

# Approach for the Mode Switching Problem in Piecewise Smooth Implicit Multilinear IVPs

Torben Warnecke<sup>1</sup> <sup>a</sup> and Gerwald Lichtenberg<sup>2</sup> <sup>b</sup>

<sup>1</sup>Deutsches Elektronen-Synchrotron DESY, Notkestraße 85, 22607 Hamburg, Germany

<sup>2</sup>Faculty of Life Sciences, Hamburg University of Applied Sciences, Ulmenliet 20, 21033 Hamburg, Germany

**Keywords:** Hybrid Dynamical Systems, Multilinear Algebra, Differential-Algebraic Models, Descriptor Systems, Switched Systems, Piecewise-Smooth Dynamical Systems.

**Abstract:** This paper addresses the mode switching problem in piecewise smooth implicit multilinear initial value problems (IVPs), which are relevant for modeling hybrid dynamical systems like HVAC and power systems. Unlike traditional switched systems with explicit mode descriptions, this work focuses on systems where mode information is implicitly encoded in binary-valued variables and switching conditions are defined by inequality constraints. The paper investigates the transversal motion discontinuities that occur when the system meets the boundary surfaces defined by these constraints. A method is presented to determine the discontinuous motion by analyzing the total derivative of the inequality constraints. The modeling framework utilizes hybrid implicit multilinear time-invariant (iMTI) functions and describes the system using inequality-constrained index-1 differential-algebraic equations (DAEs). The Jacobian matrices and thus the total derivatives can be estimated algebraically, due to the use of multilinear functions. To handle the combinatorial complexity associated with the binary variables during mode switching, the paper proposes using sparsity pattern analysis to identify and solve sub-problems more efficiently. The presented method is applied to a two-point temperature-controlled three-tank system, and simulations are performed using the MTI-Toolbox for MathWorks MATLAB.

## 1 INTRODUCTION

In recent researches implicit multilinear time invariant (iMTI) models are of special interest since they can describe fundamental dynamic relations in HVAC and power systems, (Samaniego et al., 2024), (Engels et al., 2024), (Kaufmann et al., 2023), (Luxa et al., 2022), (Pangalos et al., 2014). The multilinear model class is a hyperclass of linear and binary models, while being a subclass of polynomial and general nonlinear models, illustrated in figure 1 (Lichtenberg et al., 2022).

The mode switching problem in piecewise smooth implicit multilinear initial value problems (IVPs) is important to simulate hybrid iMTI models reliably, which gained attention due to the ability to model saturation, switches and logical functions (Warnecke and Lichtenberg, 2024a), (Lichtenberg et al., 2022). The mode switching can be understood as the transversal motion between the boundary surfaces of hybrid/switched iMTI models.

brid/switched iMTI models.

Usual switched systems have an explicit description of modes, switching conditions and transition functions. They introduce multiple sets of equations and constraints (Benzaouia, 2012) alongside multiple sets of switching conditions either by equations, inequalities (Calvo et al., 2016) or subsets  $\mathbb{S} \subset \mathbb{R}^n$  of the signal space  $\mathbb{R}^n$  (Dieci and Lopez, 2011), as well as transitions functions in between modes (Mehrmann and Wunderlich, 2009).

Here a type of continuous-time hybrid iMTI models is of interest, described by implicit multilinear DAEs, which only implicitly encode their mode information in binary-valued variables, while switching conditions and transition functions are encoded in additional inequality constraints. They include continuous states  $\mathbf{x} \in \mathbb{R}^n$ , input signals  $\mathbf{u} \in \mathbb{R}^m$ , as well as continuous- and binary-valued algebraic variables  $\mathbf{y} \in \mathbb{R}^p$  and  $\mathbf{z} \in \mathbb{B}^q$ , with  $\mathbb{B} = \{\text{true}, \text{false}\}$ .

The hybrid iMTI model framework shares structural similarities to the hybrid DAE model formulation described in Appendix B of the Modelica Language Specification (Fritzson and Engelson, 1998).

<sup>a</sup>  <https://orcid.org/0009-0004-3037-8634>

<sup>b</sup>  <https://orcid.org/0000-0001-6032-0733>

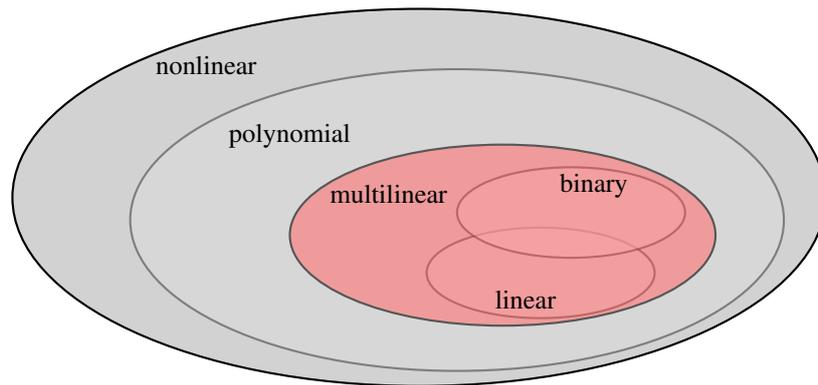


Figure 1: Illustration of system classes.

Both approaches represent fully implicit formulations of hybrid systems, lacking an explicit partitioning of system dynamics for individual modes. However, they differ in two key aspects: the iMTI framework imposes a restriction to multilinear functions, and it employs implicit inequality constraints in place of the discrete equations used in Modelica.

