Proceedings of the 1997 IEEE International
Conference on Control Applications
Hartford, CT • October 5-7, 1997

Hybrid Control of a Double Tank System

Jörgen Malmborg and Johan Eker
Department of Automatic Control
Lund Institute of Technology
Email: {jorgen|johane}@control.lth.se

Abstract

A hybrid control system consisting of a PID controller, a time-optimal controller and a switching strategy is investigated. A specific controller switching stategy, guaranteeing stability, is tested. To facilitate the real time implementation linear approximations of the optimal switching curves are used.

1. Introduction

In control practice it is quite common to use several different controllers and to switch between them with some type of logical device. One example is systems with selectors which have been used for constraint control for a long time. Systems with gain scheduling, see [2], is another example. Both selectors and gain scheduling are commonly used for control of chemical processes, power stations and in flight control. Other examples of systems with mode switching are used in robotics. Some examples are the systems described in [4].

In spite of their common use there is very little theory available for systems with mode switching. For this reason both analysis and design are often done heuristically. Variable structure systems, [7], [8] and [11] is one approach that is fairly well established. Discrete event dynamical systems is another approach. It is well known that hybrid systems are difficult to analyze. Nevertheless they are used more and more. The reason for this is that they give better performance than ordinary systems and that they can solve problems that can not be dealt with by conventional control.

In process control it is common practice to use PI control for steady state regulation and to use manual control for large changes. In this case it seems very natural to try to combine the steady state regulation with a minimum time controller for the set point changes. Such a controller can be designed very elegantly using the results of this paper.

Since systems with mode switches appear in so many contexts there is no unified approach. Different ways to deal with them are scattered in many application areas. They also appear under many different names. A characteristic feature of these systems is that they represent a mixture of ordinary differential equations and logic. The systems used in this paper is a special case of the Differential Automata described in [10]:

$$\begin{aligned} \dot{x} &= f(x(t), q(t)), \quad x \in \mathcal{R}^n \\ q(t) &= \nu(x(t), q(t^-)), \quad q \in \mathcal{Z}^{m+} \end{aligned} \tag{1}$$

where x denotes the continuous and q the discrete variables. The model does not allow for autonomous or controlled jumps. Such systems have also recently attracted attention from computer scientists. A typical tendency is that the computer scientists focus on the logic and treat the continuous aspects quite cavalierly while the control engineers do the opposite. A good approach should probably take a more balanced view.

2. Preliminaries

In this paper we will design a hybrid controller consisting of two simple controllers, one PID controller and one time-optimal controller. We will show that the use of this hybrid controller will lead to better performance. We will get a control system giving good responces to new set points and also good disturbance rejection.

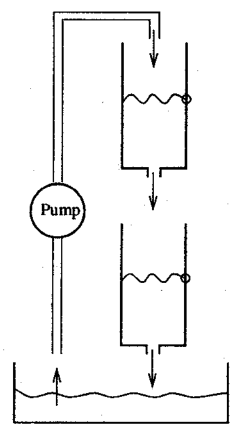

Figure 1: The double tank process

The process to be controlled, both in simulation and by the real time system consists of two water tanks in series, see Figure 1. We would like to control the level of the lower tank and indirectly the level of the upper tank.

The two tank levels are both measurable. Chosing the level of tank i as state x_i we get the state space description

$$\dot{x} = f(x,u) = \begin{bmatrix} -\alpha_1\sqrt{x_1} + \beta u \\ \alpha_1\sqrt{x_1} - \alpha_2\sqrt{x_2} \end{bmatrix}. \tag{2}$$

Where the inflow u is our control variable. The inflow can never be below zero and the maximum flow is $\bar{u} = 27 \cdot 10^{-6}\ m^3/s$. Furthermore, in our experimental setting the outflow areas are the same, giving $\alpha_1 = \alpha_2$.

2.1. Subcontroller design

As mentioned above we will use two subcontrollers together with a supervisory switching scheme. The time-optimal controller is used when the states are far away from the reference point. Coming closer the PID controller will automatically be switched in to replace the time optimal controller. At each different set point we will redesign the controller, keeping the same structure but using reference point depending parameters. Figure 2 describes the algorithm with a Grafcet. For details on Grafcet see [5]. The Grafcet for the tank controller consists of four states. Initially the controller is off. This is the Init state. Opt is the state where the time optimal controller is active and PID is the state for the PID controller. The Ref state is an intermediary state used for calculating new controller parameters before switching to a new time optimal controller.

