'$, and $f(x)$ is *monotonically decreasing* if $f(x) \geq f(x')$ whenever $x < x'$. A function $f(x_1, \ldots, x_n)$ is said to be *monotonic with respect to* x_i if for every choice $c_1, \ldots, c_n$ of values for the variables, the function $f(c_1, \ldots, c_{i-1}, x_i, c_{i+1}, \ldots, c_n)$ if either monotonically increasing or monotonically decreasing. For example, the function $x_1x_3 - x_2x_3$ is monotonic with respect to x_3 since if we fix the values c_1, c_2 (for x_1, x_2 respectively), we notice that the function $c_1x_3 - c_2x_3$ will always be either monotonically increasing (if $c_1 - c_2 \geq 0$) or monotonically decreasing (if $c_1 - c_2 \leq 0$).

A system $\dot{\boldsymbol{x}} = \boldsymbol{p}(\boldsymbol{x})$ is *monotone* if each function $\boldsymbol{p}_i$ is monotonic with respect to each variable x_j. Note that every multiaffine system [20, 21] is also monotone.

The converse is not true; for example, the system $dx_1/dt = x_1^3 + x_1$ is monotone but not multiaffine.

Monotone systems not only generalize multiaffine systems, but also inherit some of their nice properties that have been used to build powerful analysis tools and techniques for analysis of multiaffine systems [22]. In particular, the following variant of Corollary 1 from Kloetzer and Belta [22] holds for monotone functions.

Proposition 1. *If $f : \Re^n \mapsto \Re$ is a function that is monotonic (with respect to all of its argument variables) and $Box(\boldsymbol{l}, \boldsymbol{u})$ is a box defined by the diagonally opposite points $\boldsymbol{l}$ and $\boldsymbol{u}$, then for any point $\boldsymbol{c} \in Box(\boldsymbol{l}, \boldsymbol{u})$, we have*

$$\min(\{f(\boldsymbol{x}) \mid \boldsymbol{x} \in Vert(\boldsymbol{l}, \boldsymbol{u})\}) \leq f(\boldsymbol{c}) \leq \max(\{f(\boldsymbol{x}) \mid \boldsymbol{x} \in Vert(\boldsymbol{l}, \boldsymbol{u})\}).$$

Consequently, $f(\boldsymbol{x}) \sim 0$ everywhere in $Box(\boldsymbol{l}, \boldsymbol{u})$ if and only if $f(\boldsymbol{x}) \sim 0$ for all vertices $\boldsymbol{x} \in Vert(\boldsymbol{l}, \boldsymbol{u})$, where $\sim \in \{=, \leq, \geq\}$.

Quasi Monotone Systems

We further generalize the class of monotone systems and call a system $\dot{\boldsymbol{x}} = \boldsymbol{p}(\boldsymbol{x})$ *quasi monotone* if each function $\boldsymbol{p}_i$ is monotonic with respect to variable x_j *for all $j \neq i$*. In other words, we drop the requirement that $\boldsymbol{p}_i$ be monotone with respect to x_i. A *hybrid quasi monotone* system is a hybrid system in which each constituent mode is a quasi monotone system. Every monotone system is naturally also quasi monotone. The system over variable x_1 defined by $\frac{dx_1}{dt} = 1 - x_1^2$ is quasi monotone but it is not monotone (and not multiaffine).

Recall that box invariance of a polynomial system $\dot{\boldsymbol{x}} = \boldsymbol{p}(\boldsymbol{x})$ can be reduced to deciding (the satisfiability of) Formula **??**. Formula **??** is a logical statement for the fact that the vector field points “inwards” on all faces of the box. Consider one of the faces, say L^j. The requirement is that $\boldsymbol{p}_j(\boldsymbol{x}) \geq 0$ for all points $\boldsymbol{x}$ on the face L^j. In a quasi monotone system, the function $\boldsymbol{p}_j(\boldsymbol{x})$ is monotonic with respect to all variables x_i for $i \neq j$. Once we fix x_j to a_j the function $\boldsymbol{p}_j(x_1, \ldots, x_{j-1}, a_j, x_{j+1}, \ldots, x_n)$ is monotonic with respect to all its variables. Hence, we can use Proposition 1 and conclude that we only need to check that $\boldsymbol{p}_j$ is non-negative on the vertices of the face L^j. Using the same argument for each face, we conclude the following.