Hybrid iMTI models offer two principal advantages: First, they can reduce storage requirements compared to general nonlinear switched systems. This is due to the fact that the parameters and the structure of both the equations and the inequality constraints can be compactly represented through the structure and parameter matrices  $\mathbf{S}$  and  $\phi$ , respectively—eliminating the need for additional semantic information, similar to the use of system matrices  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{C}$ , and  $\mathbf{D}$  in linear state-space models. Second, unlike general nonlinear DAE formulations (such as those described in the Modelica Language Specification), the constrained class of iMTI models allows for algebraic derivation of Jacobian matrices with the direct use of the structure and parameter matrices. This, in turn, enables direct computation of total derivatives of the model functions, as detailed in Chapter 2.1.

The publications solely focuses on the transversal motion discontinuities of such IVPs and is limited to index-1 DAEs without structural changes, e.g. a change in the number of variables or variable-index systems. Also sliding motions and their Filippov solutions are not considered so far.

In (Calvo et al., 2016) an algorithm to determine the discontinuous motion and solving the IVP for explicit piecewise smooth (switched) ODEs based of the total derivative of the mode constraints is described. Here, this idea is used to determine the implicitly described discontinuity and solving the transversal motion problem of the piecewise smooth implicit multilinear IVPs. The method is described in chapter 3.

A block lower triangular (BLT) partitioning of the incidence matrix is used to reduce complexity of the

mode switching problem, similar as in other tools, e.g. Modelica (Sanz et al., 2014). The BLT partitioning approach is described in chapter 4.

## 2 MODELLING FRAMEWORK

### 2.1 Hybrid Implicit Multilinear Models

The signal space

$$\mathbf{v}(t) = \begin{bmatrix} \dot{\mathbf{x}}(t) \\ \mathbf{x}(t) \\ \mathbf{u}(t) \\ \mathbf{y}(t) \\ \mathbf{z}(t) \end{bmatrix} \in \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^p \times \mathbb{B}^q \quad (1)$$

of a hybrid implicit multilinear state-space model consists of a continuous-valued state vector  $\mathbf{x}(t) \in \mathbb{R}^n$ , a continuous-valued input vector  $\mathbf{u}(t) \in \mathbb{R}^m$ , a continuous-valued algebraic signal vector  $\mathbf{y}(t) \in \mathbb{R}^p$  and a binary-valued algebraic signal vector  $\mathbf{z}(t) \in \mathbb{B}^q$  at continuous time  $t \in \mathbb{R}$ .

**Definition 2.1** (multilinear function). A *multilinear function* is a function  $f_c : \mathbb{R}^n \rightarrow \mathbb{R}$  which is linear in each individual variable separately, i.e. if all except one arbitrary variable is held constant, it is a linear function of this variable (Lang, 2002).

**Definition 2.2** (hybrid multilinear function). A *hybrid multilinear function* is a function  $f : \mathbb{H}^n \rightarrow \mathbb{R}$  from a hybrid space to a real number which gives the same result as the corresponding multilinear function  $f_c : \mathbb{R}^n \rightarrow \mathbb{R}$  if the binary variables are converted with the standard injector

$$\tilde{v}_l = \begin{cases} 1 \in \mathbb{R} & \text{if } v_l = true \in \mathbb{B} \\ 0 \in \mathbb{R} & \text{if } v_l = false \in \mathbb{B} \\ v_l & \text{if } v_l \in \mathbb{R}. \end{cases} \quad (2)$$

**Definition 2.3** (hybrid multilinear vector function). A hybrid multilinear vector function is a function  $\mathbf{f}: \mathbb{H}^n \rightarrow \mathbb{R}^q$  which maps a hybrid space to a real vector space where each element is given by a hybrid multilinear function.

A continuous time hybrid implicit multilinear time-invariant (hybrid iMTI) model is characterized by a set of equations and inequality constraints that represent an implicit system of inequality constrained differential algebraic equations (DAEs)

$$\mathbf{f}(\mathbf{v}(t)) = \mathbf{0}, \quad (3)$$

$$\mathbf{g}(\mathbf{v}(t)) \leq \mathbf{0}, \quad (4)$$

where  $\mathbf{f}: \mathbb{R}_x^n \times \mathbb{R}_y^n \times \mathbb{R}_u^m \times \mathbb{R}_z^p \times \mathbb{B}_z^q \rightarrow \mathbb{R}^{n+p}$  and

$\mathbf{g}: \mathbb{R}_x^n \times \mathbb{R}_y^n \times \mathbb{R}_u^m \times \mathbb{R}_z^p \times \mathbb{B}_z^q \rightarrow \mathbb{R}^q$  are hybrid multilinear vector functions.

**Definition 2.4** (Discontinuity surface of a hybrid iMTI system). For the continuous-time hybrid dynamical system described by the equations (3) and inequality constraints (4), the zeros of the element of  $\mathbf{g}(\mathbf{v}(t))$  span the discontinuity surfaces  $S$ . When a discontinuity surface  $S_i$  is met at the event time  $t_s$ , meaning the function value of  $g_i(\mathbf{v}(t = t_s)) = 0$  is equal to zero, the system can exhibit a discontinuous motion and change its operation mode.

