# MODEL PREDICTIVE CONTROL - IDEAS FOR THE NEXT GENERATION

**A. Bemporad, G. Ferrari-Trecate, D. Mignone, M. Morari, F. D. Torrisi**

*Institut für Automatik*
*ETH Swiss Federal Institute of Technology*
*CH-8092 Zürich, Switzerland*
*tel.+41-1-632 7626 fax +41-1-632 1211*
{bemporad,ferrari,mignone,morari,torrisi}@aut.ee.ethz.ch
http://www.control.ethz.ch

**Keywords:**† Hybrid Systems, Moving Horizon Control

## Abstract

Mixed Logical Dynamical (MLD) systems are introduced as a new system type. The MLD form is capable to model a broad class of systems arising in many applications: linear hybrid systems; sequential logical systems (finite state machines, automata); nonlinear dynamic systems, where the nonlinearity can be expressed through combinational logic; some classes of discrete event systems; constrained linear systems. Controllability/verification and observability of MLD systems and other system theoretic properties are defined. Tests for these properties are formulated in the form of Mixed-Integer Linear Programs. Moving horizon control and estimation strategies with stability guarantees are proposed. These strategies require the iterative solution of Mixed-Integer Quadratic Programs. Several examples communicate the power and versatility of the proposed framework.

## 1 Introduction

Most control theory and tools have been developed for systems, whose evolution is described by smooth linear or nonlinear state transition functions resulting, for example, from differential or difference equations. In many applications, however, the systems include discrete components, such as on/off switches or valves, gears or speed selectors. Discrete characteristics are also often introduced by the control system or the specifications which are expressed by a series of if-then-else rules. Such systems consisting of continuous and discrete "components" are commonly referred to as *hybrid systems*. Hybrid systems arise in a large number of application areas, but our understanding of these systems is rather limited at present. In practice the control of hybrid systems is left to schemes based on heuristic rules inferred from practical plant operation. For the time being, the most common analysis tool is exhaustive simulation.

The premise of the work described in this paper is that all questions and problems related to hybrid systems are inherently difficult because of their combinatorial nature. Consequently all *useful* techniques must involve significant off-line and/or on-line computation. Against this background we introduce a new system type, Mixed Logical Dynamical (MLD) systems. We argue that many practical problems can be represented in MLD form. Control, estimation, and verification of MLD systems require the solution of Mixed-Integer Linear (or Quadratic) Programs (MILPs or MIQPs). Because efficient techniques not only for MILPs but also for MIQPs are becoming available, computation power is increasing, and mixed integer problems can be efficiently parallelized, this new approach holds much promise for tackling realistic size problems.

## 2 Mixed Logical Dynamic (MLD) Systems

Any modeling framework for hybrid systems must be a compromise which circumvents some of the complexities and leads naturally to the formulation of analysis and controller synthesis techniques which are manageable for practical problems. Our formulation is motivated by the following considerations.

- Limiting the formalism to discrete time is not overly restrictive from a practical point of view because of the sampled-data nature of the control systems, which determine the evolution of these hybrid systems.
- We restrict the dynamics to be linear with the

†The full version of this paper will appear in at-Automatisierungstechnik

exception that some of the state variables are binary. This greatly simplifies the analysis, but nevertheless permits the description of a broad class of systems.

By following standard notation (Williams 1977, Cavalier *et al.* 1990, Williams 1993), we adopt capital letters $X_i$ to represent statements, e.g. "$x \geq 0$" or "Temperature is hot". $X_i$ is commonly referred to as a *literal*, and has a *truth value* of either "T" (true) or "F" (false). Boolean algebra enables statements to be combined in compound statements by means of *connectives*: "$\wedge$" , "$\vee$", "$\sim$" , etc. One can associate with a literal $X_i$ a *logical variable* $\delta_i \in \{0,1\}$, which has a value of either 1 if $X_i$ =T, or 0 otherwise.

