IMPLICIT RELATIONS AND DISCRETE EVENTS IN PROCESS SIMULATION

David I. Wilson * **Christian Haag** **,[1]

* *Dept of Electrotechnology, Auckland University of Technology, New Zealand*
** *Dept. of Electrical Engineering, Karlstad University, Sweden*

Abstract: Simulation of hybrid systems is considerably more complex than pure continuous-time systems due to blocking states, exceptions due to discrete events, and Zeno-type behavior. Rather than esoteric, hybrid systems are actually the norm for industrial process simulation when including the possibility of tanks running dry, vessels overflowing etc. This paper compares two modelling tools; (one block-diagram, one causal) for a simple, but typical process engineering problem. Matlab/Simulink illustrates a traditional block diagram modelling tool, while MathModelica with its strong emphasis on objects and symbolic manipulation was used to demonstrate a novel approach to modelling.

Keywords: Process simulation, symbolic manipulators, Modelica, DAEs, hybrid

1. INTRODUCTION

Process simulators have long been considered an essential tool in the chemical processing industry. Commercial simulators are now mature products with detailed libraries of unit operations, advanced control options, historians, and quality assurance programs. At their very core, chemical plants are often best described as interconnections of tanks and vessels, partially filled with liquids and it is essential that any simulator can robustly cope with this.

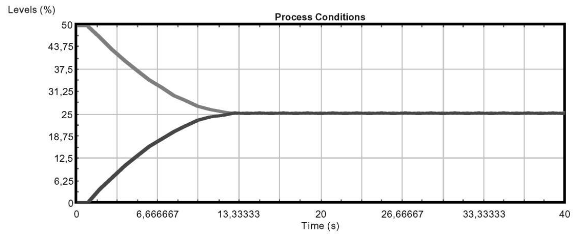
The typical chemical plant model is large, sparse, with a small number of important instances of highly nonlinear relations all with soft and hard constraints. While the importance of constraints and the nonlinearities does depend on the criteria of the simulation scenario under investigation, increasingly it is the unusual operating conditions such as start up and shut downs, recovery from major disturbances that effect the overall economics or where the payoff is to be found.

A good example of an industry that has modest demands for thermodynamics or esoteric unit operations is a paper mill. Here the key intermediate is dilute pulp which has rheological properties similar to water. Perhaps it is due to the low toxicity or the low relative value of the material, but in our experience in the pulp and paper industry, vessels regularly overflowed, or ran dry during operation. For this reason we are interested in simulation tools that reliably handle simple flow dynamics and events such as tanks running dry.

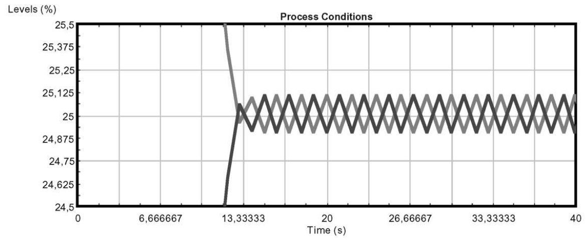
Our motivation to challenge the status quo of modelling tools is a direct result of the mediocre impact of a comprehensive pulp and paper simulation project reported in (Wilson and Balderud, 2000; Haag and Wilson, 2001; Balderud and Wilson, 2003). At the time, we favoured a block-diagram cause and effect modelling paradigm and used the dedicated simulator Simon's Ideas specially tailored for the pulp and paper industry,

[1] Partially supported by the Skogsindustricentrum, Karlstad University, Sweden.

(McGarry *et al.*, 1997). In hindsight we have questioned that decision, and some comparisons and challenges have been raised in (Haag and Wilson, 2003). For example, Fig. 1 illustrates the small, but annoying simulation artifact of persistently oscillating levels in a scenario where we model the level of liquid in two interconnected tanks at the default tolerances and step sizes using a commercial process simulator.



(a) Level trends of two connected tanks when valve is opened.



(b) Zoom version of Fig. 1(a) showing a persistent oscillation.

Fig. 1. Simulation results of levels in two interconnected tanks using IDEAS.

Consequently we were interested in the potential of 'acausal' modelling, perhaps best realised in the product DYMOLA, (Åström *et al.*, 1998). Causal/acausal tools represent an alternative classification for modelling tools from the block-oriented/equation-oriented division traditionally used by say (Marquardt, 1996). Acausal modelling tools combine object oriented thinking with a symbolic manipulator relieving the modeller from having to decide *apriori* information flow directions, i.e. what is an input, and what is an output. It is argued that this is a more natural way to build models, particularly for engineering staff whom may not be experienced modellers.