The subcontroller designs are based on a linearized version of Equation (2).

$$\dot{x} \;=\; \begin{bmatrix} -a & 0 \\ a & -a \end{bmatrix} x + \begin{bmatrix} b \\ 0 \end{bmatrix} u \tag{3}$$

In this linearized equation the parameter b has absorbed the factor $27 \cdot 10^{-6}$ and the new control variable u is in $[0,1]$. The parameters a and b are functions of α, β and the linearization level. We will later see how the neglected nonlinearities affect the performance of the method proposed in this paper. To be able to switch in the PID controller a fairly accurate knowledge of the parameters is needed.

2.2. PID controller design

We use a standard PID controller on the form

$$G_{PID} = K(1 + \frac{1}{sT_I} + sT_d).$$

The design of the PID controller parameters K, T_d and T_I is based on the linear second order transfer function,

$$G(s) \;=\; \frac{ab}{(s+a)(s+a)}$$

derived from Equation (3). The desired closed loop characteristic equation is

$$(s + \alpha\omega)(s^2 + 2\zeta\omega s + \omega^2).$$

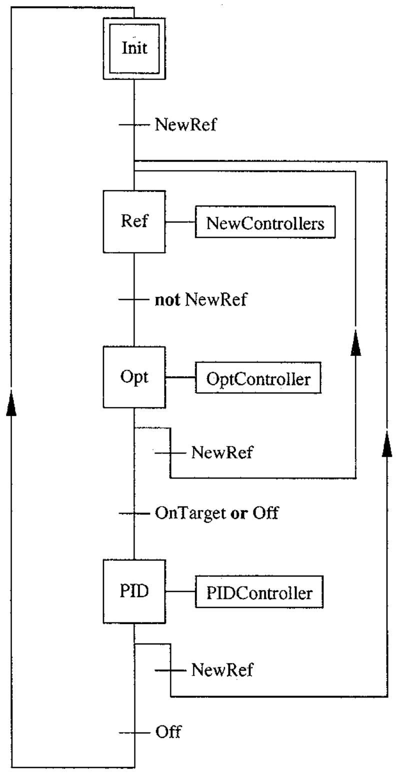

Figure 2: A Grafcet describing the control algorithm.

The parameters α, ω and ζ (1.0, 0.06 and 0.7) are chosen such that we get a reasonable behavior, both in case of set point changes and under load disturbances. For some systems it is possible to get a smaller overshoot by set point weighting. Figure 4 shows the set point and load disturbance responces for the PID controller. When implementing the real-time version of the PID algorithm a filter is used on the derivative part.

2.3. Time optimal controller design

The time optimal controller will bring the system as fast as possible from one set point to another. The Pontryagin maximum principle is used to prove that the time optimal control strategy for the System (2) is of bang-bang nature. The time optimal control is the solution to the following optimization problem

$$\max J = \int_0^T -1 \cdot dt \tag{4}$$

under the constraints:

$$\begin{aligned} x(0) &= \begin{bmatrix} x_1^0 & x_2^0 \end{bmatrix}^T \\ x(T) &= \begin{bmatrix} x_1^R & x_2^R \end{bmatrix}^T \\ u &\in [0,1] \end{aligned}$$

The Hamiltonian, $H(x,u,\lambda)$, for this problem is

$$H = -1 + \lambda_1(-a\sqrt{x_1} + bu) + \lambda_2(a\sqrt{x_1} - a\sqrt{x_2}),$$

with the adjoint equations, $\dot{\lambda} = -\frac{\partial H}{\partial x}$,

$$\dot{\lambda} = \begin{bmatrix} -\frac{a}{2\sqrt{x_1}} & \frac{a}{2\sqrt{x_1}} \\ 0 & -\frac{a}{2\sqrt{x_2}} \end{bmatrix} \lambda. \tag{5}$$

To derive the control signal we do not need the complete solution to these equations. It is sufficient to note that the solutions to the adjoint equations are monotonous. This together with the switching function

$$\sigma = \lambda_1 b u$$

results in the following possible optimal control sequences

$$\langle 0, 1\rangle, \langle 1, 0\rangle, \langle 0\rangle, \langle 1\rangle$$

The swiching times are determined by the new and the old set points. In practice we prefer to have a feedback loop instead of precalculated switching times. Hence we need an analytical solution for the switching curves. For the linear case it is possible to derive this solution.

$$x_2(x_1) = \frac{1}{a}[(ax_1 - b\bar{u})(1 + \ln(\frac{ax_1^R - b\bar{u}}{ax_1 - b\bar{u}})) + b\bar{u}]$$

where $\bar{u}$ takes values in $\{0, 1\}$.

