Analysis and Synthesis of Systems with Discrete and Continuous Control

Pieter Cuijpers[1], Aleksandar Juloski[2],
Michel Reniers[1], Maurice Heemels[2],
Jan Friso Groote[1], Paul van den Bosch[2]
[1] Department of Computer science
[2] Department of Electrical Engineering
Eindhoven University of Technology
PO Box 513, 5600 MB Eindhoven, The Netherlands
Phone: +31 40 247 8216
E-mail: P.J.L.Cuijpers@tue.nl

Abstract—**Two different lines of research are presented in the study of an industrially relevant problem of impact control. One is focused on actual control of the process, while the other is focused on the development of a general language for modeling hybrid systems, of which the case study is an appropriate example.**

Keywords— **hybrid systems, process algebra, syntax, semantics,piecewise affine systems, observer design**

I. INTRODUCTION

The theory of hybrid systems studies the interaction between continuous and discrete behaviour. When discrete software is combined with mechanical and electrical components, or is interacting with, for example, chemical processes, an embedded system arises in which the interaction between the continuous behaviour of the components and processes and the discrete behaviour of the software is important. Although there are good methods for analyzing continuous behaviour (control science / system theory) as well as for analyzing discrete behaviour (computer science / automata and process theory), the interaction between those two fields is largely unexplored. There are only a few models that can handle (some) interaction, and often these models are still dominated by one of the two original fields.

In practice, often the discrete part of a system is described and analyzed using methods from computer science, while the continuous part is handled by control science. The design of the complete system is usually such that interaction is suppressed to a minimum. Because of this suppressed interaction, analysis is possible to some extend, but it limits the design options. This is the main reason for the development of a theory on hybrid systems that allows for a less restricted analysis of interaction.

Two relatively simple case studies, a thermometer and an impact control process have been selected as a source of inspiration, that will help to bridge the gap between the two disciplines as a combined design of a control system has to be realized. Two different lines of research are currently followed: a top-down approach in which the mathematical modeling of hybrid systems is considered for which the impact problem forms a clear test case for the efficiency and elegance of the modeling language. Concepts like safety, controllability, observability, bisimulation equivalence, etc., which will be important for the extension of the case study, can easily be defined in this new language. The other approach follows a bottom-up way of thinking: specific problems (e.g. observer design and stabilization) have been considered for the case study at hand and extended to a broader scope.

These two lines of research are intended to converge to a point where the results of the one approach can be applied to the language developed in the other. The impact problem will form a nice carrier problem that requires a control system design in which safety checking, verification and scheduling (typical for the computer science domain) and continuous controller design (requiring computer engineering tools related to stabilization, tracking, etc.) have to be combined.

In the following sections, we present our view on mathematical modeling, the industrial example of the impact process and the research work motivated by it.

II. MATHEMATICAL MODELING

In this section we explain our view on the concept of mathematical modeling in general and in particular our view on how to combine mathematical models. An informal explanation is given of what a mathematical model consists of, and why.

A mathematical *formalism* provides us with a structure

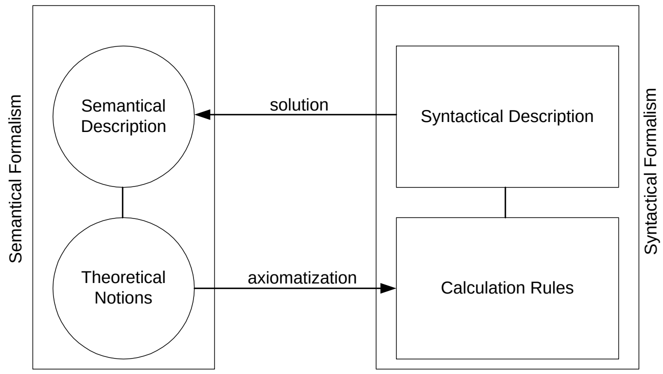

Fig. 1. Mathematical Modeling

in which we can describe systems, and in which we can analyse them. Mathematical modeling often makes use of two of such formalisms called syntax and semantics (see figure 1).