Proposition 2. *A quasi monotone system $\dot{\boldsymbol{x}} = \boldsymbol{p}(\boldsymbol{x}), \boldsymbol{x} \in \Re^n$ is box invariant iff there exist two points $\boldsymbol{l}, \boldsymbol{u} \in \Re^n$ such that for each point $\boldsymbol{c} \in Vert(\boldsymbol{l}, \boldsymbol{u})$, we have $\boldsymbol{p}_j(\boldsymbol{c}) \leq 0$ if $c_j = u_j$ and $\boldsymbol{p}_j(\boldsymbol{c}) \geq 0$ if $c_j = l_j$ for all j; that is,*

$$\exists \boldsymbol{l}, \boldsymbol{u}. \bigwedge_{\boldsymbol{c} \in Vert(\boldsymbol{l}, \boldsymbol{u}), 1 \leq j \leq n} \alpha_j(\boldsymbol{c}) \boldsymbol{p}_j(\boldsymbol{c}) \geq 0, \qquad (2)$$

where $\alpha_j(\boldsymbol{c}) = 1$ if $c_j = l_j$ and $\alpha_j(\boldsymbol{c}) = -1$ if $c_j = u_j$. □

Formula **??** had both existential and universal quantifiers. Quasi-monotonicity has allowed us to eliminate the universal quantifier and obtain simply a conjunction of $n2^n$ (existentially quantified) constraints shown in Formula 2. Any constraint solving engine that can handle nonlinear constraints can now be used (and we do not necessarily need a quantifier elimination procedure).

The test for box invariance of a hybrid quasi monotone system with no resets simply involves putting together these $n2^n$ constraints – guarded by the mode invariants – for each of the modes and solving them simultaneously. This is expressed in the following formula, which is again a existentially quantified formula with no universal quantifiers.

$$\exists \boldsymbol{l}, \boldsymbol{u}. \bigwedge_{q \in Q, \boldsymbol{c} \in Vert(\boldsymbol{l},\boldsymbol{u}), 1 \leq j \leq n} (\boldsymbol{c} \in Inv(q) \Rightarrow \alpha_j(\boldsymbol{c}) \boldsymbol{p}^q_j(\boldsymbol{c}) \geq 0), \tag{3}$$

where α_j and $\boldsymbol{p}^q$ are defined as before.

Quasi Uniformly Monotone Systems

Proposition 2 still requires checking satisfiability of an *exponential* number of (nonlinear) constraints. However, for a very useful subclass of quasi monotone systems, we can reduce the number of constraints (from $n2^n$) to $2n$. We use the notion of *uniform* monotonicity. A function $f : \mathbb{R}^n \mapsto \mathbb{R}$ is *uniformly monotonic with respect to a variable* x_j *in domain Inv* if for all points $\boldsymbol{x} \in Inv$ and $\boldsymbol{x'} \in Inv$ that differ only in the j-th component, $f(\boldsymbol{x}) \leq f(\boldsymbol{x'})$ (or $f(\boldsymbol{x}) \geq f(\boldsymbol{x'})$) whenever $\boldsymbol{x}_j < \boldsymbol{x}'_j$; that is,

$$\forall \boldsymbol{x}, \boldsymbol{x'} \in Inv. (\bigwedge_{i \neq j} \boldsymbol{x}_i = \boldsymbol{x'}_i \ \wedge \ \boldsymbol{x}_j = \boldsymbol{x'}_j \Rightarrow f(\boldsymbol{x}) \leq f(\boldsymbol{x'})), \text{ or,}$$

$$\forall \boldsymbol{x}, \boldsymbol{x'} \in Inv. (\bigwedge_{i \neq j} \boldsymbol{x}_i = \boldsymbol{x'}_i \ \wedge \ \boldsymbol{x}_j = \boldsymbol{x'}_j \Rightarrow f(\boldsymbol{x}) \geq f(\boldsymbol{x'})).$$

Note that the definition of uniform monotonicity with respect to x_j requires that f be monotonic in the *same* way (that is, either increasing or decreasing) across *all* choices of values for other variables. For example, $x_1x_3 - x_2x_3$ is not uniformly monotonic with respect to x_3, whereas it is monotonic with respect to x_3. However, $x_1x_3 - x_2x_3$ is uniformly monotonic with respect to x_1 in the domain $Inv := \{x_1 \geq 0, x_2 \geq 0, x_3 \geq 0\}$. A system $\dot{\boldsymbol{x}} = \boldsymbol{p}(\boldsymbol{x})$ is said to be a *quasi uniformly monotone system* in the domain *Inv* if, for each i, $\boldsymbol{p}_i$ is uniformly monotonic with respect to x_j in the domain *Inv* for each $j \neq i$.