As we are interested in systems which have both logic and dynamics, we wish to establish a link between the two worlds. In particular, we need to establish how to build statements from operating events concerning physical dynamics. The key idea is to use techniques described, for example, in (Williams 1993, Cavalier *et al.* 1990, Raman and Grossmann 1992) to transform propositional logic into *mixed-integer linear inequalities*, i.e. linear inequalities involving both *continuous variables* $x \in \mathbb{R}^n$ and *binary/logical variables* $\delta \in \{0,1\}$.

The resulting Mixed Logical Dynamic (MLD) Systems are described through the following linear relations

$$x(t+1) = Ax(t) + B_1u(t) + B_2\delta(t) + B_3z(t) \quad (1a)$$

$$y(t) = Cx(t) + D_1u(t) + D_2\delta(t) + D_3z(t) \quad (1b)$$

$$E_2\delta(t) + E_3z(t) \leq E_1u(t) + E_4x(t) + E_5 \quad (1c)$$

where

$$x = \begin{bmatrix} x_c \\ x_\ell \end{bmatrix}, \; x_c \in \mathbb{R}^{n_c}, \; x_\ell \in \{0,1\}^{n_\ell}, \; n \triangleq n_c + n_\ell$$

is the state of the system, with the $x_c$ components continuous and the $x_\ell$ components 0-1 . The outputs $y$ and the inputs $u$ are partitioned similarly. The auxiliary logical and continuous variables are represented by $\delta \in \{0,1\}^{r_\ell}$ and $z \in \mathbb{R}^{r_c}$, respectively.

The justification for the MLD form is that it is capable to model a broad class of systems arising in many applications: linear hybrid systems; sequential logical systems (finite state machines, automata); nonlinear dynamic systems, where the nonlinearity can be expressed through combinational logic; some classes of discrete event systems; constrained linear systems. Here the terms "combinational" and "sequential" are borrowed from digital circuit design jargon. More importantly, the MLD formalism leads to the formulation of various verification, control and estimation problems in terms of MILPs or MIQPs, for which efficient algorithms are available. These problems have not been successfully addressed by other tools or only with a much higher computational effort. In a sense the ends justify the means here.

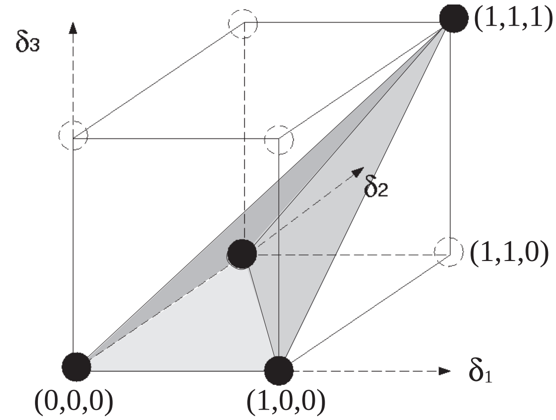


Figure 1: Convex hull of the rows of the truth table of $X_3 = X_1 \wedge X_2$

In (Bemporad and Morari 1999) some examples are described of basic systems that can be expressed in the MLD form, such as linear systems with output nonlinearities, discrete inputs, qualitative outputs, bilinear systems, piece-wise linear systems, and automata driven by events on continuous dynamics.

## 2.1 Transformation into MLD Form

The transformation of propositional logic problems into equivalent sets of linear inequalities is not unique. For instance in (Cavalier *et al.* 1990) the approach above is compared with the approach which utilizes *conjunctive normal forms* (CNF). It is clear that a proper processing of the propositional logic problem might produce large benefits for the numerical solution of the mixed integer program, which results in the analysis as well as controller and estimator synthesis problems for MLD systems.

Recently, we succeeded (Mignone *et al.* 1999) in developing the following very effective method which generates a set of linear inequalities corresponding to any complex logical expression without introducing any auxiliary variables. First, for each expression $X_n = F(X_1, X_2, \ldots X_{n-1})$ the *truth table* is calculated, showing the result $X_n$ for each possible combination of values for $X_1, X_2, \ldots$ row by row. We proved (Mignone *et al.* 1999) that the polytope $P$ obtained as the convex hull of the points defined by the rows of the truth table describes the logical expression with a minimal number binary variables. For example, the four rows of the truth table for $X_1 \wedge X_2 = X_3$ define the points and the convex hull in Fig. 1. Even though the generation of the truth table and the computation of the convex hull is time consuming, it can be performed offline.