The *semantical formalism* (in short: semantics) is intended to support the modeling of a system on a low level of abstraction. The *semantical description* of a system makes use of a relatively simple mathematical structure. Motion, on a semantical level, would for example be modeled using functions of time to space. A computer program would be modeled using a transition system. The semantical formalism contributes to the analysis of systems by the intuitive definitions it provides of the *theoretical notions* we want to analyse. Because of its simplicity in mathematical structure, the semantical formalism allows us to give an intuitive and formally precise definition of properties like equivalence, stability, absence of deadlock, controllability, and observability.

The *syntactical formalism* (in short: syntax) is intended to facilitate a less cumbersome description of a system. In contrast to the mathematically simple description method that the semantical formalism provides, *syntactical description* is focused on the ease of notation. Writing down the complex ways in which planets move using functions of time is much harder than describing them using, for example, differential equations. The high-level Pascal or C++ code of a computer program, is far more easy to write down than a transition system with the same functionality. Actually, semantical descriptions often are infinite and, therefore, impossible to write down. Syntax provides a finite way of handling these semantically infinite objects. This suggests that the syntactical and semantical formalism are coupled, which indeed they are. A differential equation has solutions in terms of functions of time. A piece of C++ code, although not formally, represents a transition system.

The contribution of syntax to the analysis of systems is through axioms and theorems that we refer to in the figure as *calculation rules*. Because syntax and semantics are coupled, the notions that are defined in the semantics, have a meaning in the syntax. The calculation rules on the syntax, should reflect these notions. Axioms (for example) usually represent notions of equivalence on the semantics, while theorems about the stability of systems should correspond to the definition of stability in semantical terms. For the analysis of systems it is important that the coupling between syntax and semantics is formal, this is one of the reasons why C++ programs are almost impossible to analyse completely. This is also the reason why a lot of investigation is put in the development of a formal semantics for the χ language [16], [2], [15] (a programming language developed at TU/e that can handle timing constraints, and is being extended towards the use of differential equations). Typical syntactical languages that were developed with the intention of analysis from the beginning, are process algebras like ACP (Algebra of Communicating Processes) [3], [6], μCRL (micro Common Representation Language) [7], [8] and CCS (Calculus of Communicating Systems) [12].

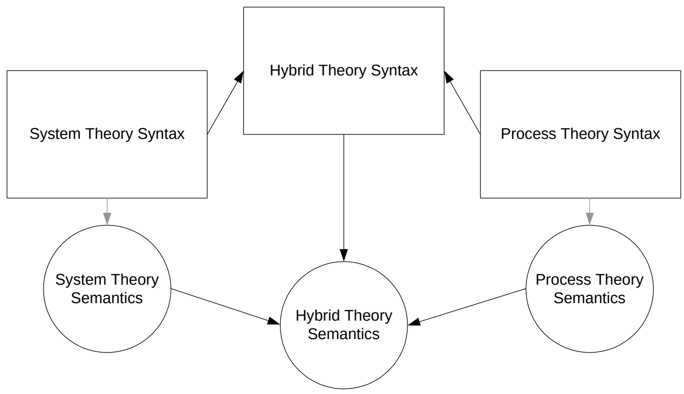

Fig. 2. Developing Hybrid Theory

In figure 2, a graphical representation is given of the general aim of our efforts. The figure shows that we want to combine the syntax used by system theorist and the syntax used by process theorist into a new hybrid syntax. A similar thing is done with the semantics of both fields. In order to be intuitive, the hybrid interpretation of a classical semantical solution of a classical syntactical statement should be the same as the hybrid semantical solution of the hybrid interpretation of that same classical syntactical statement. Hence, the figure should be commuting. A similar figure can be made with respect to for example the semantical description of hybrid systems and hybrid theoretical notions. We need to define new notions, in terms of hybrid semantics, of for example stability, that have the same meaning as their original notions when they are applied to hybrid interpretations of old system theoretic semantical models.