Proposition 3. *Let* $\dot{\boldsymbol{x}} = \boldsymbol{p}(\boldsymbol{x})$ *be a quasi uniformly monotonic system in the domain Inv such that* $Box(\boldsymbol{l}, \boldsymbol{u}) \subseteq Inv$. *Then, the* $n2^n$ *constraints in Formula 2 of Proposition 2 are equivalent to a subset of* $2n$ *constraints.* □

Proof. Consider the fact $L^j(\boldsymbol{l}, \boldsymbol{u})$ and the corresponding 2^{n-1} constraints:

$$\bigwedge_{\boldsymbol{c} \in Vert(\boldsymbol{l},\boldsymbol{u}), \boldsymbol{c}_j = l_j} \boldsymbol{p}_j(\boldsymbol{c}) \geq 0$$

Consider the single constraint $\boldsymbol{p}_j(\boldsymbol{d}) \geq 0$ where for all i, $d_i = l_i$ if either $i = j$ or $\boldsymbol{p}_j$ is uniformly increasing with respect to x_i and $d_i = u_i$ if $i \neq j$ and $\boldsymbol{p}_j$ is uniformly decreasing with respect to x_i. It is easy to see that $\boldsymbol{p}_j(\boldsymbol{d}) = \min(\{\boldsymbol{p}_j(\boldsymbol{c}) \mid \boldsymbol{c} \in \mathit{Vert}(\boldsymbol{l}, \boldsymbol{u}), \boldsymbol{c}_j = l_j\})$ and hence the single constraint $\boldsymbol{p}_j(\boldsymbol{d}) \geq 0$ subsumes all the 2^{n-1} constraints given above.

Example 1. Consider the following Phytoplankton Growth Model (see [23] and references therein):

$$\dot{x}_1 = 1 - x_1 - \tfrac{x_1 x_2}{4}, \quad \dot{x}_2 = (2x_3 - 1)x_2, \quad \dot{x}_3 = \tfrac{x_1}{4} - 2x_3^2,$$

where x_1 denotes the substrate, x_2 the phytoplankton biomass, and x_3 the intracellular nutrient per biomass. This system is not multi-affine in the sense of [20] and it is not monotonic, but it is monotonic over the domain $\mathit{Inv} := \{x_1 \geq 0, x_2 \geq 0, x_3 \geq 0\}$. It is quasi monotonic and even quasi uniformly monotonic. Hence, by Proposition 2, its box invariance is equivalent to the existence of $\boldsymbol{l}, \boldsymbol{u}$ s.t. $n2^n = 3 \cdot 2^3 = 24$ constraints are satisfied. Moreover, by Proposition 3, these 24 constraints are subsumed by the following 6 constraints:

$$\begin{array}{lll} 1 - u_1 - \tfrac{u_1 l_2}{4} \leq 0, & u_2(2u_3 - 1) \leq 0, & \tfrac{u_1}{4} - 2u_3^2 \leq 0, \\ 1 - l_1 - \tfrac{l_1 u_2}{4} \geq 0, & l_2(2l_3 - 1) \geq 0, & \tfrac{l_1}{4} - 2l_3^2 \geq 0. \end{array}$$

One possible solution for these constraints is given by $\boldsymbol{l} = (0,0,0)$ and $\boldsymbol{u} = (2,1,1/2)$ indicating that the box formed by these two points as diagonally opposite vertices is a positive invariant set. □

