Nonlinear Hybrid Model Predictive Control for building energy systems

Maximilian Mork[a,*], Nick Materzok[a], André Xhonneux[a], Dirk Müller[a,b]

[a] *Institute of Energy and Climate Research - Energy Systems Engineering (IEK-10), Forschungszentrum Jülich GmbH, Wilhelm-Johnen-Straße, 52425, Jülich, Germany*

[b] *Institute for Energy Efficient Buildings and Indoor Climate, E.ON Energy Research Center, RWTH Aachen University, Mathieustraße 10, 52074, Aachen, Germany*

Abstract

This paper presents a nonlinear hybrid Model Predictive Control (MPC) approach for building energy systems based on Modelica. The MPC approach takes into account two characteristics that are very common for building energy systems: nonlinearities (inherent in the building envelope and Heating, Ventilation and Air Conditioning (HVAC) systems) and discontinuities (in the form of on/ off operation, discrete operation states and operation modes). The hybrid MPC approach integrates both continuous and discrete optimization variables into the control concept and thus is capable of controlling building energy systems with binary or integer decision variables, switching dynamics or logic if-then-else constraints. By employing a time-variant linearization approach, nonlinear Modelica optimization problems are approximated with high accuracy and transformed into a linearized state-space representation. Based on the linearization output, a linearized optimization problem is generated automatically in every MPC iteration, which is extensible by various integer characteristics and is accessible for a wide range of mixed-integer solvers. A simulation study on a nonlinear Modelica building energy system demonstrates the control quality of the proposed toolchain revealing a small linearization error and successful integration of multiple integer characteristics. The benefits of the approach are manifested by comparing its performance with different reference control strategies.

Keywords: Hybrid Model Predictive Control, Modelica, Building energy system optimization, HVAC, Nonlinear optimization, Mixed-integer optimization, Linearization

Nomenclature

Abbreviations	
BLT	Block-Lower Triangular
CCA	Concrete Core Activation
CIA	Combinatorial Integral Approximation
COP	Coefficient Of Performance
DAE	Differential Algebraic Equations
DP	Dynamic Programming
EDP	Economic Dispatch Problem
FMI	Functional Mockup Interface
FMU	Functional Mockup Unit

*Corresponding author

Email address: m.mork@fz-juelich.de (Maximilian Mork)

GDP	Generalized Disjunctive Programming
HVAC	Heating, Ventilation and Air Conditioning
KPI	Key Performance Indicator
LP	Linear Program
LTI	Linear Time-invariant
LTV	Linear Time-variant
MILP	Mixed-Integer Linear Program
MINLP	Mixed-Integer Nonlinear Program
MIQCP	Mixed-Integer Quadratically-Constrained Program
MIQP	Mixed-Integer Quadratic Program
MLD	Mixed Logical Dynamical
MPC	Model Predictive Control
NLP	Nonlinear Program
PID	Proportional-Integrative-Differential
PSO	Particle Swarm Optimization
QP	Quadratic Program
RBC	Rule-Based Control
RMSE	Root Mean Square Error
SLP	Sequential Linear Programming
SQP	Sequential Quadratic Programming
TABS	Thermally Activated Building Systems
UCP	Unit Commitment Problem

Subscripts

approx	approximated
bin	binary
cond	condenser
cons	consumer
el	electrical
eva	evaporator
HP	heat pump
in	inlet
op	optimization
quad	quadratic
ref	reference
stor	storage

1. Introduction

Approximately 40 % of the final energy consumption and 36 % of all CO_2 emissions in the European Union are attributed to the building sector [1, 2]. Long-term goals set by the EU aim at a reduction of greenhouse gas emissions by 2030 by at least 50 % compared to 1990, energy consumption reduction by 2050 by at least 50 % compared to 2005 and a carbon-neutral building stock by 2050 [2, 3]. Heating, Ventilation and Air Conditioning (HVAC) systems, which are responsible for providing comfort to the building users, contribute to a major extent to the energy used in buildings [4, 5]. Accordingly, they hold a large potential for a significant increase of building energy efficiency and reduction of energy consumption. The control of HVAC systems is complex due to inherent nonlinear dynamics with time-delays, time-varying set-points and disturbances and a high number of interactions [6]. Traditionally, building energy systems are controlled by Rule-Based Control (RBC) (on/ off or bang-bang control) or Proportional-Integrative-Differential (PID) controllers due to their simplicity and low computational complexity [7]. However, especially for large-scale buildings, these control strategies are challenging to tune. Moreover, they are not able to integrate system or comfort constraints, future disturbance quantities and to balance the conflicting optimization goals of comfort and energy reduction.