Many algorithms and computer codes exist for determining the convex hull from a general set of points. For a detailed survey of these packages, the reader is referred to `http://www.geom.umn.edu/software/cglist/ch.html`. Modifications are necessary for our problem, because often the points defined by the truth table lie in a proper subspace, a case not dealt with by these general algorithms.

The transformation of first principles hybrid system descriptions into MLD form requires the application of a set of given rules, like the transformation technique just described. It is lengthy and tedious and is therefore a task that is preferably automated. Therefore a compiler has been developed (Anlauff *et al.* 1999) that produces the matrices $A$, $B_i$, $C$, $D_i$ and $E_i$ in (1). The problem specification language to the compiler is HYSDEL (HYbrid System DEscription Language).

# 3 Theoretical Properties of Mixed Logical Dynamic Systems

In principle, the inequality (1c) might be satisfied for many values of $\delta(t)$ and/or $z(t)$. In order to define *trajectories* in the $x$ and $y$-space for system (1), we wish that $x(t+1)$ and $y(t)$ are uniquely determined by $x(t)$ and $u(t)$, i.e., that the system is *well posed* (for a formal definition see (Bemporad and Morari 1999)). Typically, when the model derives from a real system, there is no need of checking for well-posedness. Because the transformation into MLD form is not unique it is conceivable that the MLD system is not well posed. Therefore, a simple numerical test for checking this property has been developed and is reported in (Bemporad and Morari 1999). It is based on a feasibility check of an MILP.

Needless to say, well-posedness is a *minimal* requirement for the MLD description to be meaningful. For control, however, reachability and controllability must also be understood. For the construction of estimation procedures, reconstructibility and observability are important properties. Finally, we have to define what we mean by stability and we need a technique for assessing stability for a given MLD system.

These questions are inherently difficult. Consider the problem of reaching a target set from a set of initial conditions (reachability/controllability). In the hybrid systems literature, this is exactly what is called a *formal verification* problem, which has been shown to be undecidable in general (Alur *et al.* 1993, Kesten *et al.* 1993).

However, some progress has been made in addressing these questions (Bemporad *et al.* 1999*c*). To illustrate the ideas and the unusual behavior which can occur we will briefly discuss our work on observability here.

We adopted the concept of incremental observability from (Keerthi and Gilbert 1988, Rao and Rawlings 1998). Incremental observability must be tested on a case-by-case basis. No structural properties have emerged at this point. Even for a piece-wise linear systems, a special case of an MLD system, the "observability index" is not related to the order of the constituting linear systems as is the case for LTI systems. Also the combination of observable LTI systems into a piecewise linear system is not necessarily observable. The reverse does not hold either. The combination of LTI systems which are by themselves not observable may be observable (Bemporad *et al.* 1999*c*).

# 4 Control

## 4.1 Optimal Control of MLD Systems

For an MLD system of the form (1), consider the following problem. Given an initial state $x_0$ and a final time $T$, find (if it exists) the control sequence $u_0^{T-1} \triangleq \{u(0), u(1), \dots, u(T-1)\}$ which transfers the state from $x_0$ to $x_f$ and minimizes the *performance index*

$$J(u_0^{T-1}, x_0) \triangleq \sum_{t=0}^{T-1} \|u(t)-u_f\|_{Q_1}^2 + \|\delta(t,x_0,u_0^t)-\delta_f\|_{Q_2}^2 + \|z(t,x_0,u_0^t)-z_f\|_{Q_3}^2 + \|x(t,x_0,u_0^{t-1})-x_f\|_{Q_4}^2 + \|y(t,x_0,u_0^{t-1})-y_f\|_{Q_5}^2 \tag{2}$$

subject to

$$x(T, x_0, u_0^{T-1}) = x_f \tag{3}$$

and the MLD system dynamics (1a), where $\|x\|_Q^2 \triangleq x'Qx$; $Q_i = Q_i' \geq 0$, $i = 1, \dots, 5$, are given weight matrices, and $x_f$, $u_f$, $\delta_f$, $z_f$, $y_f$ satisfy (1) in steady state for $x(t+1) = x(t) = x_f$.