III. Hybrid Syntax and Semantics

A key starting point in the research that has been going on in this project with respect to syntax and semantics, is the need for a syntactical description of hybrid systems that allows analysis through symbolic reasoning. However, when we started to develop a process algebra (or more precisely a process theory) to describe hybrid systems, the issue of a suitable semantical model arose. Literature research pointed out that many of the existing semantical models, like hybrid automata [11] and rich-time behavioural systems [13], are surely expressive enough for our needs. Nevertheless, it was felt that the structure of those models was such that for the definition of certain crucial theoretical notions (like bisimulation equivalence) would become unnecessarily cumbersome. A new semantical model is proposed in [5] that, in a sence, is a combination of Sontag machines [14] and (timed) transition systems [4]. To our taste, this structure is more suitable as a semantics for hybrid process theory. In [5] the necessary proofs are given that this semantical formalism can indeed serve to make figure 2 commuting.

Definition III.1 (Hybrid Transition System) A *hybrid transition system* is a tuple $\langle X, \Sigma, T, \phi \rangle$ consisting of a state space X, a signal space Σ, a totally ordered time axis T and a hybrid transition relation

$$\phi \subseteq (X \times T) \times ((T \mapsto \Sigma) \cup \Sigma) \times (X \times T) .$$

We restrict the model in such a way that for all transitions (x, t, σ, x', t') we assume if $\sigma \in T \mapsto \Sigma$ then the domain of σ has the form $Dom(\sigma) = [0 ... \tau)$ where $\tau = t' - t$. Furthermore we assume instantaneous actions such that if $\sigma \in \Sigma$ we have $t = t'$.

The development of a hybrid process algebraic syntax to describe systems of this kind is still in progress. A simple example of a thermostat in the preliminary syntax is given below. Note the use of differential equations ($\dot{T}_R = C(T_O - T_R)$, etc.) on the temperatures that play a role in the system in combination with discrete actions ($H - on$, $H - off$ etc.) used to steer the heater. Furthermore, we use the process algebraic operators for *sequential composition* ($p \cdot q$), *alternative composition* ($p + q$), *parallel composition* ($p \,\|\, q$) and *disruption* ($p \blacktriangleright q$). Finally, *invariants* ($\text{Bool} \lhd p$) limit the evolution of a system according to some boolean.

Example III.2 The temperature of the room is modeled by T_R while the temperature of the environment is modeled using T_O. The heat capacity of the room is given through the constant C.

$$\text{Room} \;:\; \dot{T}_R = C(T_O - T_R) .$$

The heater is modeled as a flame that heats up instantly from 0 to 5 degrees when switched on and cools down instantly when switched off.

$$\text{Heater} \;:\; (\text{H-On} \cdot T_H = 5 + \text{H-Off} \cdot T_H = 0) \blacktriangleright \text{Heater} .$$

The thermostat controller is a bang-bang controller. As long as the temperature does not drop below 1, the controller is in the C-Off mode. If the temperature drops below 1 the controller switches to C-On mode until the temperature rises above 3. Using the synchronisation function γ, the controller will synchronise its mode with the heater.

$$\begin{array}{l} \text{Controller} \;: \\ ((T_C > 1) \lhd (\text{C-Off} \cdot (\dot{t} = 1)) + \\ (T_C < 3) \lhd (\text{C-On} \cdot (\dot{t} = 1))) \cdot \text{Controller} . \end{array}$$

Now, the whole system is connected by coupling the right variables to eachother in one parallel composition, synchronizing the discrete actions through the communication function γ as is usual in languages like μCRL [8].

$$\begin{array}{l} \text{Off} = \text{H-Off} \,\gamma\, \text{C-Off} \\ \text{On} = \text{H-On} \,\gamma\, \text{C-On} \end{array}$$

$$\text{Thermostat} \;:\; \left\| \begin{array}{l} \text{Room} \\ \text{Heater} \\ \text{Controller} \\ T_O = T_H \\ T_C = T_R \end{array} \right.$$

The impact control problem, that is discussed in section IV, will serve as a future case for the hybrid syntax, and after the link between syntax and semantics has been established, we will attempt analysis of the impact control problem along system theoretic the lines that are developed furtheron in this paper, with the new hybrid theory.

