

DETECTING AND ENFORCING MONOTONICITY FOR HYBRID CONTROL SYSTEMS SYNTHESIS¹

Dmitry Gromov* Jörg Raisch*,**

* *Fachgebiet Regelungssysteme*

Technische Universität Berlin

Email: {gromov|raisch}@control.tu-berlin.de

** *Systems and Control Theory Group*

Max-Planck-Institut, Magdeburg, Germany

Abstract: Abstraction based approaches to control of hybrid systems require efficient means of computing outer approximations of reachable continuous state sets. This contribution discusses how the concept of monotonicity can be used for this purpose. It provides an efficient algorithm to check whether a given continuous system is monotone with respect to a (a-priori unknown) partial order and, if not, investigates how to use continuous feedback to enforce monotonicity. In the latter case, the resulting continuous feedback represents a (lower) control level within a hierarchical hybrid control system. *Copyright © 2006 IFAC*

Keywords: Monotone systems, hybrid control, hierarchical control.

1. INTRODUCTION

Abstraction-based approaches to hybrid control systems synthesis have become popular during the last decade (e.g. [Cury et al., 1998, Koutsoukos and Antsaklis, 2003, Chutinan and Krogh, 2000, Lemch and Caines, 1999, Moor and Raisch, 1999]). They essentially “replace” continuous dynamics by discrete abstractions and hence convert the underlying hybrid control problem into a purely discrete one, which can subsequently be addressed using standard methods from discrete event systems (DES) theory. To guarantee that desired closed-loop properties carry over from the approximation level to the underlying hybrid system, one needs to make sure that the behaviour of the abstraction covers the behaviour of the continuous dynamics on a suitable (discrete) external signal space. This, in turn, boils down

to computing guaranteed overapproximations for reachability sets in the continuous component’s state space. For general nonlinear systems, this represents a highly nontrivial problem. However, if the system under consideration is monotone (e.g. [Smith, 1995, Angeli and Sontag, 2003]) with respect to a partial order in its state space, this becomes a straightforward exercise [Moor and Raisch, 2002]. This paper addresses two problems from this context: (i) it provides a mechanism to efficiently check for the existence of a suitable partial order and hence for monotonicity. (ii) for the case when this test fails, the paper also discusses how continuous feedback can be used to enforce monotonicity.

This contribution is organised as follows: in Section 2, we briefly review the notion of a partial order. Section 3 addresses the concept of monotone, i.e., order preserving, dynamical systems, both for the autonomous and the controlled case. This section is mostly based on [Smith, 1995, Angeli

¹ Work partially done in the framework of the HYCON Network of Excellence, contract number FP6-IST-511368

and Sontag, 2003], but also contains new results on how to efficiently check monotonicity (Propositions 3 and 8). In Section 4, we briefly outline how monotonicity can be used in the context of abstraction based hybrid control synthesis. Finally, in Section 5, we investigate how appropriate continuous feedback on a lower level of a hierarchical hybrid control scheme can enforce monotonicity and hence facilitate the computation of discrete abstractions for higher level control purposes.

2. PARTIAL ORDER RELATIONS

A partial order relation \preceq on a Banach (or, more precisely, ordered metric) space B is defined as an operation satisfying the following three properties:

1. $x \preceq x \quad \forall x \in X$,
2. $(x \preceq y) \wedge (y \preceq z) \Rightarrow x \preceq z \quad \forall x, y, z \in X$,
3. $(x \preceq y) \wedge (y \preceq x) \Rightarrow x = y \quad \forall x, y \in X$.

We write $x \prec y$ if $x \preceq y$ and $x \neq y$. This relation is no longer reflexive and is referred to as a strict order relation. Usually, to introduce an order relation one uses an auxiliary set $K \subset B$, such that

1. $\alpha k \in K \quad \forall k \in K, \alpha \in \mathbb{R}_+$,
2. $k_1 + k_2 \in K \quad \forall k_1, k_2 \in K$,
3. $k \in K \wedge -k \in K \Rightarrow k = 0$.