This problem can be solved as a *Mixed-Integer Quadratic Program* (MIQP).

## 4.2 Predictive Control

It is interesting both from a theoretical and a practical point of view to ask whether an MLD system can be stabilized to an equilibrium state or can track a desired reference trajectory via feedback control. Finding such a control law is not easy, because the system is neither linear nor even smooth. *Model predictive control* (Garcia *et al.* 1989) provides tools to succeed in this task. In brief, one has to solve an optimization problem of the form (2)–(3) at each time step $t$, by finding an optimal input sequence $\{u^*(t+k)\}_{k=0,\dots,T-1}$. Then, only the first move is applied to the plant, i.e. $u(t) = u^*(t+0)$, and the whole optimization procedure is repeated at time $t+1$, when new measurements $x(t+1)$ are available.

By appropriately defining the concepts of equilibrium and stability for MLD systems, and by using Lyapunov arguments it can be proven (Bemporad and Morari 1999) that the control law, obtained by repeatedly solving (2)–(3) at each time step $t$, stabilizes the system.

From the proof it follows that suboptimal solutions do not not affect stability, although the performance deteriorates. This is particularly appealing when the available computational power does not allow the full solution of the MIQP problem (2)–(3).

# 5 Moving Horizon Estimation for MLD Systems

As shown above, model predictive controllers can be synthesized for MLD systems (1). The method requires the solution of an MIQP at each sample time. The dual problem, i.e. the moving horizon estimation problem can also be formulated in terms of an MIQP. The goals of such an estimation can be varied, like state estimation, fault detection or disturbance estimation. The common feature in all these problems is the minimization of a quadratic cost function involving the quantities to be estimated. Contrary to the control problem, the estimation horizon extends backwards in time, allowing at time $t$ to estimate the quantities of interest at times prior to $t$.

Many techniques for fault detection are modeling faults as additive unknown inputs affecting a linear system. Fault detection is then equivalent to determining if the estimated inputs exceed a certain threshold value. The MLD system framework allows the designer the formulation of more realistic fault detection problems. Faults can also be modeled as unmeasured binary disturbances affecting the system in a multiplicative manner. A faulty actuator, for example, is represented much more accurately in this manner.

The dynamics of the system in the presence of each fault is assumed to be known. After the described transformation steps to bring the system into the MLD form, one finds that for fault detection the MLD model (1) should include three unmeasured additive inputs: binary faults, input disturbances, and output disturbances. Under certain mild assumptions the stability of the estimator can be guaranteed (Bemporad *et al.* 1999*b*).

# 6 Computational Aspects

One drawback of the methods summarized in this paper lies in the complexity of the MILPs and MIQPs that must be solved. These types of optimization problems exhibit an exponential increase of the worst case complexity with an increasing number of binary variables. However, this does not necessarily preclude the application of the method. For instance, if we use branch and bound methods (considered widely to be best for these types of problems (Fletcher and Leyffer 1995)) the solution time can vary considerably according to the tree exploring strategy and the branching variable selection rule.

We have experimented successfully with a strategy which assumes that the binary variables change only infrequently over the considered time horizon (Bemporad *et al.* 1999*a*). This assumption is particularly good, when the binary variables represent faults that do not occur very often and that are usually not recoverable. Finally, as mentioned above, for control purposes it is not critical to find the global optimum to guarantee stability, a feasible suboptimal solution suffices.

CPLEX (ILO 1997) is the most widely used commercial code for solving MILPs. We have used the research codes by Fletcher and Leyffer (Fletcher and Leyffer 1995) and by Sahinidis (Ryoo and Sahinidis 1996) to solve the MIQPs arising in the optimal control and estimation problems. There the key is to pair an efficient sparse QP code for the relaxed problems with a good tree exploring strategy.