Our second interest was to explore the similarity between our immediate problem of interest (simulating levels and flows with possible underflow/overflow) with hybrid systems. Systems where continuous dynamics interact with discrete events are termed hybrid. The study of hybrid systems is an active research area due to the increasing importance of digital control, embedded systems and the modelling of phenomena that does not lend itself readily to differential equation type descriptions.

This has led some to take the approach that if one is not interested in logic circuits or bang-bang control policies, and avoids through good operating practice events such as tanks emptying, then process simulators that are primarily designed to solve large, sparse ODE systems are sufficient.

However we believe that rather than being esoteric or of only specialised interest, hybrid systems are actually far more prevalent in process control applications than seems generally appreciated. Certainly other process modelling texts such as (Hangos and Cameron, 2001) recognise the both the challenge and ubiquity of hybrid systems.

The outline of the remainder of the paper is as follows: section 2 introduces a simple, but representative process simulation model. Important hybrid considerations are summarised in section 2.1. Simulation results for a variety of schemes using standard tools are given in section 3. Finally some general observations relevant to process simulation are given in sections 4 and 5.

2. LEVEL IN A SPHERICAL TANK

The modelling of liquid level in a spherical vessel as shown in Fig. 2 is a minor modification of a classical exercise in process modelling. The

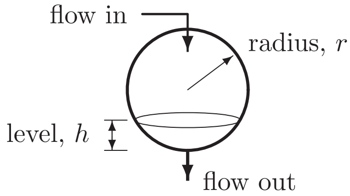


Fig. 2. Spherical tank/flow system

volume is given by a mass balance

$$\frac{dV}{dt} = F_{\text{in}} - F_{\text{out}}, \qquad V_0 = V(t = 0) \tag{1}$$

where V_0 is the initial liquid volume in the tank, F_{in} is the flow rate into the tank, and the outflow, F_{out}, follows Torricelli's theorem

$$F_{\text{out}} = \beta\sqrt{h} \tag{2}$$

where β is a constant. The volume of liquid at height h in a spherical tank is

$$V = \frac{\pi}{3}\, h^2(3r - h) \tag{3}$$

and is physically meaningful in the range $0 \leq h \leq 2r$. Eqns 1–3 define the process to be simulated.

There are two implementation issues with this index-1 DAE system. One is the presence of multiple roots due to the inversion of Eqn. 3 as shown in Fig. 3, and the other is how to reliably simulate events such as the tank emptying.

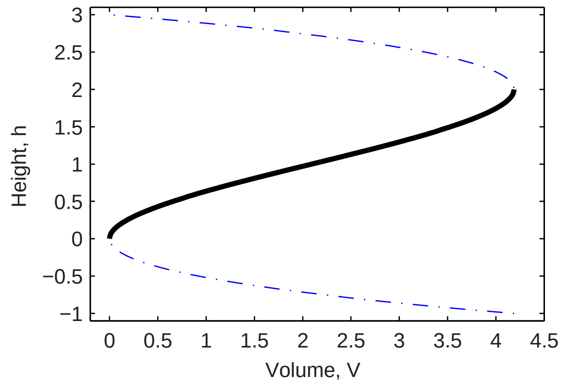


Fig. 3. The relationship between level, h, and volume, V for a spherical tank with radius $r = 1$, Eqn. 3.

2.1 *Hybrid modelling*

From a hybrid perspective, the process model is not simply Eqns 1–3 as stated above. The hybrid automaton given in Fig. 4 gives a more complete description by including the three discrete states, $Q \in \{\text{partially full, empty, overflowing}\}$, that characterize the system.

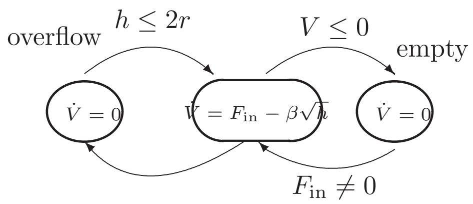


Fig. 4. Hybrid automaton for the tank level system

Under certain scenarios, the simulator rapidly switches between two discrete operating states, effectively stalling the simulation. This phenomena known as Zeno executions, (Zhang *et al.*, 2001), is unique to hybrid systems and is clearly of practical importance to simulators. In addition to accuracy and simulation time concerns, Zeno systems, or those sufficiently close to Zeno, can lead to a false sense of security when checking the hybrid system for criteria such as blocking states. This is because Zeno behaviour is not a true reflection of any physical system, rather it is an artifact of the modelling process.