**Definition 2.5** (Operation mode of a hybrid iMTI system). Different from classical switched system approaches the operation mode of the system is not explicitly stated nor governed by individual sets of equations and constraints. Instead, the operation mode is implicitly described by the combination of the binary-algebraic variables  $\mathbf{z}$ . The Domain of each mode is constrained by the inequality constraints (4). Moreover, not all combinations of  $\mathbf{z}$  must be accessible and therefore the number  $n_{mode}$  of possible operation modes is less or equal to the number of possible combinations of  $\mathbf{z} \in \mathbb{B}^q$ :

$$n_{mode} \leq 2^q. \quad (5)$$

The multilinear functions of a hybrid iMTI model can be written in a normalized canonical polyadic decomposition (Jöres et al., 2022):

$$f_j(\mathbf{v}(t)) = \sum_{k=1}^R \Phi_{F_{jk}} \prod_{i=1}^{2n+m+p+q} (1 - |S_{ik}| + S_{ik}v_i(t)), \quad (6)$$

$$g_j(\mathbf{v}(t)) = \sum_{k=1}^R \Phi_{G_{jk}} \prod_{i=1}^{2n+m+p+q} (1 - |S_{ik}| + S_{ik}v_i(t)), \quad (7)$$

where all model information is stored in the structure matrix  $\mathbf{S} \in \mathbb{R}^{(2n+m+p+q) \times R}$  and parameter matrices  $\Phi_f \in \mathbb{R}^{(n+p) \times R}$  and  $\Phi_g \in \mathbb{R}^{q \times R}$ .

**Definition 2.6** (Jacobian matrix of a hybrid multilinear vector function). The Jacobian matrix  $\mathbf{Jf}(\mathbf{v}(t))$  is a matrix arrangement of the first derivatives of a vector function  $\mathbf{f}(\mathbf{v}(t)) \in \mathbb{R}^{(n+p)}$  with the variables  $\mathbf{v}(t) \in \mathbb{H}^n$ , where an element  $Jf_{ab}(\mathbf{v}(t))$  represents the first derivative  $\frac{\partial f_a(\mathbf{v}(t))}{\partial v_b}$  of the element  $f_a$  with respect to the variable  $v_b$ . The Jacobian matrix  $\mathbf{Jf} \in \mathbb{R}^{(n+p) \times (2n+m+p+q)}$  of the hybrid multilinear vector function  $\mathbf{f}$  can be directly computed using the normalized CP decomposed representation, as shown in (Kaufmann et al., 2023). Each element

$$Jf_{ab}(\mathbf{v}(t)) = \sum_{k=1}^R \Phi_{ak} S_{bk} \prod_{i \in N} (1 - |S_{ik}| + S_{ik}v_i(t)) \quad (8)$$

can be computed at time  $t$  with the signal vector  $\mathbf{v}(t)$ , the structure matrix  $\mathbf{S}$  and the parameter matrix  $\Phi_f \in \mathbb{R}^{(n+p) \times R}$ , where  $N = \{1 \dots (2n+m+p+q)\} \setminus b$  is the set of indices of the variables excluding the index  $b$  of the variable  $v_b(t)$ . Similar for  $\mathbf{g}(\mathbf{v}(t)) \in \mathbb{R}^q$ .

**Definition 2.7** (Total derivative of a hybrid multilinear vector function). The total derivative

$$\mathbf{Df}(\mathbf{v}(t)) = \mathbf{Jf}(\mathbf{v}(t)) \frac{d\tilde{\mathbf{v}}(t)}{dt} \in \mathbb{R}^{n+p} \quad (9)$$

of a hybrid multilinear vector function  $\mathbf{f}(\mathbf{v}(t)) \in \mathbb{R}^{(n+p)}$  with  $\mathbf{v}(t) \in \mathbb{H}^{(2n+m+p+q)}$  at time  $t$  can be computed from its Jacobian matrix  $\mathbf{Jf} \in \mathbb{R}^{(n+p) \times (2n+m+p+q)}$  at time  $t$  and the time-derivative  $\frac{d\tilde{\mathbf{v}}(t)}{dt} \in \mathbb{H}^{(2n+m+p+q)}$  of the signal vector  $\mathbf{v}(t)$  (using the standard injector). The total derivative describes the evolution of the function values over time.

**Remark 2.1.** The time derivative of the binary variables is assumed to be zero, since a switch in its value can be understood as switching the systems equations. So they are treated as constant parameters of the different functions. Or differently one could state, that the time derivatives of  $\mathbf{z}$  directly before and after the switch is zero.

## 2.2 Example: 2-Point Temperature-Controlled Three Tank System

The system is illustrated in figure 2. It consists of 3 water tanks sharing one inlet flow of temperature  $u_2$ . The water tanks transfer heat between each other, as well as losing heat to the environment with the ambient temperature  $u_1$ . If multiple valves ( $z_1, z_3$  and  $z_5$ ) are opened at the same time the inlet flow will be shared uniformly between those. The three 2-Point

controllers are designed to maintain the temperatures of the tanks ( $x_1, x_2$  and  $x_3$ ) between lower ( $u_3, u_5$  and  $u_7$ ) and upper bounds ( $u_4, u_6$  and  $u_8$ ).

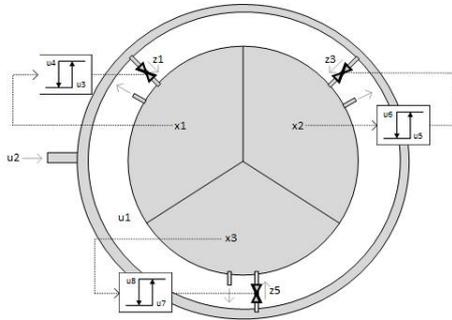


Figure 2: Schematic of the hybrid temperature-controlled three tank system.