# 7 Examples

The following examples are chosen to communicate the power and versatility of the proposed framework. More examples can be found in (Bemporad and Morari 1999) and (Bemporad *et al.* special issue at ECC 1999*d*).

## 7.1 Predictive Control

Consider the following piecewise linear system with input and state constraints:

$$\begin{aligned}
x(t+1) &= 0.8\begin{bmatrix} \cos\alpha(t) & -\sin\alpha(t) \\ \sin\alpha(t) & \cos\alpha(t) \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t) \\
y(t) &= [1\ 0]x(t) \\
\alpha(t) &= \begin{cases} \frac{\pi}{3} & \text{if} \quad [1\ 0]x(t) \geq 0 \\ -\frac{\pi}{3} & \text{if} \quad [1\ 0]x(t) < 0 \end{cases} \\
x(t) &\in [-10, 10] \times [-10, 10] \\
u(t) &\in [-1, 1]
\end{aligned} \tag{4}$$

The condition $x_1(t) \geq 0$ can be associated with a binary variable $\delta(t)$ such that

$$[\delta(t) = 1] \ \leftrightarrow \ [x_1(t) \geq 0] \tag{5}$$

and the system (4) can be rewritten in MLD form as described in (Bemporad and Morari 1999).

For this system we applied the model predictive control scheme described in section 4 to solve a regulation problem. In order to stabilize the system to the origin, the feedback control law resulting from the optimization (2) is adopted, along with the horizon $T = 3$, and the steady state values $u_f = 0$, $\delta_f = 0$, $z_f = [0\ 0\ 0\ 0]'$, $x_f = [0\ 0]'$, $y_f = 0$. For details about the interpretation of the auxiliary variables $z_f$ we defer to (Bemporad and Morari 1999). Fig. 2 shows the resulting trajectories for the states, the control action and the binary variable $\delta$. The trajectories obtained by solving the control problem at time $t = 0$ are also shown as thin lines in the plots.

## 7.2 Fault Detection

### 7.2.1 Model of Three Tank System

The three tank system represented in Fig. 3 has been adopted recently as a standard benchmark problem for fault detection and reconfigurable control (Lunze 1998, Berec and Tesař 1997). Here we report a simplified physical description of the system (more details can be

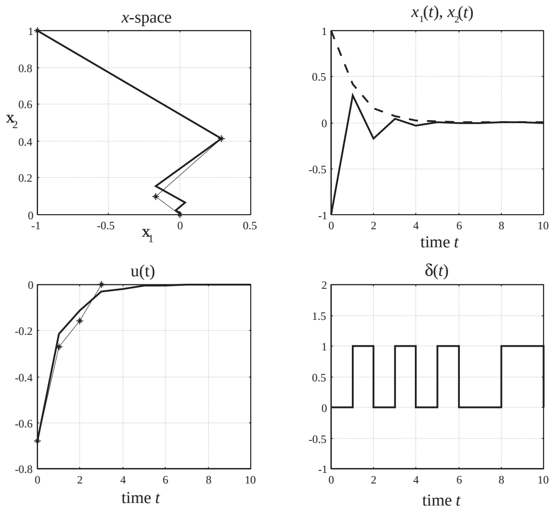


Figure 2: Closed-loop regulation problem for system (4). Closed-loop trajectories (thick lines) and optimal solution at $t = 0$ (thin lines).

found in (Dolanc *et al.* 1997)). We denote by $Q_1$ and $Q_2$ the input flows to the system. $Q_{ijVk}$ are the liquid flows between the tanks, as indicated in Fig. 3. Note for instance, that $Q_{13V1}$ is zero, if both liquid levels $h_1$ and $h_3$ are below the valve height $h_v$. $Q_{N3}$ is the nominal outflow from tank 3 and $Q_{L1}$ is a possible leak of tank 1. From the conservation of mass in the tanks we obtain a set of differential equations.