The fact that the nonlinear system has the same optimal control structure as the linearized system makes it possible to simulate the nonlinear switching curves and to compare them with the linear switching curves. Note that the linear and the nonlinear switching curves are quite close for our double tank model, see Figure 3. The diagonal line is the set of equilibrium points, $x_1^R = x_2^R$. Figure 3 shows that the linear switching curves are always below the nonlinear switching curves. This will cause the time optimal controller to switch either to late or to soon.

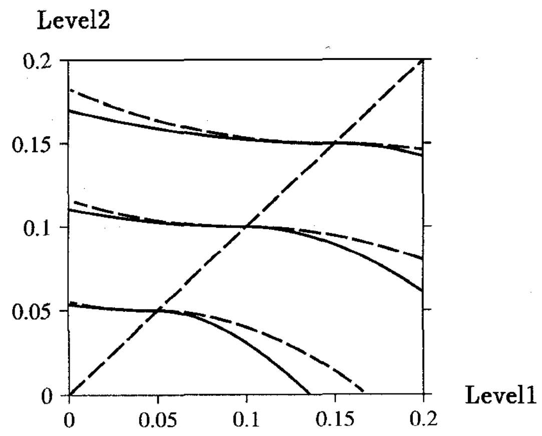

Figure 3: Linear (full) and nonlinear (dashed) switching curves

It is not necessary to use or know the exact nonlinear switching curves since the time optimal controller is only used to bring the system close to the new set point. When sufficiently close the PID controller will take over.

3. Stabilizing Switching Schemes

It is well known that switching between stabilizing controllers may lead to an unstable closed loop system. It is therefore neccessary to have a switching scheme that guarantees stability. Consider the system

$$\dot{x} = f(x, t, u_i) \tag{6}$$
$$u_i = c_i(x, t) \tag{7}$$

where the $c_i(x,t)$ represent different controllers. In a hybrid control system we would like to switch to different controllers in different regions of the state space or in different operating modes. There exist some switching schemes that guarantee stability. One of these is the min-switch strategy described in [9]. Here, a number of stabilizing controllers, c_i, are designed for System (6). For each controller c_i an operating region Ω_i is defined and a Lyapunov function V_i is derived. At every moment the supervisor selects the controller with the smallest value of its Lyapunov function. The controllers can be of different types and they need not share the same state space.

Definition 1 (Min-Switching Strategy)
Let $f_i(x,t)$ be the right-hand side of Equation (6) when control law c_i is used. Use a control signal u^* so that,

$$\dot{x} = f(x, t, u^*) = \sum_{i=1}^{n} \alpha_i f_i(x, t) \tag{8}$$

where $\alpha_i \geq 0$ satisfies $\sum \alpha_i = 1$ and where $\alpha_i = 0$ if either $x \notin \Omega_i$ or if $V_i(x,t) > \min_j[V_j(x,t)]$.

Notice that the α_i's are not unique. We have the following result.

Theorem 1 (Stability of Hybrid Systems)
Let the system be given by Equation (6). Introduce W as

$$W = \min(V_1, V_2, \ldots, V_n)$$

The closed loop system is stable with W as a non-smooth Lyapunov function if the min-switch strategy is used.

Proof See [9] for the proof.

3.1. Lyapunov function modifications

From a control designer's point of view the design of an hybrid control scheme using the *min-switching strategy* can be reduced to separate designs of n different control laws and their corresponding Lyapunov functions. To improve performance it is often convenient to change the location of the switching surfaces. This can, to some degree, be achieved by different transformations of the Lyapunov functions. One example is transformations of the form

$$\tilde{V}_i = g_i(V_i) \tag{9}$$

where $g_i(\cdot)$ are monotonously increasing functions.

In some cases there can be very fast switching, chattering, between two or more controllers having the same value of their respective Lyapunov function. One way to avoid this is to add a constant Δ to the Lyapunov functions that are switched out and subtract Δ from the Lyapunov functions that are switched in. This works as a hysteresis function.

4. Simulations

In this section we will evaluate some different switching methods. In all simulations we will use a switching surface for the time optimal controller based on the linearized equations.