The system can be modelled by the following set of implicit inequality-constrained multilinear DAEs:

$$(cm\dot{x}_1 + \gamma x_1 - \alpha_1(x_2 + x_3) - \alpha_2 u_1) y_1 + c f_v(x_1 - u_2) z_1 = 0, \tag{10}$$

$$(cm\dot{x}_2 + \gamma x_2 - \alpha_1(x_1 + x_3) - \alpha_2 u_1) y_1 + c f_v(x_2 - u_2) z_3 = 0, \tag{11}$$

$$(cm\dot{x}_3 + \gamma x_3 - \alpha_1(x_1 + x_2) - \alpha_2 u_1) y_1 + c f_v(x_3 - u_2) z_5 = 0, \tag{12}$$

$$y_1 - z_1 z_3 - z_1 z_5 - z_3 z_5 + z_1 z_3 z_5 - 1 = 0, \tag{13}$$

$$(u_3 - x_1)(1 - z_1 + z_2) \leq 0, \tag{14}$$

$$(u_5 - x_2)(1 - z_3 + z_4) \leq 0, \tag{15}$$

$$(u_7 - x_3)(1 - z_5 + z_6) \leq 0, \tag{16}$$

$$(x_1 - u_4)(1 - z_2 + z_1) \leq 0, \tag{17}$$

$$(x_2 - u_6)(1 - z_4 + z_3) \leq 0, \tag{18}$$

$$(x_3 - u_8)(1 - z_6 + z_5) \leq 0, \tag{19}$$

where  $\gamma = (\alpha_1 + 2\alpha_2)$ , and the parameters are the heat capacity  $c = 4200kJ/kgK$  and mass  $m = 5kg$  of water in each tank, the ambient heat transfer  $\alpha_1 = k_0 A_0 = 125W/K$  and heat transfer  $\alpha_2 = k_{in} A_{in} = 125W/K$  in between the tanks, and valve coefficient  $f_v = 0.09kg/s$ . The model is feasible for  $[u_4(t) \ u_6(t) \ u_8(t)]^T \leq \mathbf{x}(t)$  and  $\mathbf{x}(t) \leq [u_4(t) \ u_6(t) \ u_8(t)]^T$ .

### 3 METHOD

When one or more discontinuity surfaces  $S$  are met, meaning the function values of one or more inequality constraints  $g_s(\mathbf{v}(t = t_s)) = 0$  are equal to zero, where  $t_s$  is the event time and  $g_s$  are the according elements of the vector function  $\mathbf{g}$ , one needs to determine how the system will behave.

Usually a hybrid implicit multilinear system is capable of different motions:

- staying in the previous mode, so the elements of  $g_s$  will continuously go back to negative values and the binary algebraic variables won't change
- sliding along a boundary, where some elements of  $g_s$  will stay at zero (Filippov solutions, (Piiroinen and Kuznetsov, 2008))
- change its mode (transversal motion), where the binary variables will change, so that either the function values  $g_s$  discontinuously switch to negative values or continuously go back to negative values after the switch
- a combination of transition and sliding.

When a boundary is met, which can be numerically detected by event detection algorithms implemented in ODE solvers, one needs to solve the inequality constrained system of DAEs at the event time  $t_s$ , with testing possible combinations of  $\mathbf{z}$  until a solution for the discontinuous motion problem of the dynamical system is found.

#### Solution of the Inequality Constrained System of Equations at the Event Time $t_s$ .

The unknown state derivatives  $\dot{\mathbf{x}}(t = t_s)$  and continuous-valued algebraic variables  $\mathbf{y}(t = t_s)$  can be computed by solving the inequality constrained system of equations (3) at the event time  $t_s$  for different combinations of  $\mathbf{z}$ , where the solution must satisfy (4). A solution can be efficiently obtained by the method described in (Warnecke and Lichtenberg, 2024b) where the problem is decomposed in sub-problems and explicit solutions.

It is important to mention, that the solution to the inequality constrained system of equations at the event time  $t_s$  must not be a solution for the dynamical problem of the inequality constraint system of DAEs. Usually at the time  $t_s$  there is more than one solution to the inequality constrained system of equations, since the current system state is already one of them.

#### Solution of the Discontinuous Motion Problem.

Similar to the solutions of the inequality constrained system of equations, the solutions for the dynamical problem of the inequality constrained system of DAEs must satisfy (3) and (4). If all elements of the vector function

$$\mathbf{g}(\mathbf{v}(t = t_s)) < 0 \tag{20}$$

are negative the solution is also a solution for the dynamical system, and it will undergo a transversal motion, where the according variables  $\mathbf{z}$  represent the future mode of the system. If some values of  $\mathbf{g}(\mathbf{v}(t = t_s))$  are equal to zero:

$$g_l(\mathbf{v}(t = t_s)) = 0 \tag{21}$$

with  $l$  as the indices for the according elements of  $\mathbf{g}(\mathbf{v}(t = t_s))$ , the according elements  $Dg_l(\mathbf{v}(t = t_s))$  of the total derivative  $Dg(\mathbf{v}(t = t_s))$  of  $\mathbf{g}(\mathbf{v}(t = t_s))$  must be equal to or smaller than zero:

$$Dg_l(\mathbf{v}(t = t_s)) \leq 0, \quad (22)$$

so that the solution is also a solution for the dynamical system. If  $Dg_l(\mathbf{v}(t = t_s)) = 0$  the system will slide along the boundary. If the combination of the future and current binary variables  $\mathbf{z}$  are not similar the system undergoes a transversal motion.