There are two further challenging characteristics of the emptying tank simulation that are not shared by other common benchmark hybrid problems such as relays or bang/bang control problems. The first is that the continuous state (volume) becomes complex if the ODE system is applied outside the valid operating space. This poses problems for the numerical integrator routine that is attempting to accurately establish the emptying event location by using the standard zero crossing algorithms. The second problem is that the gradient of dV/dh at V or $h = 0$ is infinite causing further difficulties to the numerical root finder.

3. SIMULATION EXPERIENCES

For the purposes of illustration, we have chosen two general-purpose modelling environments. The first is MATLAB/Simulink which is a widely used directed block-diagram language clearly showing its classical control engineering heritage. MathModelica, (Jirstrand, 2000), is an acausal modelling product and, as the name suggests, is a hybrid of the computer algebra system Mathematica and the Modelica language compiler Dymola.

In the simulations that follow we model the emptying of a spherical tank with $r = 1, \beta = 0.09$ with no inflow, $F_{\text{in}} = 0$.

3.1 *Retaining the implicit expression for h*

From the modeller's perspective, the preferred option is to simply leave h as an implicit function of V and require that the simulation environment compute h when required. Both SIMULINK and MathModelica support DAE problems.

Fig. 5 illustrates the usage of the *Algebraic Constraint* option in SIMULINK to solve Eqn. 3. The advantage of such an approach is that the block diagram follows directly from the governing dynamic equations. The drawback is evident in the simulation results presented in Fig. 6. As the tank empties, h becomes negative, and SIMULINK quietly converts $\sqrt{h}$ to $\text{sign}(h)\sqrt{|h|}$ thereby retaining real, as opposed to complex, numbers. From this point on, *increasing* volume corresponds to *decreasing* level which is clearly nonsense.

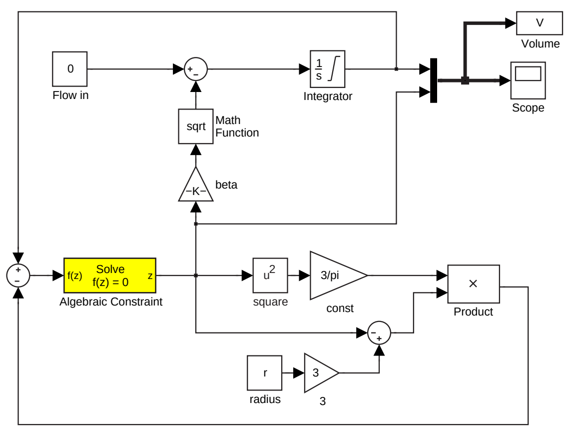


Fig. 5. Dynamic system in SIMULINK using the implicit Eqn. 3.

Initialising the simulation with the tank 100% full leads to a similar phenomenon. Here the solver again 'slips' into the wrong region and returns values of h higher than the top of the tank

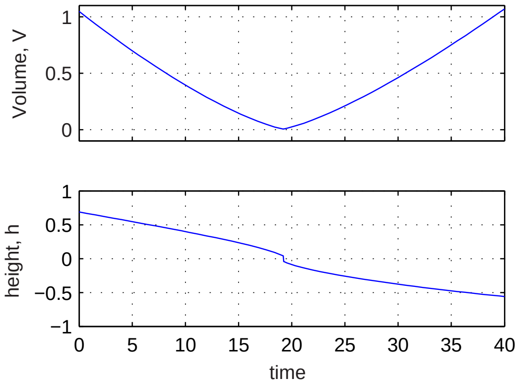


Fig. 6. Simulation result from Fig. 5 showing unrealisable levels.

indicating that *decreasing* volume corresponds to *increasing* level.

Both these phenomena are clearly artifacts due to how the model was implemented. The solver of the implicit algebraic equation is unaware of the physically valid limits of h. Adding a saturation block to constrain h to the interval $[0 \;\; 2r]$ avoids the case where the level becomes physically unrealisable, but at the expense of a spike at the emptying event, Fig. 7.