IV. Impact detection and control

In this section we present a (simplified) model of a real industrial process, namely the placement head of a pick-and place machine.

The pick-and-place machine works as follows: the mounting head, carrying an electronic component of (un)known mass is navigated to the position where the component should be placed and glued on the printed circuit board (PCB). The component is placed, released, and

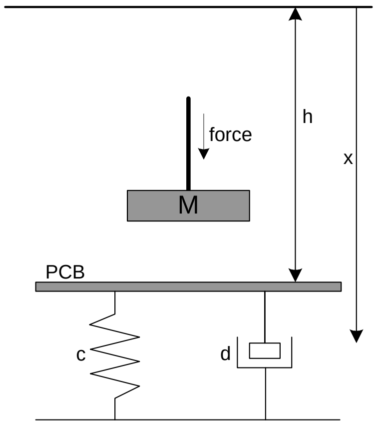

Fig. 3. Simplified model of the impact process

the whole process is repeated with the next component. The whole operation should be as fast as possible (to increase the throughput of the machine), while satisfying some procedural and safety constraints (e.g. the component must be pressed on the PCB with a sufficient, but not excessive force). During the placement an impact occurs between the component and the PCB. A contact/impact sensor is not present on the mounting head, but the occurrence of the impact must be detected as fast as possible.

The idea is to try to detect the impact on the basis of a model of the process. Typically, this model consists of two modes of operation (the mass in free motion and the mass in contact with the PCB). The mode switch is triggered by a state of the system (height of the component hitting the PCB). If, from the measured variables, all states can be estimated sufficiently fast and accurate in spite of the discontinuities in the system description, then these estimated states can be used for impact detection and control. This problem is known in system theory as an observer design problem. More specifically, an observer has to be designed for the piecewise linear (affine) model, when the currently active linear dynamics (mode) is *not* known. The solution for the general case of bi-modal PWA systems is presented in the following section. Detailed discussion can be found in [10].

V. Observers for bi-modal PWA systems

Consider the following system

$$\dot{x} = \begin{cases} A_1 x + Bu, & \text{if } H^\top x \leq 0 \\ A_2 x + Bu, & \text{if } H^\top x > 0 \end{cases} \tag{1a}$$

$$y = Cx \tag{1b}$$

where $x \in \mathbb{R}^{\mathrm{n}}$, $y \in \mathbb{R}^{\mathrm{p}}$, $u \in \mathbb{R}^{\mathrm{m}}$, $A_1, A_2 \in \mathbb{R}^{\mathrm{n}\times\mathrm{n}}$, $B \in \mathbb{R}^{\mathrm{n}\times\mathrm{m}}$, $C \in \mathbb{R}^{\mathrm{p}\times\mathrm{n}}$ and $H \in \mathbb{R}^{\mathrm{n}}$. The hyperplane defined by $\ker H^\top$ therefore separates the two half-spaces in which the state of the system resides. The considered class of bi-modal piece-wise affine systems has identical input distribution matrix B for both modes. The output distribution matrix C is taken to be the same for both modes as well, but this feature is not essential for the derivation of the results.

As an observer for the system (1), we propose a bi-modal system with the following structure:

$$\dot{\hat{x}} = \begin{cases} A_1 \hat{x} + Bu + L_1(y - \hat{y}), & \text{if } H^\top \hat{x} \leq 0 \\ A_2 \hat{x} + Bu + L_2(y - \hat{y}), & \text{if } H^\top \hat{x} > 0 \end{cases} \tag{2a}$$

$$\hat{y} = C\hat{x} \tag{2b}$$

where $\hat{x} \in \mathbb{R}^{\mathrm{n}}$ and L_1 and $L_2 \in \mathbb{R}^{\mathrm{n}\times\mathrm{p}}$ are matrices. For any pair of matrices L_1 and L_2 the *observed system* is defined by the joint equations (1) and (2). Specifically, let the *state estimation error* e be defined as