Thus, K is a convex pointed cone. The relation is defined by $x \preceq y$ if and only if $y - x \in K$.² If K has nonempty interior $\text{int}K$ then we define $x \prec\prec y$ iff $y - x \in \text{int}K$. It is stronger than \prec or \preceq as $x \prec\prec y$ implies $x \prec y$ and therefore $x \preceq y$. In Euclidean space \mathbb{R}^n , orthants can play the role of cones. Each orthant $\mathbb{R}_\delta^n \subset \mathbb{R}^n$ is characterised by its signature, i.e. the n -tuple $\delta = \{\delta_1, \dots, \delta_n\}$ whose elements take values from the two-element set $\{0, 1\}$. \mathbb{R}_δ^n is defined as $\mathbb{R}_\delta^n = \{x \in \mathbb{R}^n | (-1)^{\delta_i} x_i \geq 0\}$. Hence, the zero signature corresponds to the positive orthant. We use notation \preceq_δ (resp., \prec_δ and $\prec\prec_\delta$) to show that the corresponding relation is defined with respect to the orthant \mathbb{R}_δ^n . Relation symbols without index refer to relations w.r.t. the positive orthant.

3. MONOTONE DYNAMICAL SYSTEMS

A monotone dynamical system is a dynamical system on an ordered metric space which has the property that ordered states remain ordered when time progresses. In other words, monotone systems are order preserving dynamical systems. In this section we give some conditions for an arbitrary autonomous dynamical system to be monotone. Furthermore, these results are extended to dynamical systems with inputs.

² $x \prec y$ iff $y - x \in K \setminus \{0\}$

3.1 Autonomous systems

Consider the dynamical system:

$$\dot{x}(t) = f(x(t)), \quad (1)$$

where $x(t) \in X \subset \mathbb{R}^n$, $f : X \rightarrow \mathbb{R}^n$ is a continuously differentiable vector field. The solution of (1) that starts at the point x_0 at $t = 0$ is defined as $\phi_t(x_0)$ and referred to as the flow of (1). To make an assertion about qualitative properties of the above dynamical system we have to introduce some classification.

Definition 1. A vector field $f : X \rightarrow \mathbb{R}^n$ is said to be of type K_δ on an open subset $D \subset X$ if for each $i \in \{1, \dots, n\}$, $(-1)^{\delta_i} f_i(a) \leq (-1)^{\delta_i} f_i(b)$ for any two points a and b in D satisfying $a \preceq_\delta b$ and $a_i = b_i$.

The following proposition ([Smith, 1995], Chapt.3, Prop.5.1) asserts that the type K_δ condition is necessary and sufficient for the order preserving property to hold.

Proposition 2. Let f be of type K_δ on D and $x_0, y_0 \in D$. If $x_0 \preceq_\delta y_0$ (resp., $x_0 \prec_\delta y_0$ or $x_0 \prec\prec_\delta y_0$), $t > 0$, and if $\phi_t(x_0)$ and $\phi_t(y_0)$ are defined and in D , then $\phi_t(x_0) \preceq_\delta \phi_t(y_0)$ (resp., $\phi_t(x_0) \prec_\delta \phi_t(y_0)$ or $\phi_t(x_0) \prec\prec_\delta \phi_t(y_0)$).

The most natural way to decide whether a vector field f is of type K_δ is to analyse the sign structure of the Jacobian matrix of f . More specifically, it can be shown ([Smith, 1995]) that the vector field $f(x)$ is of type K_δ on the convex subset D if and only if

$$(-1)^{\delta_i + \delta_j} \frac{\partial f_i}{\partial x_j}(x) \geq 0, \quad i \neq j, x \in D. \quad (2)$$

Condition (2) can be checked in two steps: **Step 1:** Check whether the off-diagonal elements of the Jacobian matrix are *sign-stable*, i.e.

$$\left\{ \frac{\partial f_i(x)}{\partial x_j} \geq 0 \quad \forall x \in D \right\} \vee \left\{ \frac{\partial f_i(x)}{\partial x_j} \leq 0 \quad \forall x \in D \right\} \quad (3)$$

and *sign-symmetric*, i.e.

$$\frac{\partial f_i(x)}{\partial x_j} \cdot \frac{\partial f_j(x)}{\partial x_i} \geq 0 \quad \forall x \in D \quad (4)$$

for all $i, j \in \{1, \dots, n\}$ such that $i \neq j$.