Model Predictive Control (MPC) is a promising control technique that demonstrates various benefits over the classical control strategies. Its anticipatory control is based on a building energy system model, is capable of integrating constraints and future disturbances as well as balancing conflicting optimization goals [8]. MPC has been implemented in buildings in the form of simulative or practical studies which have proven the simultaneous improvement of comfort and reduction of energy use. Applying MPC to an office building in Prague, energy savings of 15 to 28 % were obtained compared to the conventional heating curve based control [9]. The experimental demonstration of the MPC on an HVAC system of two office buildings in Australia resulted in energy savings of 19 % [10]. These figures coincide with other use-cases showing more than 20 % of energy savings for a research laboratory in Illinois [11], more than 20 % of primary energy reduction for an office building in Brussels [12] and a primary energy reduction of 17 % for a large-scale simulation in a Swiss office building [13].

Building energy systems exhibit a wide range of nonlinearities particularly introduced by the integrated HVAC systems. Furthermore, building energy systems are often characterized by discontinuities in the form of binary or integer optimization variables, operation modes including switching dynamics or logical if-then-else relationships. Integrating both continuous and discrete decision variables requires a hybrid MPC concept. The combined integration of both nonlinearities and discontinuities in an optimization problem leads to Mixed-Integer Nonlinear Programs (MINLPs) which are extremely difficult to solve. Apart from the numerical challenges of the optimization problems, the success of implementing MPC on a large scale strongly depends on the modeling process and efforts as this part takes up most of the time during an MPC development against the background that every building is unique [9]. Among different modeling languages, Modelica stands out due to its modularity, flexibility and extensive open-source simulation libraries for buildings and HVAC systems which are developed and maintained by experts.

This work is structured as follows. In Section 1.1, a literature review is given on nonlinear hybrid MPC including tractable approximations of these characteristics in optimization problems. Section 1.2 details the contributions of the proposed MPC approach. In Section 2, a building thermal zone and a coupled buffer storage and heat pump model are introduced which build the simulative test bed for the implementation of the proposed nonlinear hybrid MPC. Section 3 focuses on the methodology and the different modules of the nonlinear hybrid MPC approach. In Section 4, the MPC method is implemented on the simulative case study and the performance is compared to different reference control strategies. The paper completes with a conclusion and an outlook on potential extensions of the framework.

1.1. Background

Linear MPC is the most widely applied class for MPC implementations in the building sector [14]. However, nonlinearities in buildings arise from specific physical phenomena in the building envelope (such as heat convection, radiation or absorption and transmission of solar gains through windows) as well as from HVAC systems (e.g. heat pumps, fans, pumps) and corresponding working curves and tables. Compared to linear MPC, nonlinear MPC preserves its accuracy in reproducing the building behavior over a broader range of operating conditions. It allows for higher flexibility in the formulation of the optimization problem including the system dynamics and cost function and enables an exploitable MPC potential closer to the theoretical performance bound [15]. In linear MPC, often only intermediate quantities or set-points are calculated optimizing an approximation of the cost function whereas the exact conversion into actuator signals is allocated to a post-processing or a suboptimal low-level controller. The use of nonlinear MPC comes at the cost of a higher computational demand but computational power capacities are significantly increasing within the last years based on improvements of optimization solvers and the use of server and cloud computing [7].

Nonlinear system behavior can be approximated by linear models applying different linearization techniques and relinearization frequencies. Linearized models can be classified into linear time-invariant (LTI) and time-variant (LTV) models. Most building MPC implementations employ optimization formulations based on LTI models [6]. Nonlinear models that are approximated by LTI models are linearized and approximated only once before the MPC execution (accordingly classified as *offline* linearization). The linearization is generally carried out around a certain equilibrium or operation point and it is assumed that the system is controlled in a range close to this reference point during operation. LTI models can be calculated via Jacobian linearization around operating points [16, 17, 18, 19] or identified based on parameter estimation approaches [20, 21, 22]. For building envelope dynamics, Picard et al. [23] create LTI models with a special focus on approximations for the nonlinear heat transfer phenomena of convection and radiation around a working point. Another type of LTI model is constituted by linear piecewise models which reproduce the nonlinear building and HVAC behavior over a wider operating range. They are created based on linearization around different operating points, which are representative for an operating range each and introduce additional binary variables into the optimization problem for each operating range [24, 25].

LTV models, on the other hand, are relinearized during an MPC execution (therefore classified as *online* linearization) and linearization reference points and corresponding linearization matrices are updated continuously. By updating the reference points and linearization matrices, the accuracy of the linearized models can be increased as the operating conditions used as a reference for the approximation are adjusted based on the behavior of the dynamic system. The classification of LTV models can be further subdivided into linearization around a reference point or a reference trajectory.

In the case of linearization around a reference point, the linearization matrices and reference points are updated for every MPC iteration but remain constant during the prediction horizon of each MPC iteration. This approach is followed by Pčolka et al. [26, 27] controlling Thermally Activated Building Systems (TABS) fed by a hot water storage tank. In their work, auxiliary variables forming the coefficients of the linearization matrices are introduced which are updated based on the most recent measurements representing the current operation point in every MPC iteration.