$$e = x - \hat{x}.$$

The dynamics of the state estimation error is then described by

$$\dot{e} = \begin{cases} (A_1 - L_1 C)e, & H^\top x \leq 0, \quad H^\top \hat{x} \leq 0 \\ (A_2 - L_2 C)e + \Delta A x, & H^\top x \leq 0, \quad H^\top \hat{x} > 0 \\ (A_1 - L_1 C)e - \Delta A x, & H^\top x > 0, \quad H^\top \hat{x} \leq 0 \\ (A_2 - L_2 C)e, & H^\top x > 0, \quad H^\top \hat{x} > 0 \end{cases} \tag{3}$$

where x satisfies (1a), $\hat{x}$ satisfies (2a), and $\Delta A = A_1 - A_2$. By substituting $\hat{x} = x - e$ in (3), we see that the right-hand side of the state estimation error dynamics is piece-wise linear in the variable $v := \mathrm{col}(e, x)$. (Here, col stacks subsequent entries of its argument in a column matrix).

Note that the error dynamics in two modes of (3) is described by an n dimensional autonomous state equation, while in the two other modes by a n dimensional state vector plus external signal $x(t)$ which, by (1a), depends on the input u. Assume that the time evolution of the input signal $u(t)$ is fixed, and equal to some signal $u_0(t)$. The evolution of the system state $x(t)$ is then completely determined by the initial condition at time $t = 0$ and the values of the input signal $u_0(t)$. Denote this evolution as $x_0(\cdot)$. The signal $x_0(\cdot)$ then appears in the evolution of the error dynamics (3) as a known signal, independent of the estimation error e. Hence, for fixed input signals it is possible to consider the evolution of the error e in (3) as a time dependent equation of the form

$$\frac{de}{dt}(t) = f_{x_0}(t, e(t)). \tag{4}$$

Standard concepts and results of Lyapunov stability theory (see for instance [17]) can now be applied to equation (4).

Consider system (1), observer (2), and the error dynamics (3). The way to choose the observer gains L_1 and L_2 in (2a) is given by the following theorem.

Theorem V.1 *The state estimation error dynamics* (3) *is globally asymptotically stable (in the sense of Lyapunov) if there exist matrices* $P = P^\top > 0$, L_1, L_2 *of appropriate dimensions and constants* $\lambda_1, \lambda_2 \geq 0$ *such that the following set of matrix inequalities is satisfied:*

$$(A_1 - L_1C)^\top P + P(A_1 - L_1C) < 0 \qquad (5a)$$

$$\begin{bmatrix} (A_2 - L_2C)^\top P \\ +P(A_2 - L_2C) & \begin{matrix} P\Delta A \\ +\lambda_1 \frac{1}{2} HH^\top \end{matrix} \\ \begin{matrix} \Delta A^\top P + \\ \lambda_1 \frac{1}{2} HH^\top \end{matrix} & -\lambda_1 HH^\top \end{bmatrix} \leq 0 \qquad (5b)$$

$$\begin{bmatrix} \begin{matrix} (A_1 - L_1C)^\top P \\ +P(A_1 - L_1C) \end{matrix} & \begin{matrix} -P\Delta A \\ +\lambda_2 \frac{1}{2} HH^\top \end{matrix} \\ \begin{matrix} -\Delta A^\top P \\ +\lambda_2 \frac{1}{2} HH^\top \end{matrix} & -\lambda_2 HH^\top \end{bmatrix} \leq 0 \qquad (5c)$$

$$(A_2 - L_2C)^\top P + P(A_2 - L_2C) < 0. \qquad (5d)$$

Remark V.2 The inequalities (5a)-(5d) are not linear in $\{P, L_1, L_2, \lambda_1, \lambda_2\}$, but are linear in $\{P, L_1^\top P, L_2^\top P, \lambda_1, \lambda_2\}$.