Step 2: If the tests in Step 1 are satisfied, we need to check whether the (Boolean) equalities

$$\delta_i \oplus \delta_j = s_{ij}, \quad i < j \quad (5)$$

hold, where \oplus represents “exclusive OR” and the $n(n-1)/2$ variables s_{ij} , $i < j$, $j = 2, \dots, n$ are defined as follows:

$$s_{ij} = \begin{cases} 0 & \text{if } \frac{\partial f_i(x)}{\partial x_j} > 0 \vee \left(\frac{\partial f_i(x)}{\partial x_j} = 0 \wedge \frac{\partial f_j(x)}{\partial x_i} > 0 \right) \\ 1 & \text{if } \frac{\partial f_i(x)}{\partial x_j} < 0 \vee \left(\frac{\partial f_i(x)}{\partial x_j} = 0 \wedge \frac{\partial f_j(x)}{\partial x_i} < 0 \right) \\ \text{arbitrary in } \{0, 1\} & \text{if } \frac{\partial f_i(x)}{\partial x_j} = \frac{\partial f_j(x)}{\partial x_i} = 0. \end{cases} \quad (6)$$

Often, one wants to check whether a given vector field is of type K_δ for some (yet unknown) sign structure δ . Step 1 obviously remains the same, but in Step 2 we need to decide whether (5) is solvable for the unknown $\delta = \{\delta_1, \dots, \delta_n\}$. The following proposition presents an easy way to do this. Moreover, it shows that if the answer is positive, the orthant signature can be easily extracted from the sign structure of the Jacobian matrix.

Proposition 3. The system of Boolean equations (5) is solvable w.r.t. δ_i if and only if the following condition is satisfied:

$$s_{ij} \oplus s_{ik} = s_{jk}, \quad i < j, j < k, i, j, k \leq n. \quad (7)$$

Proof (necessity) . Let us rewrite expression $s_{ij} \oplus s_{ik}$ using (5):

$$s_{ij} \oplus s_{ik} = \delta_i \oplus \delta_j \oplus \delta_i \oplus \delta_k.$$

By definition, $a \oplus b \equiv b \oplus a$, $a \oplus a \equiv 0$ and $a \oplus 0 \equiv a$. Thus, $s_{ij} \oplus s_{ik} = \delta_j \oplus 0 \oplus \delta_k = \delta_j \oplus \delta_k = s_{jk}$, and we have shown that (7) follows from (5). (**sufficiency**). We now show that (7) implies that

$$\delta = \{0, s_{12}, \dots, s_{1n}\} \quad (8)$$

is a solution of (5). $\delta_1 \oplus \delta_j = 0 \oplus s_{1j} = s_{1j}$ holds trivially for $j = \{2, \dots, n\}$, and $\delta_i \oplus \delta_j = s_{1i} \oplus s_{1j} = s_{ij}$, $i, j \in \{2, n\}, j > i$, where the last equality follows from (7). \square

Similarly, it can be shown that

$$\tilde{\delta} = \{1, s_{12} \oplus 1, \dots, s_{1n} \oplus 1\} \quad (9)$$

is also a solution of (5) if (7) holds. Furthermore, (8) and (9) represent the only solutions. This can be shown by considering a vector δ' with $\delta'_i \neq \delta_i$ (i.e. $\delta'_i = \delta_i \oplus 1$) for some $i \in \{1, \dots, n\}$ and $\delta'_j = \delta_j$ for some $j \neq i$. Hence,

$$\delta'_i \oplus \delta'_j = \delta_i \oplus 1 \oplus \delta_j = s_{ij} \oplus 1 \neq s_{ij},$$

which shows that δ' is not a solution of (5).

Remark 4. (8) and (9) signify orthants that are symmetric w.r.t. the origin.

For linear systems

$$\dot{x}(t) = Ax(t),$$

the Jacobian matrix is the A matrix, i.e. $J(x, u) = A$. Thus, the sign structure of the Jacobian is completely determined by the signs of the elements a_{ij} . Obviously, they are sign-stable, so we need to check only conditions (4) and (5). Condition (4) (*sign-symmetry*) holds if $a_{ij}a_{ji} \geq 0$, $i \neq j$. The second step is to check the corresponding Boolean equation (5) using the method described in Prop. 3.

Example 5. Let us consider the Jacobian matrix with the following sign structure

$$J = \begin{bmatrix} * & + & 0 & - \\ + & * & + & 0 \\ 0 & + & * & 0 \\ - & 0 & 0 & * \end{bmatrix}.$$

Here we use asterisks to stress the fact that diagonal elements do not affect the monotonicity property. According to (6), we have $\{s_{12}, s_{14}, s_{23}\} = \{0, 1, 0\}$ while s_{13}, s_{24} , and s_{34} are arbitrary. From Proposition 3 we can deduce that (5) is solvable iff $\{s_{13}, s_{24}, s_{34}\} = \{0, 1, 1\}$. The corresponding signature is $\delta = \{0, 0, 0, 1\}$.