For the LTV models based on linearization around a reference trajectory, the reference points and linearization matrices are updated for every MPC iteration and additionally in each MPC iteration for every reference point along a reference trajectory. In an MPC execution, the reference trajectory can be generated by applying the optimization results from the previous MPC iteration to the building simulation model. In contrary to the moving horizon MPC scheme, where only the input for the first sampling period is applied, a simulation is conducted over the full prediction horizon to obtain new reference trajectories. Using multiple reference points for the linearization in each MPC iteration, the nonlinear model is approximated at or close to reference points at which the system will be operated assuming the application of the previous optimal control inputs. By applying this relinearization technique and increasing the number of reference points along the trajectory, the gap between the nonlinear and the linear time-invariant model is continuously bridged. The technique of linearizing LTV models around a reference trajectory is pursued in [28, 29] operating a heat pump connected to a building including floor heating and a thermal storage.

A similar technique based on linearization around a trajectory can be employed to solve optimization problems based on bilinear models. Bilinear models are created if the differential algebraic equations (DAE) incorporate products of states and/ or control inputs. A common modeled bilinear phenomenon appears in ventilation models where the heating/ cooling energy supplied to a room is quantified by multiplying the control input (mass flow rate) with the state (difference of room air temperature entering and leaving the room (state)). The bilinear problem can be solved by Sequential Linear Programming (SLP) (for a linear cost function; Sequential Quadratic Programming (SQP) for a quadratic cost function): During each MPC iteration, the nonlinear optimization problem (cost and constraint functions) is iteratively linearized around the current solution trajectory, solved and repeated until convergence is achieved. Bilinear controller models are presented in [30, 31] where linear models are used for the building dynamics and bilinear models for the HVAC and shading models.

The simultaneous integration of nonlinear and integer characteristics into an optimization problem leads to MINLPs which are extremely difficult to solve [14]. Using MINLP solvers in real-time control applications may fail due to computation time and intractability issues. Therefore, nonlinear hybrid MPC implementations for building energy systems generally avoid MINLPs apart from a few exceptions [32, 33]. In a multitude of nonlinear hybrid MPC model implementations, the MINLP is converted into a Mixed-Integer Linear Program (MILP). In a nonlinear operational optimization of a stratified thermal storage including switching dynamics due to different cooling modes, Berkenkamp and Gwerder [34] propose an LTV model relinearized in every iteration based on the previous optimization results. Feng et al. [35] and Khakimova et al. [36] introduce LTI models as a base for a nonlinear hybrid building MPC including switching dynamics for different operation modes. Based on offline Jacobian linearization of the nonlinear dynamics around an operating point, Mayer et al. [19] implement a building MILP including switching dynamics depending on the charging state of a storage and the on/ off operation state of a heat pump.

A further technique to reduce the computational complexity of an MINLP is the decoupling of the MINLP into subproblems which solve the overall problem by coordination or communication. In an operation of multiple chillers and a thermal storage providing thermal demand of a campus, Deng et al. [37] split an MINLP into a hierarchical optimization of a Dynamic Programming (DP) and an MILP problem. The DP generates an optimal operation profile of the thermal storage which is is used as input for the MILP calculating optimal chiller operations based on time-variant linearization of the nonlinear dynamics. Fiorentini et al. [38] tackle an MINLP for a solar-assisted HVAC system of a residential building including switching dynamics applying a similar hierarchical procedure. The high-level optimization calculates the optimal sequence of the operation modes and the low-level optimization optimizes the operation mode selected by the high-level. Discrete fan speeds are introduced to eliminate the model nonlinearity in both layers. For control of a home energy management system, Huang et al. [39] decouple the MINLP into a particle swarm optimization (PSO) and an SQP. In this approach, the PSO generates the initial guesses for the discrete and continuous values to be used by the SQP which just reoptimizes the continuous variables. In an operational optimization of a district heating system, Schweiger et al. [40] decompose the original MINLP into a unit commit problem (UCP) formulated as Mixed-Integer Quadratically-Constrained Program (MIQCP) and an economic dispatch problem (EDP) formulated as Modelica-based Nonlinear Program (NLP). The UCP calculates the optimal schedule and operation status for all productions units which is used as input for the EDP calculating optimal control variables for the production units. Zhong et al. [41] decouple an MINLP for an integrated electricity and heating system into an MILP for the power network and an NLP for the heating network providing thermal comfort to a building. The overall problem is solved by a distributed optimization approach.