The presented theory will be illustrated by means of the following bimodal system:

$$A_1 = \begin{bmatrix} -5 & -4 \\ -1 & -2 \end{bmatrix}, A_2 = \begin{bmatrix} -2 & -4 \\ 20 & -2 \end{bmatrix}$$

$$B = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, H = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, C = \begin{bmatrix} 0 & 1 \end{bmatrix}.$$

This example originates from [9], where it was used to illustrate the need to use piece-wise quadratic Lyapunov functions for stability analysis of piece-wise affine systems. Characteristic feature of this example is that the autonomous dynamics is continuous over the switching plane, and in this special case, observer can also be designed using circle criterion approach [1]. Note that the continuity plays no role in our approach. We see that the switching is driven by the first state variable x_1, while x_2 is measured. Hence, the discrete mode can not be reconstructed directly from the measurements.

The linear matrix inequalities (5a)-(5d) were solved using the MATLAB LMI toolbox, where nonstrict inequalities (5b),(5c) were replaced by strict inequalities, of the form:

$$\text{lhs}(\cdot) < \begin{bmatrix} 0 & 0 \\ 0 & \varepsilon I_n \end{bmatrix}$$

where ε is a small positive constant. A strictly feasible solution to this set of LMI's was obtained, and checked to satisfy the original, non-strict inequalities. The following observer gains were computed in this way:

$$L_1 = \begin{bmatrix} 0.1525 \\ 24.5475 \end{bmatrix}, \quad L_2 = \begin{bmatrix} 1.1811 \\ 24.9615 \end{bmatrix}.$$

For the purpose of the simulation, the following input was applied to the system:

$$u(t) = \begin{cases} 1, & \lfloor t \rfloor \text{ even} \\ 0, & \lfloor t \rfloor \text{ odd} \end{cases}$$

The initial conditions for the system were chosen as $x(0) = \begin{bmatrix} -1 & -1 \end{bmatrix}^\top$, and for the observer $\hat{x}(0) = \begin{bmatrix} 1 & 1 \end{bmatrix}^\top$. The simulation results are shown in figures 4,5. We see that the estimated state converges asymptotically to the true state of the system.

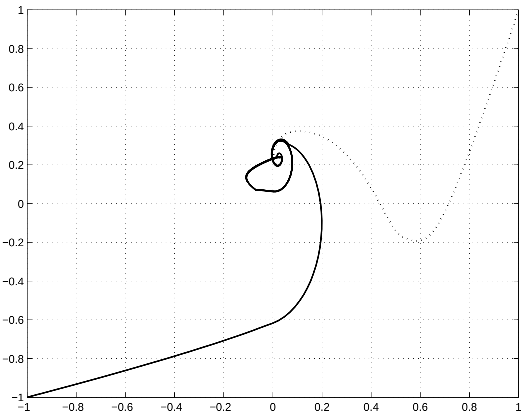

Fig. 4. System (solid) and observer (dotted) response in the phase plane

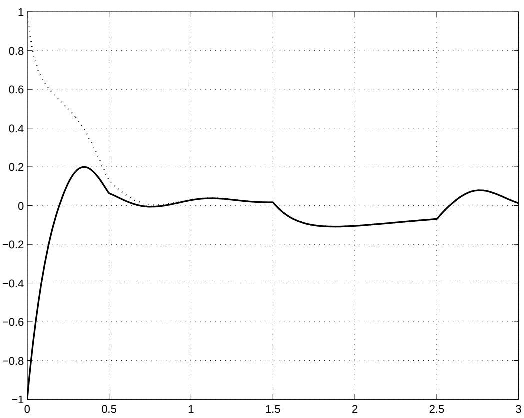

Fig. 5. System (solid) and observer (dotted) response in time domain

An important phenomenon which can occur in nonsmooth systems is so-called sliding motion. Namely, un-der certain conditions, the system remains on the switching surface $H^T x = 0$, while the state evolution is described by a convex combination of the two constituting linear dynamics. A detailed analysis shows that even in this case the proposed observer is a good one in the sense that the estimation error converges to 0 (see [13] and the references therein for a further discussion on sliding phenomena in non-smooth systems).