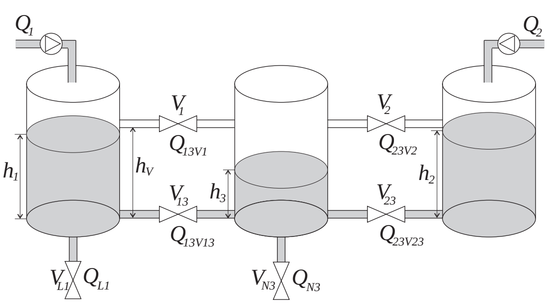


Figure 3: COSY Three-Tank Benchmark.

In this model all valves are of the on-off type. A switching controller for valve $V_1$ is used to keep the liquid level in tank 3 at some desired value. Tank 2 is only used for reconfiguration purposes. The MLD description can be readily derived according to (Bemporad *et al.* 1999*b*). The following two types of faults are considered: The fault $\phi_1$ denotes a leak in tank 1 and the fault $\phi_2$ implies that valve $V_1$ is blocked closed. Note that the failure of valve $V_1$ is modeled as a multiplicative fault. In Fig. 4 we simulated the occurrence of the faults at different times.

Both faults are detected correctly with a few time steps of delay. Note however that during the startup there are a few false alarms of fault $\phi_2$, i.e. blocking of valve $V_1$. These wrongly detected faults are due to the fact,

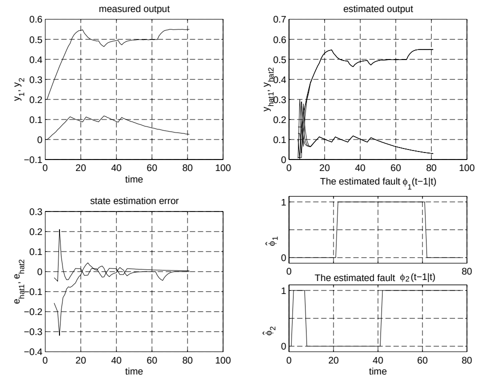


Figure 4: Simulation of a leak in tank 1 ($\phi_1$) from $t = 20$ until $t = 60$, and a blocking valve ($\phi_2$) from $t = 40$ until $t = 80$.

that the level in tank 1 has not yet reached the height of valve $V_1$. Therefore no liquid can pass through $V_1$, which is indistinguishable from a blocked valve $V_1$. To avoid this problem it is very natural to formulate the clause $[h_1 \leq h_v] \Rightarrow \phi_2 = 0$. This is just an additional constraint that can be added to the other constraints of the optimization problem. With this correction, the fault estimates are free of any errors, as can be seen in Fig. 5.

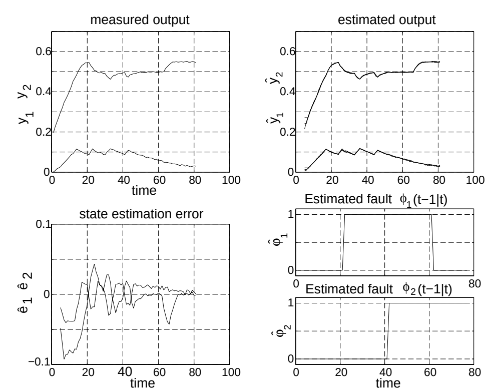


Figure 5: The same simulation as in Fig. 4, with the requirement $[h_1 \leq h_v] \Rightarrow \phi_2 = 0$.

# References


Alur, R., C. Courcoubetis, T.A. Henzinger and P.H. Ho (1993). Hybrid automata: an algorithmic approach to the specification and verification of hybrid systems. In: *Hybrid Systems* (A.P. Ravn R.L. Grossman, A. Nerode and H. Rischel, Eds.). Vol. 736 of *Lecture notes in computer Science*. pp. 209–229. Springer Verlag.

Anlauff, M., A. Bemporad, A. Chakraborty, P. Kutter, D. Mignone, M. Morari, A. Pierantonio and L. Thiele

(1999). From Ease in Programming to Easy Maintenance: Extending DSL Usability with Montages. In: *submitted to Usenix, DSL 99, 2nd conference on domain specific languages, Austin Texas*.

Bemporad, A. and M. Morari (1999). Control of Systems Integrating Logic, Dynamics, and Constraints. *Automatica* **35**(3), 407–427. ftp://control.ethz.ch/pub/reports/postscript/AUT98-04.ps.