There are further approaches that propose to determine the scheduling of different operation modes in a heuristics-based pre-processing step before a continuous optimization. Ma et al. [42] select storage tank operation modes in a pre-optimization stage based on heuristics of the plant operation and cost including fixing the timing of the respective modes. Based on the fixed tank operation modes the optimal chiller control is determined in an NLP satisfying a required cooling load. For a thermal storage management coupled to an HVAC system and a building, Touretzky and Baldea [43] prespecify the sequence of the operation modes within the prediction horizon. Based on the prefixed sequence, the timing and duration of each respective operation mode is optimally determined in an NLP using exclusively continuous variables.

An alternative approach to solve an approximated form of an MINLP is developed by Bürger et al. [44] implementing the software package *pyycombina* and is applied on a solar thermal climate system coupled to a thermal zone. In a first step, an NLP as a result of a relaxed MINLP is solved eliminating the binary constraints. The obtained relaxed binary constraints are approximated by non-relaxed binary constraints in a Combinatorial Integral Approximation (CIA) formulated as an MILP. In the last step, an NLP is solved again with the fixed binary variables from the CIA to adjust the continuous variables.

Reviewing potential modeling frameworks for MPC implementations, Modelica is a modeling language that suits the use in an MPC realization as it is an open-source, equation-based, acausal and object-oriented modeling language with a user-friendly graphical interface to connect components [45]. Due to the fact that every building is unique, a modeling language is required to be flexible and modular which is fulfilled by Modelica. In common projects, large-scale open-source Modelica simulation libraries including models of buildings and HVAC systems are developed and maintained by experts [46]. JModelica.org [47] is a software tool that is capable of gradient-based optimization of Modelica models and is equipped with an interface to IPOPT [48], an open-source solver to solve large-scale nonlinear problems. As characteristic of general NLP solvers, IPOPT requires models to have constraints and cost functions that are twice continuously with respect to the optimization variables. Accordingly, integer decision variables, binary on/ off variables, discrete operation states or operation modes are not supported. This limits the use of JModelica.org for building control problems where HVAC systems often include on/ off control, discrete control states, logical if-then-else relationships or switching dynamics due to different HVAC modes. To integrate discrete, integer or Boolean decision variables into JModelica.org, varied approaches are pursued to approximate the mixed-integer problems by employing post-processing and heuristics [49] or by using model approximations with relaxed continuous decision variables and subsequent mapping [50]. Accordingly, the automatability and applicability of JModelica.org and extensions such as TACO [50] to an arbitrary building and HVAC type are restricted. Jorissen [50] and Schweiger et al. [51] identify the support of integer decision variables as a major potential and challenge in the field of optimization of Modelica models.

1.2. Contribution

The major contribution of this work consists of a nonlinear hybrid MPC approach for building energy systems based on Modelica which bridges the gap between Modelica and mixed-integer optimization. It simultaneously accounts for the inherent nonlinearities of building energy systems and integer decision variables. The framework supports discrete operation states, binary on/ off control, logical if-then-else expressions, switching dynamics due to different operation modes and further building-relevant integer characteristics detailed in Section 3.2. Based on JModelica.org, a linearization module is implemented which is capable of performing time-invariant (LTI) and time-variant linearization (LTV) around a reference point or trajectory based on generated simulation trajectories from the previous MPC iteration. The linearization module generates a state-space formulation of the Modelica optimization problem including a quadratic approximation of the potentially nonlinear cost function. The resulting mathematical formulation is a self-contained form of the optimization problem which can be accessed and implemented into different modeling frameworks and corresponding interfaced optimization solvers. In this framework, the linearized optimization problem is generated automatically in Pyomo [52] in every MPC iteration based on the output of the linearization module, extended by problem-specific integer characteristics and coupled to the state-of-the-art MILP/ MIQP (Mixed-Integer Quadratic Program)/ MIQCP solver Gurobi [53]. As the linearization and the mapping of Modelica onto mathematical variables are performed automatically, there is no need for manual model approximation or linearization, formulation of piecewise linear models or post-processing preserving a high automatability and wide applicability of the framework in building control.

The overall framework allows for optimizing Modelica optimization problems as NLP or MILP/ MIQP/ MIQCP using IPOPT or any Pyomo-interfaced solver. It is evaluated in a holistic optimization of energy demand and supply of a thermal zone model supplied by a thermal buffer storage and a heat pump. In contrary to various works on building MPC which focus on accurate modeling and optimization of either the demand or the supply side ([54, 34, 43]), here both domains are modeled with high accuracy and optimized in a hierarchical structure.

In summary, the innovations of this paper within the scope of MPC for building energy systems are:

- Nonlinear hybrid MPC for building energy systems based on Modelica
- Time-variant linearization approach for nonlinear Modelica optimization problems
- Automated creation of the linearized optimization problem in Pyomo in every MPC iteration, extension by integer characteristics and coupling to the mixed-integer optimization solver Gurobi

2. Modeling

In this work, the open-source Modelica simulation library AixLib [55] forms the base for generating the building controller and simulation models. Based on this library an optimization library is developed by adjusting the AixLib models to be compatible with the used optimization framework JModelica.org and the solver IPOPT. For more details regarding the necessary modifications of the AixLib models, the reader is referred to [56]. Modified AixLib models are validated through comparative simulations.