3.2 Controlled systems

Some of the previous results can be extended to dynamical systems driven by an exogenous input signal. A system

$$\dot{x}(t) = f(x(t), u(t)), \quad (10)$$

where $x(t) \in X \subset \mathbb{R}^n$, $u(t) \in U \subset \mathbb{R}^m$, $f : X \times U \rightarrow \mathbb{R}^n$, generates a flow $\phi_t(x_0, u_\tau)$, $u_\tau = u(\tau)$, $0 \leq \tau \leq t$, which represents a solution of (10) with initial condition $x(0) = x_0$ and external input signal u .

Definition 6. A controlled dynamical system (10) is monotone w.r.t. the orthants \mathbb{R}_δ^n and \mathbb{R}_γ^m if the following implication holds for all $t \geq 0$:

$$x_1 \preceq_\delta x_2, u_1(\tau) \preceq_\gamma u_2(\tau), 0 \leq \tau \leq t \Rightarrow \phi_t(x_1, u_{1\tau}) \preceq_\delta \phi_t(x_2, u_{2\tau}).$$

In [Angeli and Sontag, 2003], a condition for the controlled system (10) to be monotone w.r.t. the orthants \mathbb{R}_δ^n and \mathbb{R}_γ^m has been proposed.

Proposition 7. ([Angeli and Sontag, 2003]). The system (10) is monotone w.r.t. the orthants \mathbb{R}_δ^n and \mathbb{R}_γ^m if and only if the following properties hold for all $x \in D$ and all $u \in U$:

$$(-1)^{\delta_i + \delta_j} \frac{\partial f_i}{\partial x_j}(x, u) \geq 0, \quad i \neq j, i, j \leq n$$

$$(-1)^{\delta_i + \gamma_j} \frac{\partial f_i}{\partial u_j}(x, u) \geq 0, \quad i \leq n, j \leq m.$$

The above conditions are, in fact, the extended variant of condition (2) from the previous section. Hence, in addition to conditions (3), (4) and (5), which are used to check (2), the following tests need to be performed:

First, the partial derivatives w.r.t. the control variables need to be *sign stable*, i.e.

$$\frac{\partial f_i(x, u)}{\partial u_j} \geq 0 \text{ or } \frac{\partial f_i(x, u)}{\partial u_j} \leq 0, \quad \forall x \in D, \forall u \in U \quad (11)$$

for all $i \leq n$, $j \leq m$. Moreover, the set of Boolean equations

$$\delta_i \oplus \gamma_j = q_{ij}, \quad i \leq n, j \leq m, \quad (12)$$

where

$$q_{ij} = \begin{cases} 0 & \text{if } \frac{\partial f_i(x)}{\partial u_j} > 0, \\ 1 & \text{if } \frac{\partial f_i(x)}{\partial u_j} < 0, \\ \text{arbitrary in } \{0, 1\} & \text{if } \frac{\partial f_i(x)}{\partial u_j} = 0, \end{cases} \quad (13)$$

needs to be solvable with respect to the vector $\gamma = \{\gamma_1, \dots, \gamma_m\}$.

The following proposition gives a necessary and sufficient condition for equations (5) and (12) to be solvable.

Proposition 8. The systems of Boolean equations (5) and (12) are solvable if and only if the following conditions are satisfied:

$$s_{ij} \oplus s_{ik} = s_{jk}, \quad i < j, j < k, i, j, k \leq n, \quad (14)$$

$$q_{ij} \oplus q_{kj} = s_{ik}, \quad i \neq k, i, k \leq n, j \leq m. \quad (15)$$

Moreover,

$$\begin{aligned} \delta &= \{0, s_{12}, \dots, s_{1n}\}, \\ \gamma &= \{q_{11}, \dots, q_{1m}\} \end{aligned}$$

is a solution.

Proof. The proof can be carried out according to the same scheme as in Prop. 3. \square

Remark 9. It can be shown that the solution is also defined up to inversion, i.e.

$$\begin{aligned} \tilde{\delta} &= \{1, s_{12} \oplus 1, \dots, s_{1n} \oplus 1\}, \\ \tilde{\gamma} &= \{q_{11} \oplus 1, \dots, q_{1m} \oplus 1\} \end{aligned}$$

is the only other solution of (5), (12).

Remark 10. Condition (15) can be represented as

$$\begin{aligned} q_{k1} \oplus q_{l1} &= s_{kl}, \quad k < l, k, l \leq n, \\ \text{col}_1(Q) &\stackrel{\oplus}{=} \text{col}_j(Q) \quad \forall i, j \leq m, \end{aligned}$$

where $Q = q_{ij}$, $i \in \{1, \dots, n\}$, $j \in \{1, \dots, m\}$, and $\stackrel{\oplus}{=}$ denotes an equality up to the inversion w.r.t. \oplus .