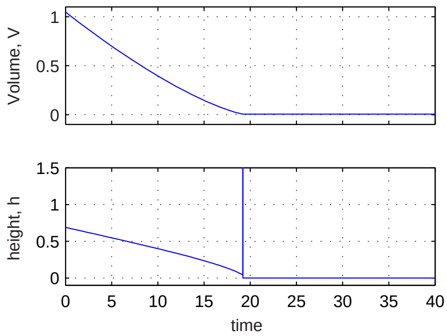


Fig. 7. Level and volume trends when saturating the level.

This crude method of trying to limit h is clearly suboptimal as it only cuts the algebraic equation solver's guess of h, $\hat{h}$, at the limits. It does not actually limit the solver's search region. Once the solver searches outside the limits, the input is effectively decoupled from the output, making any evaluation impossible.

3.2 MathModelica

One of MathModelica's design aims was to leverage Mathematica's symbolic capability within a simulation environment. In the MathModelica listing below, one simply lists the constituent equations leaving any necessary re-arrangement to the solver leading to a modelling approach many process engineers would be immediately comfortable with. To help ensure physically meaningful results, we specify the valid ranges for h and V.

```
model SphereTank "Spherical tank model"
// Constants
   constant Real pi=3.141592;
   constant Real g=9.81;

// Parameters
   parameter Real r=1.0 "Radius [m]";
   parameter Real beta = 0.02*sqrt(2*g);
   parameter Real Fin=0.0 "Inflow [m^3/s]";

// Variables
   Real V(min=0.0, max=4/3*pi*r^3,
        start=4/3*pi*r^3*0.5) "Volume of fluid [m^3]";
   Real h(min=0.0, max=2*r, start=2*r*0.5) "Level [m]";
   Real Fout "Outflow [m^3/s]";

equation
   V =  pi*r*h^2 - pi*h^3/3; // volume-level eqn
   Fout = beta*sqrt(abs(h)); // flow out
   der(V) = Fin - Fout;     // Conservation of mass
end SphereTank;
```

Fig. 8 indicates that MathModelica can easily manage the DAE, but as the level approached zero, the simulation stalled. Studying the diagnostics from the integrator routine revealed that it was still attempting to use negative numbers for h regardless of the specified valid range for h. Such Zeno-like behaviour is unsatisfactory in a process simulator.

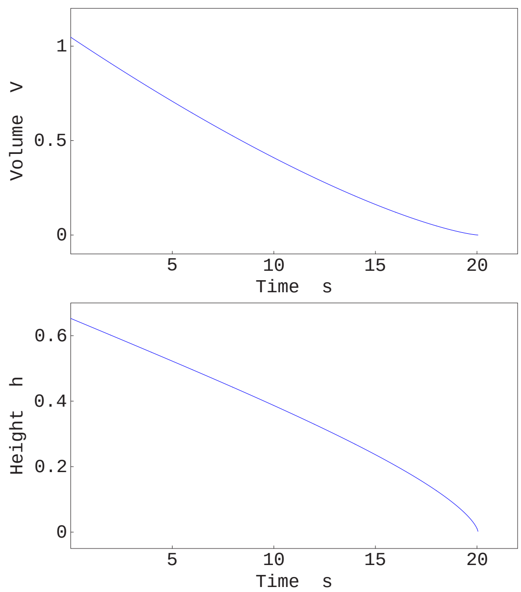


Fig. 8. MathModelica simulation stalling at the emptying event, $t \approx 20$s.

Subsequent communication with the developers confirmed that the Dymola kernel solver does not support the maximum/minimum attributes (as stated in the documentation), but rather only tests the valid range at $t = 0$.

3.3 Explicitly solving for h

The algebraic constraint, Eqn. 3, is a cubic equation which can be solved analytically, perhaps using a computer algebra tool, for h

$$h_1 = r + \frac{2^{1/3}\pi r^2}{M} + \frac{M}{2^{1/3}\pi} \tag{4}$$

$$h_2 = r - \frac{(1+i\sqrt{3})\pi r^2}{2^{2/3}M} - \frac{(1-i\sqrt{3})M}{\sqrt[3]{16\pi}} \tag{5}$$

$$h_3 = r - \frac{(1-i\sqrt{3})\pi r^2}{2^{2/3}M} - \frac{(1+i\sqrt{3})M}{\sqrt[3]{16\pi}} \tag{6}$$

where

$$M = \sqrt[3]{2\pi^3 r^3 - 3\pi^2 V + \pi^2\sqrt{3}\sqrt{3V^2 - 4\pi r^3 V}} \tag{7}$$

Not obvious from the above equations is that while all three roots for h are real in the region of interest $0 \leq V \leq 4\pi r^3/3$ as shown in Fig. 3, it is only h_3 that gives physically meaningful values. This can be proved from the observation that it is only dh_3/dV that is positive in the valid range. The SIMULINK implementation in Fig. 9 incorporates the explicit solution of Eqns 6 and 7.