In this case study, both the energy demand and supply domain are accurately modeled and optimized. The energy demand model is constituted by a thermal zone model. An overview of the thermal zone model including actuators and control inputs is given in Fig. 1. The zone model comprises a dynamic air model, external and inner walls, windows with modeled external Venetian blinds for active solar shading [56], simplified pumps, a convector and TABS implemented as Concrete Core Activation (CCA). The pumps provide heating water flows at fixed temperatures to the convector and CCA. Occupancy is taken into account by a model calculating human heat emissions according to a characteristic office schedule (8 a.m. – 12 p.m. and 1–6 p.m.). Weather data is used based on a historic resource file from the AixLib for San Francisco from January 1999 which implies a heating period. Control inputs to the thermal zone are the water mass flows to the convector and CCA as well as the vertical position and inclination angle of the Venetian blinds considering both thermal and visual comfort (minimum illuminance of 500 lux in occupied times). The artificial lighting power consumption is assumed to vary linearly with the remaining required artificial illuminance. The thermal comfort range spans from 293–295 K.

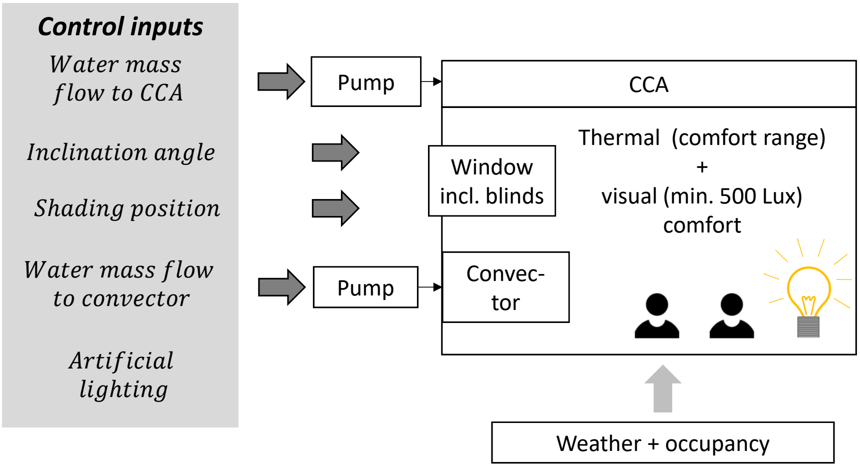


Figure 1: Overview of the thermal zone including actuators and control inputs

The energy supply model consists of a coupled thermal buffer storage and a heat pump. The buffer storage provides the heating power requested by the thermal zone model and couples the heating pump circuit with the circuit providing the heating water to the heating actuators of the thermal zone (convector and CCA). The modeled components for the supply side can be considered characteristic of current and future electrified heating energy systems including the need for a storage system due to the volatility of renewable energy generation. The basic structure of the energy supply model and the integration into the overall energy system are depicted in Fig. 2. Control inputs to the supply model (marked with a gray background) are the water mass flow between storage and consumer $\dot{m}_{\mathrm{stor,cons}}$, the water mass flow through the condenser of the heat pump $\dot{m}_{\mathrm{cond}}$, the thermal power supplied by the condenser $\dot{Q}_{\mathrm{cond}}$ and the on/off operation state of the heat pump δ_{HP}. Simplified pumps are modeled as ideal flow sources supplying prescribed mass flows between storage and heat pump as well as storage and building.

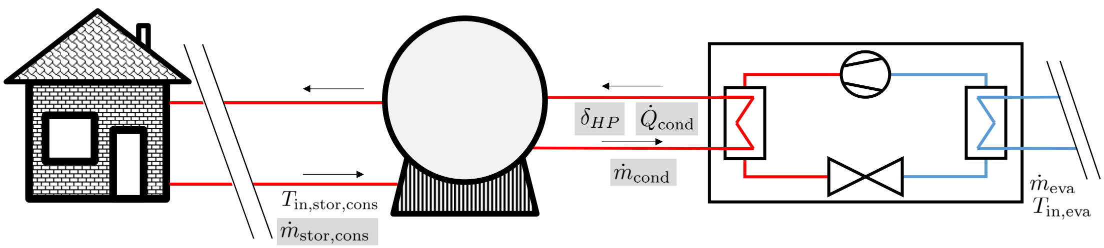


Figure 2: Scheme of the energy supply model and integration in the overall energy system

The heat pump model is based on the AixLib model *AixLib.Fluid.HeatPumps.Carnot_y* which is reformulated by manipulating the heating power provided by the condenser $\dot{Q}_{\text{cond}}$ instead of the part load ratio of the compressor. The coefficient of performance (COP) of the heat pump quantifying the ratio of thermal power delivered by the condenser to the electrical power added to the compressor is adjusted based on the Carnot efficiency, which is a function of the outlet temperatures of the condenser and evaporator. When using this model attention should be given that sufficient mass flows enter the condenser and evaporator otherwise high, unrealistic temperatures can occur. In this case study, only the control inputs of the condenser side are manipulated within the optimization problem; the controlled evaporator mass flow $\dot{m}_{\text{eva}}$ varies linearly with the evaporator thermal power and the inlet evaporator temperature $T_{\text{in,eva}}$ is fixed.