All simulations have been done in the Omola/Omsim environment [1], which supports the use of hybrid systems.

4.1. Pure time optimal and pure PID control

This first simulation set, Figure 4, shows control of a linearized system using either a time optimal controller or a PID controller. Note that PID control gives a large overshoot. The time optimal controller works fine until the level of the lower tanks reaches its new set point. Then the control signal starts to chatter between its minimum and maximum value.

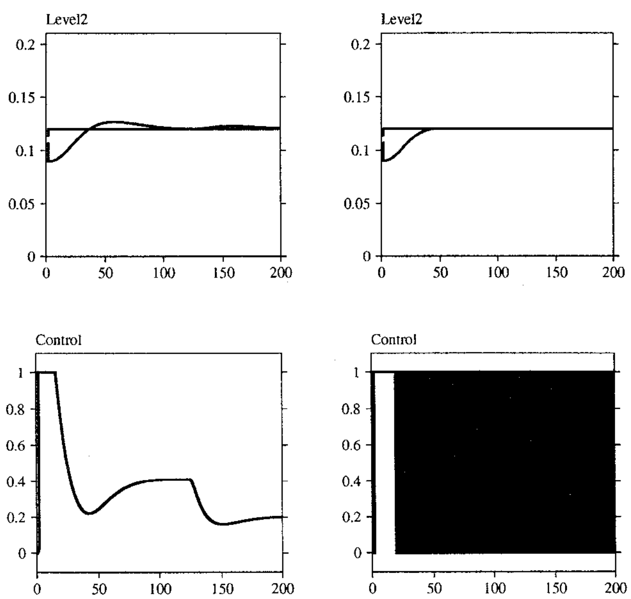

Figure 4: Pure PID (left) and pure time optimal control (right)

4.2. A natural, simple swithing strategy

A natural switching strategy would be to pick the best parts from both PID control and time optimal control. One way to accomplish this is to use the time optimal controller when far away from the equilibrium point and the PID controller when coming closer. As a measure of closeness we use the function

$$V_{close} \;=\; \begin{bmatrix} x_1^R - x_1 \\ x_2^R - x_2 \end{bmatrix}^T P(\theta,\gamma) \begin{bmatrix} x_1^R - x_1 \\ x_2^R - x_2 \end{bmatrix}$$

$$P(\theta,\gamma) = \gamma_1 \begin{bmatrix} \cos^2\theta + \gamma_2 \sin^2\theta & (1-\gamma_2)\sin\theta\cos\theta \\ (1-\gamma_2)\sin\theta\cos\theta & \sin^2\theta + \gamma_2\cos^2\theta \end{bmatrix}$$

The switching strategy here is to start with the time optimal controller and then switch to the PID controller when $V_{close} < \rho$. With the γ and θ parameters we can change the size and shape of the catching region. In this simulation set we do not allow switching back to the time optimal controller until there is a new reference value. See Figure 2 for a graphical description of the algorithm. The simulation results, Figure 5, show how the best parts from the subcontrollers are used to give very good performance.

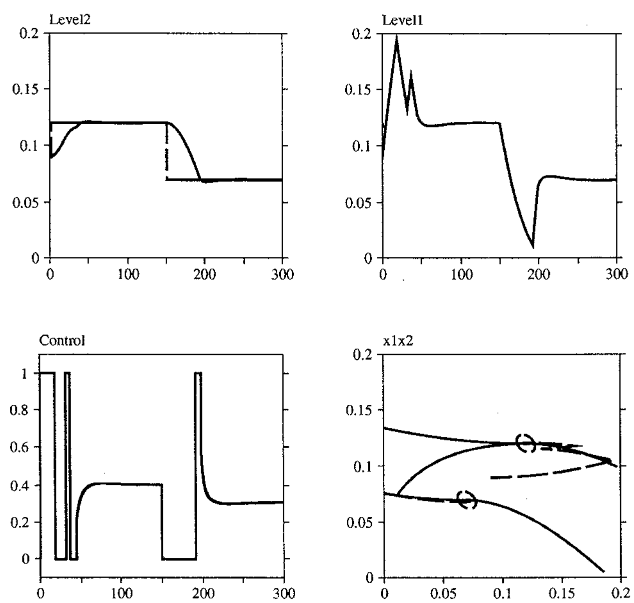

Figure 5: Simulation of the natural, simple swithing strategy. Catching regions are shown in lower right subfigure.