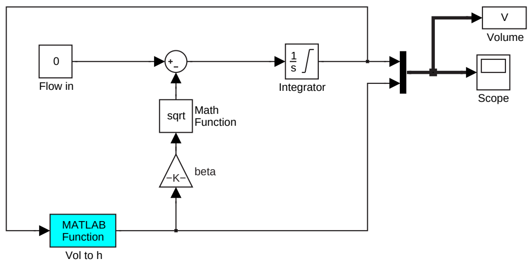


Fig. 9. Simulink model using the explicit expression for h given V.

Not surprisingly given the manual intervention required to construct the analytical inversion of Eqn. 3, the simulation performs as expected without the artifacts exhibited by the previous methods. However as the trends in Fig. 10 show, even this scheme is not completely robust. The 'hiccups' evident in the level, and to a lesser degree in the volume when the vessel is practically empty are almost two orders of magnitude above the requested numerical tolerance of 10^{-4}.

As Modelica has no intrinsic support for complex numbers, the implementation of equations 6 and 7 requires substantial re-working. While it is possible to manually define a complex number as a *struct* and develop the corresponding complex arithmetical libraries, this was considered too much work in general.

3.4 Casting level as the state

As this problem is an index-1 DAE, (refer (Ascher and Petzold, 1998, p232)), then a standard index reduction method is to differentiate the constraint equation, Eqn. 3, once and solving for dh/dt,

$$\frac{dh}{dt} = \frac{F_{\text{in}} - \beta\sqrt{h}}{\pi h(2r-h)} \tag{8}$$

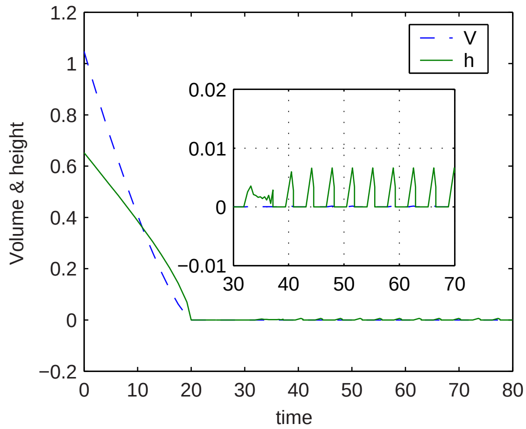


Fig. 10. Volume and level trends using the explicit expression from Fig. 9. The zoom portion in the insert shows evidence of chattering.

giving an index-0 problem. This is equivalent to considering level as the state variable as opposed to volume.

This eliminates the DAE problem, and by defining $F_{\text{out}} = \beta\,\text{sign}(h)\sqrt{|h|}$, as was done inadvertently in Fig. 6, then one can employ standard zero crossing detection techniques which is the commonly recommended solution approach to hybrid problems. One integrates up until the state-dependent event, stops and possibly adds or removes state dynamics, and then subsequently restarts until the next exception. The redefinition for F_{out} is even correct physically in the case where the height reflects the pressure differential between the bottom of the tank and atmospheric, and a reverse flow is possible. However changing the corresponding level dynamics from Eqn. 8 to $(F_{\text{in}} - \beta\sqrt{h})/(\pi|h|(2r-h))$ introduces a singularity at the event which cannot be integrated past using an adaptive scheme such as Runga-Kutta-Fehlberg.

4. DISCUSSION

The modelling of vessels emptying should be a trivial simulation exercise. However combining non-standard vessel shapes with the possibility of running dry, causes admittedly small, but still annoying, simulation hiccups. The emptying of a spherical tank combines implicit constraints with important exceptions. These problems are not just limited to tanks running dry, similar process scenarios are models involving implicit friction factor expressions operating across different flow regimes, reaction kinetics coupled with thermodynamics, and mechanical systems with friction.

While the emptying of a tank is clearly a toy problem, a full paper machine such as reported in (Balderud *et al.*, 2001) is not. In this application, adaptive error control was needed to complete the simulations in a reasonable time, but it should not stumble or exhibit Zeno-like behaviour when reaching underflows and overflows.