**Remark 3.1.** *If for all possible combinations of  $\mathbf{z}$  the function values  $\mathbf{g}(\mathbf{v}(t = t_s)) \leq 0$  are smaller or equal to zero and the total derivatives  $Dg_l(\mathbf{v}(t = t_s)) > 0$  of boundaries  $g_l(\mathbf{v}(t = t_s)) = 0$  is bigger than zero, the behavior can be described as zeno behavior; since even when the mode changes, the system will immediately be pushed towards the boundary.*

To compute  $Dg(\mathbf{v}(t = t_s))$  the missing time-derivatives  $\frac{d\dot{\mathbf{x}}}{dt}$  and  $\frac{d\dot{\mathbf{y}}}{dt}$  of the signal vector  $\mathbf{v}(t)$  can be computed with the total derivative  $\mathbf{Df}(\mathbf{v}(t))$  of the vector function  $\mathbf{f}(\mathbf{v}(t))$  of the equations (3), with the assumption that the time-derivative of the binary-valued algebraic signals is equal to zero right after the switch. Since the value of the vector function is constant and equal to zero at every time  $t$ , the total derivative

$$\mathbf{Df}(\mathbf{v}(t)) = \mathbf{Jf}(\mathbf{v}(t)) \frac{d}{dt} \begin{bmatrix} \dot{\mathbf{x}}(t) \\ \dot{\mathbf{x}}(t) \\ \mathbf{u}(t) \\ \mathbf{y}(t) \\ \mathbf{z}(t) \end{bmatrix} = \mathbf{Jf}(\mathbf{v}(t)) \begin{bmatrix} \ddot{\mathbf{x}}(t) \\ \dot{\mathbf{x}}(t) \\ \dot{\mathbf{u}}(t) \\ \dot{\mathbf{y}}(t) \\ \mathbf{0} \end{bmatrix} = \mathbf{0} \quad (23)$$

is also equal to zero at every time  $t$  as well.

(23) can be rearranged for calculating  $\ddot{\mathbf{x}}(t)$  and  $\dot{\mathbf{y}}(t)$  as the solution of the linear system of equations

$$\mathbf{Jf}_{\dot{\mathbf{x}},\dot{\mathbf{y}}}(\mathbf{v}(t)) \begin{bmatrix} \ddot{\mathbf{x}}(t) \\ \dot{\mathbf{y}}(t) \end{bmatrix} = -\mathbf{Jf}_{\mathbf{x},\mathbf{u}}(\mathbf{v}(t)) \begin{bmatrix} \dot{\mathbf{x}}(t) \\ \dot{\mathbf{u}}(t) \end{bmatrix}, \quad (24)$$

if  $\mathbf{Jf}_{\dot{\mathbf{x}},\dot{\mathbf{y}}}(\mathbf{v}(t))$  is invertible, where  $\mathbf{Jf}_{\mathbf{x},\mathbf{u}}(\mathbf{v}(t))$  represents the columns of the Jacobian matrix according to the states  $\mathbf{x}(t)$  and inputs  $\mathbf{u}(t)$  and  $\mathbf{Jf}_{\dot{\mathbf{x}},\dot{\mathbf{y}}}(\mathbf{v}(t))$  represents the columns of the Jacobian matrix according to the state derivatives  $\dot{\mathbf{x}}(t)$  and continuous-valued algebraic signals  $\mathbf{y}(t)$  at time  $t$ . The derivatives  $\dot{\mathbf{u}}(t)$  can be computed from the given input signal  $\mathbf{u}(t)$ , e.g. when using linear interpolation for a given set of input points as the slopes of the linear piece-wise function.

With  $\ddot{\mathbf{x}}(t)$ ,  $\dot{\mathbf{x}}(t)$ ,  $\dot{\mathbf{y}}(t)$  and  $\dot{\mathbf{u}}(t)$  the total derivative

$$\mathbf{Dg}(\mathbf{v}(t)) = \mathbf{Jg}(\mathbf{v}(t)) \begin{bmatrix} \ddot{\mathbf{x}}(t) \\ \dot{\mathbf{x}}(t) \\ \dot{\mathbf{u}}(t) \\ \dot{\mathbf{y}}(t) \\ \mathbf{0} \end{bmatrix} \quad (25)$$

of the inequality constraints  $\mathbf{g}(\mathbf{v}(t))$  can be computed.

**Remark 3.2.** *If one is only interested in one solution, and it can be any of the possible solution, it would be easy to iterate through all possible combinations until an appropriate solution is found.*

**Remark 3.3.** *Numerical precision can be crucial for solving the discontinuous motion problem, because the possible solutions differ drastically around the function values of zero and slight inaccuracy can change the result, especially regarding the inequality constraints and binary variables. To solve such numerical issues normalization techniques for the function values could be used, e.g. by normalizing both, the signal and parameter space. Also, one could think about unilateral tolerances of the numerical procedure.*

Still the problem remains of high combinatorial complexity with  $2^q$  possible combinations of  $\mathbf{z}$ . For most systems it would be reasonable to first try the solutions with the smallest hamming distance. Also, in most IVPs the problem can be reduced by sparsity pattern analysis, as described in the following chapter.