The next step is to apply the developed theory on a real example. The impact detection problem can be described within the framework of the bimodal piecewise affine systems. We will try to use the presented theory to design the observer which will, on the basis of process measurements, provide the mode estimate and hence, detect the impact. The corresponding practical set-up is currently being realized in our laboratory.

VI. Conclusions and Future Work

To summarize, two different lines of research are used to study the industrially relevant problem of impact control. One is focused on short time results with respect to control. Using a mainly system theoretic way of modeling the process, results have been booked regarding the observability of piece-wise affine systems, of which the impact control process is an example. The other line of research gave rise to a general language for the description of hybrid processes. A small case study of a thermometer was used to illustrate the type of language that is being developed, while in the near future also the impact process is going to be addressed this way. The ultimate goal is to make the two lines of research converge by integrating the results of the first into the language of the second.

Acknowledgements

This research is financially supported by STW/PROGRESS grant EES5173 and also received contributions from Philips CFT and Assembleon. We would also like to thank Erik Gaal, Arno de Vet (Philips CFT) and Ben Smeets (Assembleon) for their many useful suggestions.

References

[1] M. Arcak and P. Kokotović. Observer based control of systems with slope-restricted nonlinearities. *IEEE Transactions on Automatic Control*, 46(7):1146–1150.

[2] N.W.A. Arends. *A Systems Engineering Specification Formalism*. PhD thesis, TU/e, 1996.

[3] J.C.M. Baeten and W.P. Weijland. *Process Algebra*, volume 18 of *Cambridge Trancts in Theoretical Computer Science*. Cambridge University Press, Cambridge, 1990.

[4] J.A. Bergstra, A. Ponse, and S.A. Smolka, editors. *Handbook of Process Algebra*. Elsevier, Amsterdam, 2001.

[5] P.J.L. Cuijpers, M.A. Reniers, and W.P.M.H. Heemels. Hybrid transition systems (to appear).

[6] W. Fokkink. *Introduction to Process Algebra*. Springer-Verlag Berlin, 1998.

[7] J.F. Groote and B. Lisser. Computer assisted manipulation of algebraic process specifications. Technical report, CWI, Amsterdam.

[8] J.F. Groote and A. Ponse. The syntax and semantics of μCRL. In A. Ponse, C. Verhoef, and S.F.M. van Vlijmen, editors, *ACP: Algebra of Communicating Processes*, Workshops in Computing, pages 26–62, Utrecht, 1995. Springer-Verlag.

[9] M. Johansson and A. Rantzer. Computation of piecewise quadratic lyapunov functions for hybrid systems. *IEEE Transactions on Automatic Control*, 43(4):555–559.

[10] A. Juloski, M. Heemels, and S. Weiland. Observer design for a class of piecewise affine systems. In *accepted for presentation at CDC 2002, Las Vegas, USA*, 2002.

[11] N. Lynch, R. Segala, and F. Vaandrager. Hybrid i/o automata. Technical report, University of Nijmegen, 2001.

[12] R. Milner. A calculus of communicating systems. *Lecture Notes in Computer Science*, 92.

[13] A.J. van der Schaft and J.M. Schumacher. *An Introduction to Hybrid Dynamical Systems*, volume 251 of *Lecture Notes in Control and Information Sciences*. Springer-Verlag, London, 2000.

[14] E.D. Sontag. *Mathematical Control Theory*, volume 6 of *Texts in Applied Mathematics*. Springer, 1998.

[15] D.A. van Beek and J.E. Rooda. Languages and applications in hybrid modelling and simulation.

[16] V.Bos and J.J.T. Klein. *Formal specification and analysis of industrial systems*. PhD thesis, TU/e, 2002.

[17] M. Vidyasagar. *Nonlinear Systems Analysis*. Prentice Hall, New Jersey, 1993.