If approximate trends suffice, then crude integrators without error control typically suffice, oblivious of the numerical subtleties of the problem. To obtain clean results takes considerably more effort, effort that a modern process simulation environment could in principle do.

It is possible with manual intervention, employing analytical solutions where possible, judicious use of 'G-stops' and resetting the integrator, we can circumvent some of the simulation hiccups. However when using specific analytical solutions, one must take care that small changes to the problem formulation (such as plant rebuilds) do not necessitate major changes to the solution methodology.

Our SIMULINK simulations typically delivered a level trend, but not always the trend expected. When faced with complex levels,the automatic corrective action taken is justifiable. Adding saturation gives reasonable trends before and after the event, but spikes at the event causing severe problems to any derivative based level controller.

MathModelica, with the potential to combine symbolic manipulation and numerical calculation failed to simulate past the emptying event, and it lacked support for complex numbers making the alternative approach (solving the constraint equations off-line) unwieldy.

5. CONCLUSIONS

The realisation that the hybrid system is important in process simulation and not just appropriate in discrete logic circuits or discrete manufacturing is, we believe, important. The hybrid nature stems from both events that have often traditionally been sidestepped, if not outright ignored, in process simulation such as tanks emptying, in combination with the numerical integrator within the process simulator environment.

This paper investigates the potential of acausal modelling tools for process engineering applications that use symbolic manipulation as a means to ease the modeller's development task. Acausal tools have not yet quite reached the point where the modeller can simply specify the various dynamic and constraint equations, and confidently leave the resultant solution to the executive. Issues such as singularities and Zeno executions are still common in any non-trivial process simulation. Notwithstanding, we believe that process simulators that make use of symbolic manipulators are the way to achieve this goal.

REFERENCES

Ascher, Uri M. and Linda R. Petzold (1998). *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations.* Society for Industrial and Applied Mathematics. Philadelphia, USA.

Åström, K.J., H. Elmqvist and S.E. Mattsson (1998). Evolution of Continuous-Time Modeling and Simulation. In: *The 12th European Simulation Multiconference, ESM'98.* Manchester, UK.

Balderud, Jonas and David Wilson (2003). The dynamic behaviour of the wet-end of a board machine. In: *CHEMECA 2003: 31st Annual Australasian Chemical Engineering Conference* (Dzuy Nguyen, Ed.). Adelaide, Australia.

Balderud, Jonas, Christian Haag and David Wilson (2001). Large-scale dynamic paper machine models. In: *6th World Congress of Chemical Engineering.* Melbourne, Australia. ISBN 0 7340 2201 8.

Haag, Christian and David Wilson (2001). Experiences of large-scale board machine modelling. In: *Proceedings of the IASTED Modelling, Identification and Control* (M.H. Hamza, Ed.). Innsbruck, Austria. pp. 745–750.

Haag, Christian and David Wilson (2003). Potential for symbolic-assisted process simulation. In: *IASTED conference on Modelling and Simulation 2003* (M. H. Hamza, Ed.). IASTED/ACTA Press. Palm Springs, Ca, USA. pp. 572–577.

Hangos, Katalin and Ian Cameron (2001). *Process Modelling and Model Analysis.* Academic Press.

Jirstrand, Mats (2000). Mathmodelica — a full system simulation tool. In: *Product Models 2000 - The Sixth Conference on Product Models, Global Product Development.* Linköping, Sweden.

Marquardt, W. (1996). Trends in Computer-Aided Process modelling. *Computers in Chemical Engineering* **20**(6/7), 591–609.

McGarry, M., B. Bickell and G. Pelkey (1997). A case study of the use of actual controls in simulation trainers. In: *83rd Annual meeting technical section — paper, people, progress.* Vol. 78. Canadian Pulp and Paper Association. Montreal, Que, Canada. pp. B357–B360.

Wilson, David I. and Jonas Balderud (2000). Experiences building and validating a large-scale dynamic paper machine model. In: *Model Validation for Plant Control and Condition Monitoring* (D.H. Owens, Ed.). IEE, Savoy Place London, WC2R 0BL, UK. pp. 2/1–2/7.

Zhang, Jun, Karl Henrik Johansson, John Lygeros and Shankar Sastry (2001). Zeno hybrid systems. *International Journal of Robust and Nonlinear Control* **11**, 435–451.