3.3 Special cases

In the following we point out two special cases of a controlled system (10) where the simpler Proposition 3 suffices to check monotonicity.

a) If $u(t)$ is entirely defined by the present state $x(t)$, i.e. $u(t) = u(x(t))$, the Jacobian of the closed loop system is

$$J_{ij}(x) = \frac{\partial f_i(x, u(x))}{\partial x_j} + \sum_{l=1}^m \frac{\partial f_i(x, u(x))}{\partial u_l} \frac{\partial u_l(x)}{\partial x_j}. \quad (16)$$

and the procedure described in Proposition 3 can be applied to (16).

b) In a hybrid control context, the control vector often consists of two components, i.e. $u'(t) = [u'_1(t), u'_2(t)]$, where $u_1(t) \in \mathbb{R}^k$, $k < m$ is determined by continuous state feedback, i.e. $u_1(t) = u_1(x(t))$, and u_2 is a piecewise constant signal with finite range $\mathcal{U} \subset \mathbb{R}^{m-k}$, $|\mathcal{U}| = N \in \mathbb{N}$. In this case, the system can be treated separately on intervals, where u_2 is constant, i.e. $u_2(t) = u_\kappa \in \mathcal{U}$ $t \in [t_\kappa, t_{\kappa+1})$, and Proposition 3 can be applied again. The value u_κ is interpreted as a parameter, and the Jacobian is given by

$$J_{ij}^\kappa(x, u_\kappa) = \frac{\partial f_i(x, u_1(x), u_\kappa)}{\partial x_j} + \sum_{l=1}^k \frac{\partial f_i(x, u_1(x), u_\kappa)}{\partial u_{1l}} \frac{\partial u_{1l}(x)}{\partial x_j}$$

$\forall \kappa \in \mathbb{N}$.

Note that in our hybrid systems context, monotonicity is only needed to compute safe abstraction. Hence, having monotonicity w.r.t. different orthants for different values of κ will not pose any problems.

4. THE ROLE OF MONOTONICITY IN ABSTRACTION BASED HYBRID CONTROL SYNTHESIS

To put the previous discussion into context, we will now briefly describe the specific hybrid systems scenario we envisage as an application. Consider a continuous system

$$\dot{x}(t) = g_{u_2(t)}(x(t)) \quad (17)$$

where, as indicated before, u_2 is a piecewise constant signal with finite range \mathcal{U} , $|\mathcal{U}| = N$.

$$z(t) = h(x(t)) \quad (18)$$

is a discrete-valued output signal with finite range, i.e. $h : \mathbb{R}^n \rightarrow \mathcal{Z}$, $|\mathcal{Z}| = M < \infty$. Let us further assume that the system (17), (18) is sampled, either on a regular sampling grid (“time-driven sampling”) or on the sampling grid defined by the output signal z switching values (“event-driven

sampling”). In the latter case, the input may only be switched at the time instances where the output changes. In both cases, eqns. (17), (18) and the considered sampling device form a continuous system (with state set $X \subset \mathbb{R}^n$) evolving in discrete time \mathbb{N} on a discrete external signal space $\mathcal{U} \times \mathcal{Z}$. Let $\mathcal{B} \subset (\mathcal{U} \times \mathcal{Z})^{\mathbb{N}_0}$ denote its behaviour, i.e. the set of all pairs of discrete input and output signals compatible with the model assumptions. For abstraction based control synthesis, we need a discrete approximation, evolving on the same external signal space and exhibiting behaviour $\mathcal{B}_{\text{ab}} \supseteq \mathcal{B}$. In [Moor and Raisch, 1999], strongest ℓ -complete approximation was advocated as a particularly suitable abstraction. It is characterised by the behaviour

$$\mathcal{B}_\ell := \left\{ (u_2, z) : \mathbb{N}_0 \rightarrow \mathcal{U} \times \mathcal{Z} \mid (u_2, z)|_{[k, k+\ell]} \in \mathcal{B}|_{[0, \ell]} \forall k \in \mathbb{N}_0 \right\},$$

where the restriction operator $(\cdot)|_{[k, k+\ell]} : (\mathcal{U} \times \mathcal{Z})^{\mathbb{N}_0} \rightarrow (\mathcal{U} \times \mathcal{Z})^{(\ell+1)}$ picks the finite string ranging from the k -th to the $(k + \ell)$ -th pair of external events and disregards its absolute location on the time axis; it can naturally be extended to sets of signals.