## 4 PROBLEM REDUCTION BY SPARSITY PATTERN ANALYSIS

The sparsity pattern of a system of equations and constraints can be represented by the incidence matrix  $\mathbf{P}$  of the IVP. By exploiting sparsity, e.g. by reordering of the sparsity pattern, one can reduce the complexity of the discontinuous motion problem. A sparsity pattern  $\mathbf{P}$  of a (hybrid) iMTI model can be calculated as described in (Warnecke and Lichtenberg, 2024b). By ordering the sparsity pattern of the unknown variables in a block lower triangular matrix  $\hat{\mathbf{P}}$  distinct sub-sets and sub-problems of the system of inequality constrained DAEs can be identified (Baharev et al., 2019). Different heuristics can be used for the NP-complete problem (Vassilevska and Pinar, 2004), e.g. by using algorithms like the Dulmage–Mendelsohn decomposition (Dulmage and Mendelsohn, 1958), as in (Pothen and Fan, 1990), or simple heuristics as in (Pinar and Heath, 1999).

**Definition 4.1** (sub-set). *Sub-sets are sets of equations and constraints of a system of implicit equations and constraints, which do not have any similar unknown variables (unknown regarding the IVP) and therefore can be solved independent of each other (Warnecke and Lichtenberg, 2024b).*

The index set  $A$  denotes the set of indexes according to the equations and constraints of the sub-set

(rows of the sparsity pattern  $\hat{\mathbf{P}}$ ). The index set  $B$  denotes the set of indexes according to the variables of the sub-set (columns of the sparsity pattern  $\hat{\mathbf{P}}$ ). In a block lower triangular sorted sparsity pattern  $\hat{\mathbf{P}}$ , sub-sets are diagonal blocks which do not have any overlapping blocks to other diagonal blocks. The elements  $\hat{p}_{ik}$  of the rows and columns adjacent to such a block must only consist of zeros:

$$\hat{p}_{ik} = 0, \tag{26}$$

for  $i \notin A$  and  $k \in B$ , as well as  $i \in A$  and  $k \notin B$ .

**Definition 4.2** (sub-problem). *A sub-problem is a set of equations and constraints, which is part of a sub-set of a system of implicit equations and constraints (Warnecke and Lichtenberg, 2024b).*

The index-set  $C$  denotes the set of indexes according to the equations and constraints of the sub-problem (corresponding to rows in the sparsity pattern  $\hat{\mathbf{P}}$ ). The index-set  $D$  denotes the set of indexes according to the variables of the sub-problem (corresponding to columns of the sparsity pattern  $\hat{\mathbf{P}}$ ). Unlike sub-sets, different sub-problems can have similar (unknown) variables.

In a block lower triangular sparsity pattern  $\hat{P}$ , sub-problems are diagonal blocks which can have (lower and left) overlapping blocks to other diagonal blocks, but the elements  $\hat{p}_{ik}$  of the right and upper parts of the rows and columns adjacent to the sub-problem must only consist of zeros:

$$\hat{p}_{ik} = 0, \tag{27}$$

for  $i < c$ , where  $c \in C$  and  $\forall_x \in C c \leq x$ , and  $k \in D$ , as well as  $i \in C$  and  $k > D$ , where  $d \in D$  and  $\forall_x \in D d \geq x$ .

A sub-problem determines the unknown variables according to its columns of the sparsity pattern, by solving the set of implicit equations and constraints according to its rows, but can be dependent on previously solved variables of other sub-problems.

For the example model the sorted sparsity pattern

$$\hat{P} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \end{pmatrix}, \tag{28}$$

regarding the unknown variables of the IVP ( $\dot{\mathbf{x}}$ ,  $\mathbf{y}$  and  $\mathbf{z}$ ) reveals one overall sub-set with 7 sub-problems. The sub-problems can be computed in series, where

instead of  $2^6 = 64$  overall combinations of binary variables, the number of possible combinations that need to be tested reduces to  $3 \times 2^2 = 12$ , since they are independent of each other.

With the use of the sub-problems and sub-sets, the combinatorial complexity of the discontinuous motion problem can be reduced. In figure 3 the reduced brute force method is shown.

$\mathbf{z}^i$  denote a combination of the binary variables  $\mathbf{z}$ , where  $\mathbf{z}^{i+1}$  denote the next possible combination (e.g. with the smallest hamming distance to  $\mathbf{z}^i$ ). The values of the binary variables  $\mathbf{z}_k$  which belong to a previous sub-problem in each sub-set of the events  $\mathbf{g}_s$  are fixed, since they cannot be influenced by the events, thus reducing the possible combinations of  $\mathbf{z}$ .

## 5 SIMULATION OF THE EXAMPLE

The three tank system is modelled and simulated using the MTI-Toolbox (Lichtenberg et al., 2024) for MATLAB, where the method is incorporated. The smooth sections of the IVP are calculated using the ode15i-solver with a BDF integration scheme (Shampine, 2002). If a boundary surface is met, located by the event detection algorithm of the solver, integration stops and the procedure from figure 3 is executed. When a solution for the transversal motion has been found, integration continues with the new values of  $\mathbf{z}$ .