The model for the buffer storage is based on the AixLib *AixLib.Fluid.Storage.BufferStorage* model. The water storage is stratified into three layers, which are connected allowing heat and fluid transfer. The top storage layer is coupled to the water returning from the heat pump and the water circuit supplying the consumer. This layer incorporates the highest water temperatures. The bottom storage layer is coupled to the water supplied to the heat pump water circuit and the water returning from the consumer. The heat transfer model *HeatTransferBuoyancyWetter* is used for modeling buoyancy in the storage.

The energy supply layer provides the thermal power $\dot{Q}_{\text{ref}}(t)$ requested by the energy demand layer consisting of the thermal zone model which accounts for both the heating power supplied to the convector and CCA. The mixed temperature of the water flow returning from convector and CCA is equal to the inlet temperature $T_{\text{in,stor,cons}}$ flowing into the storage. The scheme of the hierarchical optimization concept is shown in Fig. 3.

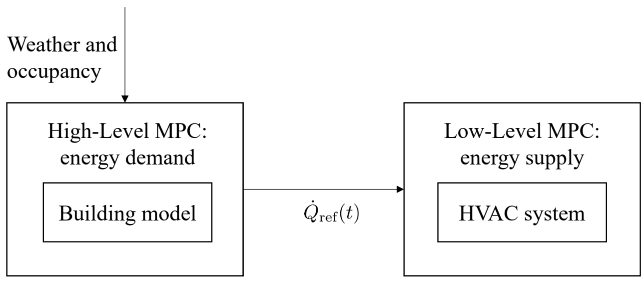


Figure 3: Hierarchical optimization approach for energy demand and supply

3. Methodology

In order to couple the nonlinear hybrid Modelica-based MPC to a real-time capable mixed-integer solver, a linearization tool is developed based on an extension of JModelica.org. In Section 3.1, the original linearization module is extended towards a time-variant linearization around multiple reference points, which allows for a continuously updated linearization along a simulation trajectory. The simulation trajectory is generated by applying the optimization results from the previous MPC iteration to the simulation model. The extended linearization tool further creates a quadratic approximation of the original cost function and includes inequality constraints. In the final framework execution, the linearized optimization problem is automatically generated in Pyomo in every MPC iteration based on the output of the linearization module and can be extended by various integer characteristics that are specified in Section 3.2.

3.1. Linearization

The linearization approach builds on the Python submodule *linearization* of JModelica.org which integrates CasADi [57] for automatic differentiation and calculation of first-order gradients. Input to the module is a compiled Modelica optimization problem and a simulation trajectory from a Functional Mockup Unit (FMU). The FMU constitutes a model unit to which the Modelica simulation model is compiled based on PyFMI [58] of JModelica.org according to the Functional Mockup Interface (FMI) standard for the exchange of compiled dynamical models between different modeling and simulation tools. In the original *linearization* submodule, just a linearization approach around a reference point is implemented generating only linearization matrices for the mathematical DAEs. The basic submodule is extended by:

- Time-variant linearization around multiple reference points along a simulation trajectory
- Quadratic approximation of the cost function
- Integration of inequality constraints
- Saving of the linearization output in an efficient, self-contained form
- Rounding of the coefficients of the linearization matrices

The listed extensions allow for linearization of an entire optimization problem including an approximated cost function and inequality constraints, further increase the linearized model accuracy by applying time-variant linearization and enhance automatability and reusability of the optimization formulation. The different extensions are detailed in the following.

The linearization procedure starts with a grouping of the original Modelica variables into the different variable types of state derivatives dx, states x, inputs u, algebraic variables w and potential free parameters p. A mapping dictionary is created where every Modelica variable is mapped to a specific variable type and a type-specific index (for example {"BufferStorage.T" : ("w",7)}). In the following step of the extended linearization module, for every DAE equation, a Jacobian linearization is performed either around a reference point or a reference trajectory (= multiple reference points). For this purpose, the CasADi function *jacobian()* is used. Based on the linearization matrices and reference points, a mathematical state-space representation of the DAEs of the optimization problem is created.