4.3. Lyapunov based switching

In this third simulation set we will use the *min switching strategy* that guarantees stability for the linearized system. We define two Lyapunov functions

$$V_{PID} = \begin{bmatrix} x_1^R - x_1 \\ x_2^R - x_2 \\ x_3^R - x_3 \end{bmatrix}^T P(\theta,\gamma) \begin{bmatrix} x_1^R - x_1 \\ x_2^R - x_2 \\ x_3^R - x_3 \end{bmatrix}$$

$$V_{TO} = \textit{time left to reach new set point}$$

$$P(\theta,\gamma) = \gamma_1 \begin{bmatrix} \cos^2\theta + \gamma_2 \sin^2\theta & (1-\gamma_2)\sin\theta\cos\theta & 0 \\ (1-\gamma_2)\sin\theta\cos\theta & \sin^2\theta + \gamma_2\cos^2\theta & 0 \\ 0 & 0 & \gamma_3 \end{bmatrix}$$

The state x_3 is the integral state in the PID controller. x_3^R is its steady state value. As in the previous simulation set we can use the γ and θ parameters to shape the catching region. The new state x_3 is preset to its value at the new equilibrium point, i.e. x_3^R, any time there is a set point change. We also choose only to update this state after the first switch to PID control. Using this method we get a similar two-dimensional catching region as in the previous simulation set.

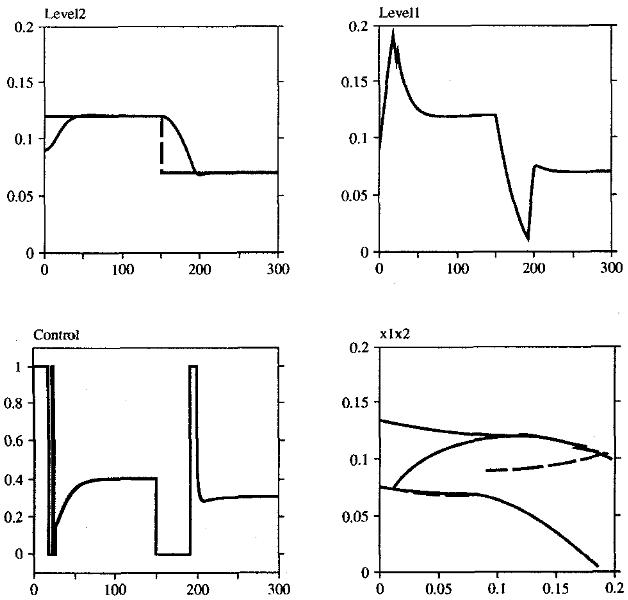

Figure 6: Lyapunov based switching

This supervisory scheme may lead to two types of chattering behavior. One is due to the nonlinearities. The nonlinear switching curve lies above the linear, see Figure 3. That causes the trajectory of the nonlinear system to cross the linear switching curve. One way to remove this problem is to introduce a hysteresis function for going from minimum to maximum control signal in the time optimal controller. There can also be chattering between the PID and the time optimal controller if their Lyapunov functions have the same value. One solution to this problem is to add and remove the constant Δ as discussed in the section on Lyapunov functions modifications. The simulation results can be seen in Figure 6.

5. Implementation Issues

The traditional way of implementing real-time systems using languages such as C or ADA gives very poor support for algorithms expressed in a state machine fashion. The need for a convenient way to implement hybrid systems is evident. The programming language must allow the designer to code the controller as a state-machine, as a period process, or as a combination of both. The later alternative is especially useful in the case where the controller is divided into several controllers, with some common code. A typical example of this, is a state feedback controller for a plant, where it is not possible to measure all states. To get information about the nonmeasurable states an observer or a filter can be used. Typically this information is needed by the whole set of controllers, and thus the controller itself can be implemented as a hybrid system, consisting of one global periodic task which handles the filtering of process data, and a set of controllers which can be either active or inactive.

In this paper we use the PAL-language, [3], to describe our algorithms. PAL is a dedicated control algorithm language, which supports both period and sequential algorithms, or a mixture of them. Furthermore, the language supports data-types such as polynomials and matrices, which are extensively used in control theory.

Control algorithms coded in PAL, are executed in a run-time environent called PALSJÖ, [6]. PALSJÖ is a framework for real-time control systems, which allows on-line configuration. All experiments in this paper are implemented using PALSJÖ and PAL.