The example is simulated for 1000 seconds, with the initial states  $\mathbf{x}(t = 0) = [37 \ 30 \ 40]^T$  and constant input signals  $\mathbf{u}(t) = [0 \ 80 \ 30 \ 40 \ 25 \ 35 \ 35 \ 45]^T \forall t$ . In figure 4 the trends of the states  $\mathbf{x}$ , valve positions  $z_1$ ,  $z_3$  and  $z_5$  and algebraic signal  $y_1$  can be seen. The simulation has taken 1.19 seconds. It can be seen that the system has state-continuity, which means that the states cannot jump in mode transitions and are smooth in between, while the algebraic variables  $\mathbf{y}$  can be discontinuous (jump) in between mode transitions. Even if the continuous-valued algebraic variables  $\mathbf{y} \subseteq \mathbb{R}$  are in general restricted to a subset  $\mathbb{Y}$  of the real number space  $\mathbb{R}$ , in this example the variable  $y_1$  is restricted to a subset of the natural number space  $\{1, 2, 3\} \subset \mathbb{N}$ , since the signal  $y_1$  is exclusively dependent on  $z_1$ ,  $z_3$  and  $z_5$ .

In the phase diagram in figure 5 it can be seen, that the trend of the states has corners at the discontinuity surfaces marked by the black squares. More over the diagram reveals a period-2 limit cycle.

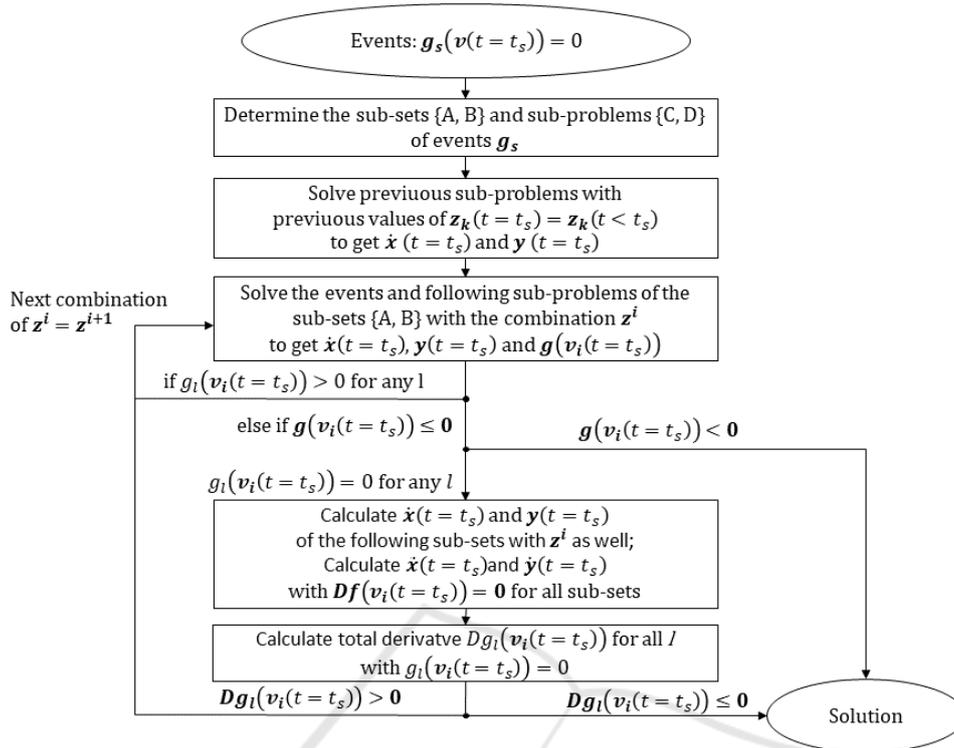


Figure 3: Brute force procedure for solving the discontinuous motion problem when sub-sets and sub-problems are known.

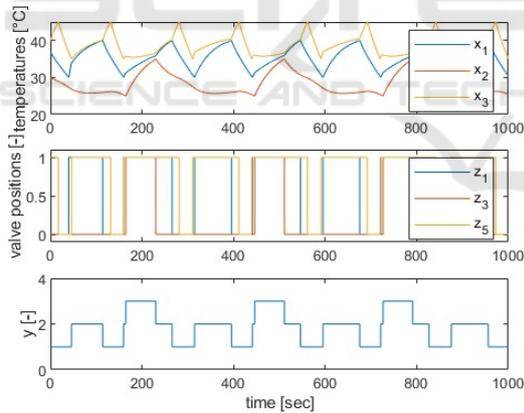


Figure 4: Time signals of the simulations results of the three tank model.

## 6 CONCLUSION

The publication focuses on the transversal motion discontinuities of piecewise smooth implicit multilinear initial value problems. Unlike usual switched systems with explicit descriptions of modes, switching conditions, and transition functions, the hybrid iMTI models implicitly encode mode information in binary-valued variables, with switching conditions and tran-

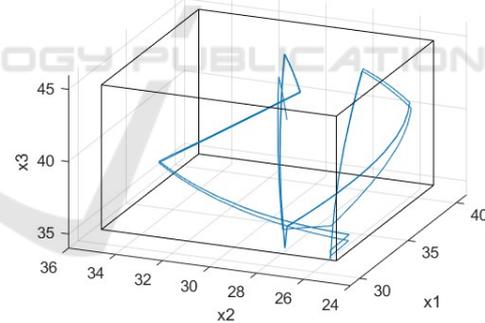


Figure 5: Phase diagram of the simulations results of the three tank model.

sitions defined by inequality constraints.

A method has been introduced to determine the implicitly described discontinuity and solve the transversal motion problem by adapting an idea from explicit piecewise smooth ODEs based on the total derivative of mode constraints.

A 2-point temperature-controlled three-tank system is used to illustrate the application of the hybrid iMTI modeling framework, as well as the method to solve the piecewise smooth IVP. The method for handling mode switching when discontinuity surfaces are met involves solving an inequality-constrained sys-

tem of equations at the event time for different combinations of binary variables and checking conditions based on the function values and their total derivatives to determine the system's behavior.