Bemporad, A., D. Mignone and M. Morari (1999*a*). An Efficient Branch and Bound Algorithm for State Estimation and Control of Hybrid Systems. *European Control Conference*.

Bemporad, A., D. Mignone and M. Morari (1999*b*). Moving Horizon Estimation for Hybrid Systems and Fault Detection. In: *Proceedings of the American Control Conference*.

Bemporad, A., G. Ferrari-Trecate and M. Morari (1999*c*). Observability and Controllability of Piecewise Affine and Hybrid Systems. *submitted to CDC*.

Bemporad, A., G. Ferrari-Trecate, D. Mignone, M. Morari and F. D. Torrisi (special issue at ECC 1999*d*). A Framework for Control, State Estimation, Fault Detection, and Verification of Hybrid Systems. *Automatisierungstechnik*.

Berec, L. and L. Tesař (1997). Testing fault detection methods via three-tank system. Technical Report Issue A. Copernicus Project CT94-0237.

Cavalier, T.M., P.M. Pardalos and A.L. Soyster (1990). Modeling and integer programming techniques applied to propositional calculus. *Computers Opns Res.* **17**(6), 561–570.

Dolanc, G., D. Juricic, A. Rakar, J. Petrovcic and D. Vrancic (1997). Three-tank Benchmark Test. Technical Report Copernicus Project Report CT94-02337. J. Stefan Institute. ftp://ftp.utia.cas.cz/pub/staff/tesar/fault/latest/.

Fletcher, R. and S. Leyffer (1995). Numerical experience with lower bounds for MIQP branch-and-bound. Technical report. University of Dundee, Dept. of Mathematics. Scotland, U.K. submitted to SIAM Journal on Optimization, http://www.mcs.dundee.ac.uk:8080/~sleyffer/miqp_art.ps.Z.

Garcia, C.E., D.M. Prett and M. Morari (1989). Model Predictive Control: Theory and Practice – a Survey. *Automatica* **25**(3), 335–348.

ILO (1997). *Using the CPLEX Callable Library*. 5.0 ed.

Keerthi, S.S. and E.G. Gilbert (1988). Optimal Infinite-Horizon Feedback Laws for a General Class of Constrained Discrete-Time Systems: Stability and Moving-Horizon Approximations. *Journal of Optimization Theory and Applications* **57**(2), 265–293.

Kesten, Y., A. Pnueli, J. Sifakis and S. Yovine (1993). Integration graphs: a class of decidable hybrid systems. In: *Hybrid Systems* (A.P. Ravn R.L. Grossman, A. Nerode and H. Rischel, Eds.). Vol. 736 of *Lecture notes in computer Science*. pp. 179–208. Springer Verlag.

Lunze, J. (1998). Laboratory Three Tanks System — Benchmark for the Reconfiguration Problem. Technical report. Tech. Univ. of Hamburg-Harburg, Inst. of Control. Eng.. Germany. http://www.tu-harburg.de/rts/software/cosy/.

Mignone, D., A. Bemporad and M. Morari (1999). A Framework for Control, Fault Detection, State Estimation and Verification of Hybrid Systems. *Proceedings of the American Control Conference*.

Raman, R. and I.E. Grossmann (1992). Integration of logic and heuristic knowledge in MINLP optimization for process synthesis. *Computers and Chemical Engineering* **16**(3), 155–171.

Rao, C. V. and J. B. Rawlings (1998). Nonlinear Moving Horizon Estimation. In: *International Symposium on Nonlinear Model Predictive Control: Assessment and Future Directions (Ascona, Switzerland)*.

Ryoo, H. S. and N.V. Sahinidis (1996). A Branch-and-Reduce Approach to Global Optimization. *Journal of Global Optimization* **8**, 107–138.

Williams, H.P. (1977). Logical problems and integer programming. *Bulletin of the Institute of Mathematics and Its Applications* **13**, 18–20.

Williams, H.P. (1993). *Model Building in Mathematical Programming*. John Wiley & Sons. Third Edition.