Three different settings for the relinearization frequency can be chosen for the linearization process during an MPC execution:

- Time-invariant linearization (LTI)
- Time-variant linearization around a reference point (point-LTV)
- Time-variant linearization around a reference trajectory (trajectory-LTV)

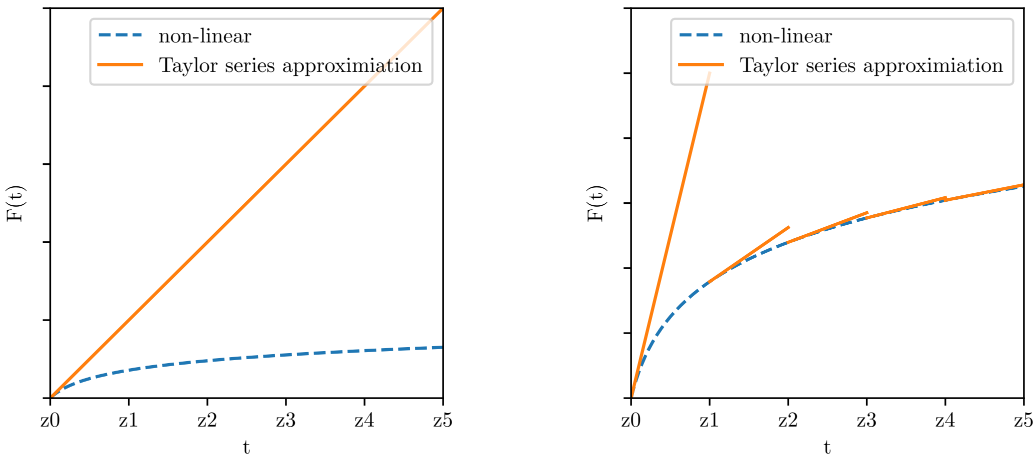
This allows for a flexible formulation of the Modelica optimization problem according to the nonlinearity and dynamics of the modeled system as well as the desired model accuracy employing LTV models for higher model accuracy and reduced linearization errors. In the time-invariant approach, the linearization matrices and corresponding reference points are fixed throughout the MPC execution. In the time-variant approach around a reference point (referred to as point-LTV) the matrices and reference points are updated for every MPC iteration but are fixed in every prediction horizon of each MPC iteration. In contrast, in the time-variant approach around a reference trajectory (referred to as trajectory-LTV) the matrices and reference points are both updated for every MPC iteration and in the course of each prediction horizon of each MPC iteration around multiple reference points along a trajectory. The conceptual difference between a time-invariant and time-variant linearization is shown in Fig. 4 based on a nonlinear function $F(t)$ and multiple reference points z.

The nonlinear DAEs of the original optimization problem can be described by:

$$0 = F(t, dx, x, u, w, p) \tag{1}$$

The resulting LTI model linearized around one time-invariant reference point $z_0 = (t_0, dx_0, x_0, u_0, w_0, p_0)$ is represented by the following state-space representation:

$$E \cdot (dx - dx_0) = A \cdot (x - x_0) + B \cdot (u - u_0) + C \cdot (w - w_0) + D \cdot (t - t_0) + G \cdot (p - p_0) + h \tag{2}$$



(a) Time-invariant

(b) Time-variant

Figure 4: Comparison of a time-invariant and time-variant linearization

, with the time-invariant matrices E corresponding to $\left.\frac{\partial F}{\partial dx}\right|_{z_0}$, A corresponding to $-\left.\frac{\partial F}{\partial x}\right|_{z_0}$, B corresponding to $-\left.\frac{\partial F}{\partial u}\right|_{z_0}$, C corresponding to $-\left.\frac{\partial F}{\partial w}\right|_{z_0}$, D corresponding to $-\left.\frac{\partial F}{\partial t}\right|_{z_0}$, G corresponding to $-\left.\frac{\partial F}{\partial p}\right|_{z_0}$ and h corresponding to $F(t_0, dx_0, x_0, u_0, w_0, p_0)$.

The dynamics for an LTV model based on linearization around multiple reference points extend the LTI formulation by time-varying terms for the linearization matrices and reference points $z_0(t) = (t_0(t), dx_0(t), x_0(t), u_0(t), w_0(t), p_0(t))$:

$$\begin{aligned} E(t) \cdot (dx - dx_0(t)) =& A(t) \cdot (x - x_0(t)) + B(t) \cdot (u - u_0(t)) + \\ & C(t) \cdot (w - w_0(t)) + D(t) \cdot (t - t_0(t)) + G(t) \cdot (p - p_0(t)) + h(t) \end{aligned} \tag{3}$$

In the next step, a quadratic approximation of the original cost function is formed by calculating the Jacobians and Hessians with respect to every variable type. The chosen setting of either applying a time-invariant or time-variant (point- or trajectory-LTV) linearization of the model DAEs also applies to the matrices and reference points of the approximated cost function. Creating the optimization problem just based on the Jacobians results in a Linear Program (LP) whereas by using both the Jacobians and Hessians the cost approximation accuracy can be further increased resulting in a Quadratic Program (QP).