To address the combinatorial complexity arising from the binary variables, it is proposed to use sparsity pattern analysis to identify sub-sets and sub-problems within the system, which can be solved more efficiently. The simulation results of the example show the applicability of the method. The system's behavior at discontinuity surfaces can be seen and reveal a limit cycle.

## REFERENCES

- Baharev, A., Neumaier, A., and Schichl, H. (2019). A manifold-based approach to sparse global constraint satisfaction problems. *Journal of Global Optimization*, 75(4):949–971.
- Benzaouia, A. (2012). Introduction to switched systems. In *Saturated Switching Systems*, pages 77–94. Springer.
- Calvo, M., Montijano, J., and Rández, L. (2016). Algorithm 968: Disode45: A matlab runge-kutta solver for piecewise smooth ivps of filippov type. *ACM Transactions on Mathematical Software*, 43:1–14.
- Dieci, L. and Lopez, L. (2011). Fundamental matrix solutions of piecewise smooth differential systems. *Math. Comput. Simul.*, 81:932–953.
- Dulmage, A. L. and Mendelsohn, N. S. (1958). Coverings of bipartite graphs. *Canadian Journal of Mathematics*, 10:517–534.
- Engels, M., Lichtenberg, G., and Knorn, S. (2024). An approach to parameter identification for boolean-structured multilinear time-invariant models\*. In *2024 European Control Conference (ECC)*, pages 2077–2082.
- Fritzson, P. and Engelson, V. (1998). Modelica—a unified object-oriented language for system modeling and simulation. In *ECOOP'98—object-oriented programming: 12th European conference Brussels, Belgium, July 20–24, 1998 proceedings 12*, pages 67–90. Springer.
- Jöres, N., Kaufmann, C., Schnelle, L., Yáñez, C., Pangalos, G., and Lichtenberg, G. (2022). Reduced cp representation of multilinear models. *Proceedings of the 12th International Conference on Simulation and Modeling Methodologies, Technologies and Applications*.
- Kaufmann, C., Garcia, D. C., Lichtenberg, G., Pangalos, G., and Yáñez, C. C. (2023). Efficient linearization of explicit multilinear systems using normalized decomposed tensors. *IFAC-PapersOnLine*, 56(2):7312–7317.
- Lang, S. (2002). *Algebra. Graduate Texts in Mathematics*, volume 211, chapter Matrices and Linear Maps §4. Determinants, page 511. Springer New York.
- Lichtenberg, G. et al. (2022). Implicit multilinear modeling. *at - Automatisierungstechnik*, 70(1):13–30.
- Lichtenberg, G., Uhlenberg, E., Warnecke, T., Cateriano, C., Engels, M., Samaniego, L., Schnelle, L., and Kaufmann, C. (2024). MTI-Toolbox. <http://mti.systems>.
- Luxa, A. et al. (2022). Multilinear modeling and simulation of a multi-stack PEM electrolyzer with degradation for control concept comparison. *Proceedings of the 12th International Conference on Simulation and Modeling Methodologies, Technologies and Applications*.
- Mehrmann, V. and Wunderlich, L. (2009). Hybrid systems of differential-algebraic equations - analysis and numerical solution. *Journal of Process Control*, 19(8):1218–1228. Special Section on Hybrid Systems: Modeling, Simulation and Optimization.
- Pangalos, G., Eichler, A., and Lichtenberg, G. (2014). Hybrid multilinear modeling and applications. *Advances in Intelligent Systems and Computing*, page 71–85.
- Piironen, P. and Kuznetsov, Y. (2008). An event-driven method to simulate filippov systems with accurate computing of sliding motions. *ACM Transactions on Mathematical Software*, 34.
- Pinar, A. and Heath, M. T. (1999). Improving performance of sparse matrix-vector multiplication. *Proceedings of the 1999 ACM/IEEE conference on Supercomputing*.
- Pothen, A. and Fan, C.-J. (1990). Computing the block triangular form of a sparse matrix. *ACM Trans. Math. Softw.*, 16(4):303–324.
- Samaniego, L., Uhlenberg, E., Kaufmann, C., Engels, M., Pangalos, G., Cateriano Yáñez, C., and Lichtenberg, G. (2024). An approach to multi-energy network modeling by multilinear models. pages 1522–1527.
- Sanz, V., Urquia, A., and Casella, F. (2014). Improving efficiency of hybrid system simulation in Modelica. volume 2014.
- Shampine, L. (2002). Solving  $0=f(t,y(t),y'(t))$  in matlab. *Journal of Numerical Mathematics*, 10.
- Vassilevska, V. and Pinar, A. (2004). Finding nonoverlapping dense blocks of a sparse matrix. *Lawrence Berkeley National Laboratory*.
- Warnecke, T. and Lichtenberg, G. (2024a). Hybrid implicit multilinear modeling of complex hvac systems. In Wagner, G., Werner, F., and De Rango, F., editors, *Simulation and Modeling Methodologies, Technologies and Applications*, pages 79–104, Cham. Springer Nature Switzerland.
- Warnecke, T. and Lichtenberg, G. (2024b). Hybrid implicit multilinear simulation using difference algebraic equations reordering by sparsity patterns\*. In *International Conference on Control, Decision and Information Technologies 2024, 2024 10th International Conference on Control, Decision and Information Technologies (CoDIT)*, pages 2578–2583.