References

1. Blanchini, F.: Set invariance in control. Automatica **35** (1999) 1747–1767
2. Abate, A., Tiwari, A.: Box invariance of hybrid and switched systems. In: 2nd IFAC Conf. on Analysis and Design of Hybrid Systems, ADHS. (2006) 359–364
3. Abate, A., Tiwari, A.: Box invariance for biologically-inspired dynamical systems. In: Proc. IEEE Conf. on Decision and Control, CDC. (2007) To appear.
4. Voicu, M.: Componentwise aymptotic stability of linear constant dynamical systems. IEEE Transactions on Automatic Control **29**(10) (October 1984)
5. Pastravanu, O., Voicu, M.: Necessary and sufficient conditions for componentwise stability of interval matrix systems. IEEE Transactions on Automatic Control **49**(6) (June 2004)
6. Kiendl, H., Adamy, J., Stelzner, P.: Vector norms as Lyapunov functions for linear systems. IEEE Transactions on Automatic Control **37**(6) (June 1992)
7. Pastravanu, O., Voicu, M.: Norm-based approach to componentwise aymptotic stability. In: 11th IEEE Mediterranean Conf. on Control and Automation. (2003)
8. Sankaranarayanan, S., Sipma, H., Manna, Z.: Constructing invariants for hybrid systems. In: Hybrid Systems: Computation and Control, HSCC 2004. Volume 2993 of LNCS., Springer (2004) 539–554
9. Prajna, S.: Barrier certificates for nonlinear model validation. In: Proc. IEEE Conference on Decision and Control. (2003)

10. Prajna, S., Jadbabaie, A.: Safety verification of hybrid systems using barrier certificates. In: Hybrid Systems: Computation and Control, HSCC 2004. Volume 2993 of LNCS., Springer (2004) 477–492
11. Prajna, S., Jadbabaie, A., Pappas, G.J.: Stochastic safety verification using barrier certificates. In: Proc. 43rd IEEE Conf. on Decision and Control (CDC). (2004)
12. Prajna, S., Jadbabaie, A., Pappas, G.J.: A framework for worst-case and stochastic safety verification using barrier certificates. IEEE Transactions on Automatic Control (March 2005)
13. Tiwari, A.: Approximate reachability for linear systems. In Maler, O., Pnueli, A., eds.: HSCC. Volume 2623 of LNCS., Springer (April 2003) 514–525
14. Tiwari, A., Khanna, G.: Nonlinear Systems: Approximating reach sets. In: HSCC. Volume 2993 of LNCS., Springer (March 2004) 600–614
15. Rodriguez-Carbonell, E., Tiwari, A.: Generating polynomial invariants for hybrid systems. In: HSCC. Volume 3414 of LNCS., Springer (2005) 590–605
16. Yazarel, H., Pappas, G.J.: Geometric programming relaxations for linear system reachability. In: Proc. 2004 American Control Conference. (2004)
17. Yazarel, H., Prajna, S., Pappas, G.J.: S.O.S. for safety. In: Proc. 43rd IEEE Conference on Decision and Control. (2004)
18. Tarski, A.: A Decision Method for Elementary Algebra and Geometry. University of California Press (1948) Second edition.
19. Collins, G.E.: Quantifier elimination for the elementary theory of real closed fields by cylindrical algebraic decomposition. In: Proc. 2nd GI Conf. Automata Theory and Formal Languages. Volume 33 of LNCS., Springer (1975) 134–183
20. Belta, C., Habets, L., Kumar, V.: Control of multi-affine systems on rectangles with applications to hybrid biomolecular networks. In: Proc. 41st Conf. on Decision and Control, IEEE (2002) 534–539
21. Lincoln, P., Tiwari, A.: Symbolic systems biology: Hybrid modeling and analysis of biological networks. In: Hybrid Systems: Computation and Control. LNCS 2993. Springer (2004) 660–672
22. Kloetzer, M., Belta, C.: A fully automated framework for control of linear systems from ltl specifications. In: Proc. 9th Intl. Workshop on Hybrid Systems: Computation and Control. Volume 3927 of LNCS., Springer (2006) 333–347
23. Bernard, O., Gouze, J.L.: Global qualitative description of a class of nonlinear dynamical systems. Artificial Intelligence **136** (2002) 29–59
24. Julius, A., Halasz, A., Kumar, V., Pappas, G.: Controlling biological systems: the lactose regulation system of Escherichia Coli. In: American Control Conference 2007. (2007)
25. Sorensen, J.T.: A physiologic model of glucose metabolism in man and its use to design and assess improved insulin therapies for diabetes. PhD thesis, Dept. Chem. Eng., Massachusetts Inst. Technology (MIT), Cambridge (1985)
26. Parker, R.S., Doyle, F.J., Peppas, N.A.: A model-based algorithm for blood glucose control in type I diabetes patients. IEEE transactions on biomedical engineering **46**(2) (February 1999)