From a computational point of view, determining the strongest ℓ -complete approximation boils down to deciding whether a given string of input and output symbols $(u_{20}, \dots, u_{2\ell}, z_0, \dots, z_\ell)$ is an element in $\mathcal{B}|_{[0, \ell]}$. To obtain a precise answer, we would need to compute the evolution of the quantisation cell $h^{-1}(z_0)$ under the flow $\phi_{u_{20}}$ associated with $g_{u_{20}}$, intersect the result with $h^{-1}(z_1)$, track the evolution of the result under the flow $\phi_{u_{21}}$ associated with $g_{u_{21}}$ etc. To obtain safe approximation, or abstraction, it is sufficient to compute outer approximations of these solutions. Clearly, if g is monotone w.r.t. the partial order \preceq , and quantisation cells are “boxes” w.r.t. \preceq , then $\phi_{u_{2i}}(h^{-1}(z_i))$ is “trapped” within the evolution of “external points”, i.e. $a \preceq h^{-1}(z_i) \preceq b$ implies

$$\phi_{u_{2i}}(a) \preceq \phi_{u_{2i}}(h^{-1}(z_i)) \preceq \phi_{u_{2i}}(b).$$

It is then a straightforward exercise to compute the required outer approximations and hence the desired safe abstraction [Moor and Raisch, 2002]. On the basis of such an abstraction, one can compute a discrete non-blocking supervisor enforcing a language-type specification. In [Moor and Raisch, 1999] it has been shown that the resulting supervisor will also be non-blocking and enforce the specification when connected to the underlying continuous model (17), (18).

5. MONOTONISATION THROUGH FEEDBACK

We consider linear control systems

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t), \\ y(t) = Cx(t) \end{cases} \quad (19)$$

where $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$, $y(t) \in \mathbb{R}^l$, B has full column rank and C has full row rank. If the monotonicity test fails, we may still be able to enforce monotonicity by appropriate feedback. For this purpose, we divide the vector of control inputs, $u' = [u'_1, u'_2]$, where $'$ means “transpose” and $u_1(t) \in \mathbb{R}^k$, $k < m$, is the part of the control input devoted to enforce monotonicity.

The system (19) then takes the form

$$\dot{x}(t) = Ax(x) + B^1 u_1(t) + B^2 u_2(t). \quad (20)$$

Defining the control input $u_1(t)$ as a linear function of the current output, $u_1(t) = Ky(t) = KCx(t)$, we change the Jacobian to $J = A + B^1 KC$ and, therefore, alter its sign structure accordingly. But we still do not have a clear algorithm to solve this problem in general because of the large number of degrees of freedom (recall that the number of orthants for an n -dimensional system is equal to 2^{n-1}).

The proposed semiformal algorithm uses an approach based on the successive reduction of the number of available degrees of freedom.

- (1) If either i -th row of B^1 or the j -th column of C is identical to zero, the elements a_{ij} of the Jacobian remain unchanged. We can now check these elements for consistency by investigating whether Conditions (4) and (5) are satisfied. Clearly, if this is not the case, the monotonicity condition cannot be enforced by feedback from $y(t)$ to $u_1(t)$.
- (2) If the result in Step 1 is positive, we can deduce the signs of some other elements of the Jacobian from (7). Note the following “extreme” case: suppose, as above, that the i -th row of B^1 (resp., the j -th column of C) are zero and that all the elements in the corresponding row (resp., column) of A are nonzero (apart possibly from the entry on the diagonal). Then, the corresponding s_{ik} , $k \neq i$ (resp., s_{kj} , $k \neq j$) completely determine the required sign structure of J , as (see (7))

$$s_{k_1 k_2} = s_{i k_1} \oplus s_{i k_2}, \quad (21)$$

resp.,

$$s_{k_1 k_2} = s_{k_1 j} \oplus s_{k_2 j}. \quad (22)$$

If, on the other hand, elements in the corresponding row (resp., column) of A are zero, there may be several admissible orthants.

- (3) In the next step, we isolate the entries of the Jacobian exhibiting inappropriate signs. We

now need to determine a feedback matrix K to adjust these elements without changing the signs of the other entries. For this, the elements of the real $[k \times l]$ -matrix K have to satisfy

$$(-1)^{s_{qp}}(a_{qp} + \sum_{i=1}^k \sum_{j=1}^l b_{qi}^1 k_{ij} c_{jp}) \geq 0, q \neq p.$$

The extension of the proposed algorithm to the class of nonlinear control systems is not straightforward. Usually, an arbitrary nonlinear control system admits monotonicity only in some subset of the state space, if it does at all. Let's denote by $X_\delta \subset X$ a subset of the state space X where the system can be rendered monotone w.r.t. the orthant with signature δ . It is quite common that some subspaces have nonempty intersection, i.e. $X_{\delta_1} \cap X_{\delta_2} \neq \emptyset$. Then, one must choose between several orthants. In this case, a decision can be made on the basis of heuristic considerations and can hardly be formalised. However, in some special cases (e.g. positive systems) the procedure can be successfully applied as is illustrated in the following example.