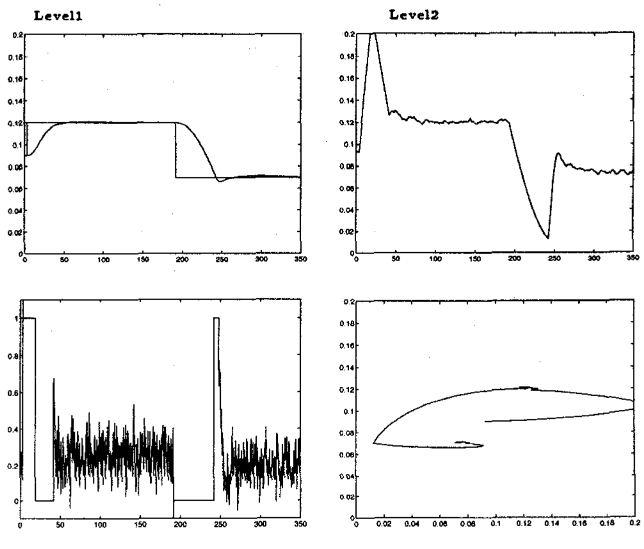

Figure 7: Lab experiment

6. Experiments

The theory and the simulations are verified by experiments. For simplicity only the switching strategy in Sec. 4.2. is implemented. Figure 7 shows the results of that experiment with the double tanks.

The measurements from our lab process have a high noise level as can be seen in Figure 7 . We used a first order low-pass filter

$$G_f(s) = \frac{1}{s+1}$$

to eliminate some of it. To further reduce the impact of the noise we added a filter to the derivative part of the PID controller in a standard way.

The parameters in the simulation model were chosen to match the parameters of our lab process. It's is thus possible to compare the experimental results directly with the simulations. Comparing Figure 5 and Figure 7 shows the close correspondence between the simulation and experimantal results.

During experiments we found that the difference between the linear and the nonlinear switching curves were not so important. However, we need a good model of the static gains in the system. If there is a large deviation it cannot be guaranteed that the equilibrium points are within the catching regions.

7. Summary

We have shown that a hybrid controller, consisting of a time optimal controller together with a PID controller gives very good performance.

The controller is easy to implement. It gives, in one of its forms, a guaranteed closed loop stability.

It is fairly easy to combine this method with a tuning experiment that gives a second order model of the system. Doing so, we automatically get a PID controller and an approximate time optimal controller.

The use of linear switching curves works well for the double tank model. For systems with more dominating nonlinearities it might be necessary to compensate the switching curves.

The suggested hybrid controller solves a practical and frequent problem. Many operators go to manual control when handling start-up and set point changes. The authors are currently involved in a full scale prototype implementation of a heating-ventilation system together with Diana Control AB.

References

[1] Mats Andersson. *Object-Oriented Modeling and Simulation of Hybrid Systems.* PhD thesis, December 1994.

[2] Karl Johan Åström and Björn Wittenmark. *Adaptive Control.* Addison-Wesley, Reading, Massachusetts, second edition, 1995.

[3] Anders Blomdell. The pålsjö algorithm language. Master's thesis, Department of Automatic Control, Lund Institute of Technology, 1997.

[4] R. W. Brockett. Hybrid models for motion control systems. In H. L Trentelman and J. C. Willems, editors, *Essays on Control: Perspectives in the Theory and its Application.* Birkhäuser, 1993.

[5] R. David and H. Alla. *Petri Nets and Grafcet: Tools for modelling discrete events systems.* Prentice-Hall, 1992.

[6] J. Eker and Anders Blomdell. A structured interactive approach to embedded control. In *4th Intelligent Robotic System*, pages 191–197, Lisbon, Portugal, July 1996.

[7] S. V. Emelyanov. *Variable Structure Control Systems.* Oldenburger Verlag, Munich, FRG, 1967.

[8] U. Itkis. *Control Systems of Variable Structure.* Halsted Press, Wiley, New York, 1976.

[9] Jörgen. Malmborg, B. Berhardsson, and K. J. Åström. A stablizing switching scheme for multcontroller systems. In *Proceedings of the 1996 Triennal IFAC World Congress, IFAC'96*, volume F, pages 229–234, San Francisco, California, USA, July 1996. Elsevier Science.

[10] L. Tavernini. Differential automata and their discrete simulators. *Nonlinear Analysis, Theory, Methods and Applications*, 11(6):665–683, 1987.

[11] V. I. Utkin. Variable structure systems with sliding modes. AC-22:212–222, 1977.