5.1 Example

To illustrate the applicability of the developed approach we consider a model of the biological processes in an activated sludge process, the so-called IAWQ's³ Activated Sludge Model No.1 (see [Henze et al., 1987, Lindberg, 1997]). This model describes the three following biological processes: removal of organic matter, nitrification, and denitrification. The considered process is an ideally mixed bioreactor with three components, which can be described by the following differential equations:

$$\begin{aligned} \frac{dX_b}{dt} &= \frac{Q_{in}}{V} X_{b,in} - \frac{Q_{out}}{V} X_b + \mu(S_s)X_b - bX_b \\ \frac{dS_s}{dt} &= \frac{Q_{in}}{V} S_{s,in} - \frac{Q_{out}}{V} S_s - \frac{1}{Y} \mu(S_s)X_b \\ \frac{dS_o}{dt} &= \frac{Q_{in}}{V} S_{o,in} - \frac{Q_{out}}{V} S_o - \frac{1-Y}{Y} \mu(S_s)X_b - bX_b \end{aligned} \quad (23)$$

where X_b , S_s and S_o represent the concentrations of biomass, soluble substrate and dissolved oxygen in the reactor. $X_{b,in}$, $S_{s,in}$ and $S_{o,in}$ are the influent concentrations of biomass, soluble substrate and dissolved oxygen. $\mu(S_s)$ is the specific growth rate of the biomass. It is described by Monod's equation,

$$\mu(S_s) = \frac{\bar{\mu} S_s}{K_s + S_s},$$

where $\bar{\mu}$ is the maximum specific growth rate and K_s is the half-velocity constant. The tank volume is denoted V , and the incoming and outgoing flows are Q_{in} and Q_{out} , respectively. The growth yield is Y and b is the decay rate. It is worth noting that all concentrations, input and output variables as well as parameters, are positive. Moreover, the growth yield Y is always less than one.

Using the conventional notation $x := [X_b, S_s, S_o]'$ and $u := [X_{b,in}, S_{s,in}, S_{o,in}, Q_{in}]'$ one can rewrite (23) as

$$\begin{aligned} \dot{x}_1 &= \frac{u_1 u_4}{V} - \frac{Q_{out}}{V} x_1 + \mu(x_2) x_1 - b x_1 \\ \dot{x}_2 &= \frac{u_2 u_4}{V} - \frac{Q_{out}}{V} x_2 - \frac{1}{Y} \mu(x_2) x_1 \\ \dot{x}_3 &= \frac{u_3 u_4}{V} - \frac{Q_{out}}{V} x_3 - \frac{1-Y}{Y} \mu(x_2) x_1 - b x_1. \end{aligned} \quad (24)$$

The Jacobian matrix has the following form

$$\frac{Df}{Dx} = \begin{bmatrix} * & \frac{\bar{\mu} K_s x_1}{(K_s + x_2)^2} & 0 \\ -\frac{1}{Y} \frac{\bar{\mu} x_2}{K_s + x_2} & * & 0 \\ -\frac{1-Y}{Y} \frac{\bar{\mu} x_2}{K_s + x_2} - b & -\frac{1-Y}{Y} \frac{\bar{\mu} K_s x_1}{(K_s + x_2)^2} & * \end{bmatrix}. \quad (25)$$

We see that the partial derivatives $\frac{\partial f_1}{\partial x_2}$ and $\frac{\partial f_2}{\partial x_1}$ do not satisfy the *sign-symmetry* condition. Now one has to determine, which one has the "right" sign. The remaining elements of the Jacobian matrix satisfy conditions (3) and (4). The corresponding variables are $s_{13} = 1$, $s_{23} = 1$. Then, from (22) $s_{12} = 0$. This means that both $\frac{\partial f_1}{\partial x_2}$ and $\frac{\partial f_2}{\partial x_1}$ must be nonnegative. The easiest way to change the sign of $\frac{\partial f_2}{\partial x_1}$ is to use the control u_2 , because it does not enter the remaining equations. Considering the control u_2 as a function of the state variables, $u_2 = u_2(x)$, we can rewrite the Jacobian (25) as:

$$\frac{Df}{Dx} = \begin{bmatrix} * & \frac{\bar{\mu} K_s x_1}{(K_s + x_2)^2} & 0 \\ \frac{u_4}{V} \frac{\partial u_2(x)}{\partial x_1} - \frac{1}{Y} \frac{\bar{\mu} x_2}{K_s + x_2} & * & \frac{u_4}{V} \frac{\partial u_2(x)}{\partial x_3} \\ -\frac{1-Y}{Y} \frac{\bar{\mu} x_2}{K_s + x_2} - b & -\frac{1-Y}{Y} \frac{\bar{\mu} K_s x_1}{(K_s + x_2)^2} & * \end{bmatrix}$$

Hence, the control $u_2(x)$ has to be chosen to satisfy the following conditions:

³ International Association for Water Quality.

$$\frac{\partial u_2}{\partial x_1}(x) \geq \frac{V}{u_4 Y} \frac{\mu x_2}{K_s + x_2}, \quad (26)$$

$$\frac{\partial u_2}{\partial x_3}(x) \leq 0, \quad (27)$$

$$\forall x \in \mathbb{R}_{\geq 0}^3, u_4 \neq 0.$$

Conditions (26), (27) define a family of control laws. In particular, a control law can be chosen as

$$u_2(x) = c_1 x_1,$$

where $c_1 = \frac{V\bar{\mu}}{u_4^* Y}$, $u_4^* = \min u_4$. Thus, the system can be rendered monotone by a simple linear feedback. Moreover, it turns out, that the monotonicisation procedure does not require the measurement of all state variables.

6. CONCLUSION

We discussed the question how the concept of monotonicity can be used in the context of hybrid systems. We provided a simple and efficient algorithm to check whether an arbitrary continuous system is monotone with respect to some (a priori unknown) partial order relation. This algorithm was extended to the case of control systems. It was also shown how to enforce monotonicity with the help of feedback. The developed semiformal approach was illustrated by an example. We considered a nonlinear model of an ideally mixed bioreactor and showed that this system can be rendered monotone by a very simple linear feedback.

7. ACKNOWLEDGEMENTS

The authors gratefully acknowledge helpful discussions with D. Flockerzi, MPI Magdeburg.

REFERENCES

- David Angeli and Eduardo D. Sontag. Monotone control systems. *IEEE Transactions on Automatic Control*, 48(10):1684 – 1698, October 2003.
- A. Chutinan and B. H. Krogh. Computing approximating automata for a class of hybrid systems. *Mathematical and Computer Modeling of Dynamical Systems: Special Issue on Discrete Event Models of Continuous Systems*, 6:30 – 50, March 2000.
- J.E.R. Cury, B.H. Krogh, and T. Niinomi. Synthesis of supervisory controllers for hybrid systems based on approximating automata. *IEEE Transactions on Automatic Control*, 43(4):564 – 568, April 1998.

- M. Henze, C.P.L. Grady, W. Gujer, G.v.R Marais, and T. Matsuo. *Activated sludge model No.1*. IAWPRC Scientific and technical reports, No.1. International Association on Water Pollution Research and Control, IAWPRC, London, 1987.
- X.D. Koutsoukos and P.J. Antsaklis. Safety and reachability of piecewise linear hybrid dynamical systems based on discrete abstractions. *Discrete Event Dynamic Systems*, 13(3):203 – 243, July 2003.
- E.S. Lemch and P.E. Caines. Hybrid partition machines with disturbances: hierarchical control via partition machines. In *Proceedings of the 38th IEEE Conference on Decision and Control*, volume 5, pages 4909 – 4914, 1999.
- C.-F. Lindberg. *Control and estimation strategies applied to the activated sludge process*. PhD thesis, Uppsala University, 1997.
- T. Moor and J. Raisch. Abstraction based supervisory controller synthesis for high order monotone continuous systems. In S. Engell, G. Frehse, and E. Schnieder, editors, *Modelling, Analysis, and Design of Hybrid Systems*, LNCIS 279, pages 247 – 265. Springer-Verlag, 2002.
- T. Moor and J. Raisch. Supervisory control of hybrid systems within a behavioural framework. *Systems & Control Letters*, 38(3):157 – 166, October 1999.
- Hal L. Smith. *Monotone dynamical systems: an introduction to the theory of competitive and cooperative systems*, volume 41 of *Mathematical surveys and monographs*. American Mathematical Society, Providence, RI, 1995.