The time-variant formulation of the quadratic approximation of the original nonlinear cost function J evaluated at multiple reference points is given by the second-order Taylor approximation:

$$J_{quad,approx} = \sum_{z \in \{dx,x,u,w,t,p\}} \nabla J|_{z_0(t)} \cdot (z - z_0(t)) + 0.5 \cdot (z - z_0(t))^T \cdot H_J|_{z_0(t)} \cdot (z - z_0(t)) + m(t) \tag{4}$$

, where the vector z corresponds to the variable types of the optimization problem, $z_0(t)$ to the time-varying reference points of the respective variable type, ∇J to the time-varying Jacobians, H_J to the time-varying Hessians and m to the time-varying cost function offset $J(t_0(t), dx_0(t), x_0(t), u_0(t), w_0(t), p_0(t))$.

In order to generate the output of the linearization module in a form that is also accessible outside the Python script, all relevant linearization results and information are also saved to *.csv* and *.xlsx* files. The aggregation of all saved files provides a self-contained form of the linearized optimization problem which can be integrated into different modeling tools and optimization solvers. Apart from the time-invariant or time-variant reference points, linearization and cost function matrices a *.xlsx* file is created containing information about the dimensions of the respective matrices and vectors, the variable bounds, the variable mapping and the initial state values. A focus is placed on a data-efficient way of saving by writing only the non-zero coefficients of the linearization matrices, Jacobians and Hessians to the output file. If an index pair of a matrix coefficient does not appear in the header of the respective *.csv* file it is assumed to be zero. For a linearized optimization in JModelica.org, there is also the possibility to automatically generate a *.mop* file in the Modelica-extension language Optimica [59].

The linearization performance can be adjusted by activating a rounding scheme which performs a rounding for all calculated matrices and vectors according to a chosen accuracy, e.g. using just the first three digits after the first non-zero digit. By activating the rounding a reduced amount of data is saved and the tractability for an optimization solver can be improved.

A benefit of the chosen approach consists in creating a linearized formulation of the optimization problem that works without introducing intermediate or surrogate quantities or variables. The generated optimization problem can be optimized with respect to the control inputs that are implemented in the real system. Apart from the automatic variable mapping (Modelica variables onto mathematical variables and vice versa) no further pre- and post-processing is necessary. As no model approximations are necessary, the automation degree is very high and the output of the linearization module can be used to automatically create an optimization problem in a modeling language which is implemented in the proposed framework based on Pyomo. In contrast to linear piecewise models which approximate the nonlinear system behavior via piecewise linear functions, no additional binaries are introduced into the problem which would increase the computational complexity. By updating the reference points and linearization matrices for every MPC iteration and during the prediction horizon of each MPC iteration it can be ensured that the linearization is performed around operating points at which or close to which the system is operated. Accordingly, the linearization error can be reduced and the model accuracy can be improved. The rounding of the matrix and vector coefficients can balance the trade-off of model accuracy versus the saved amount of data and tractability of the optimization problem.

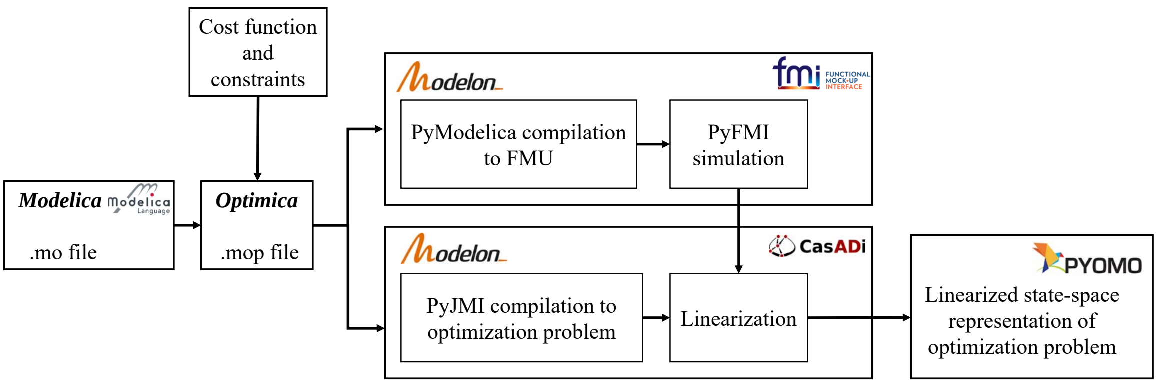


Figure 5: Linearization toolchain based on JModelica.org and Pyomo

Based on the output of the linearization module the optimization problem is automatically generated in the developed toolchain using Pyomo and *pyomo.dae* [60]. The full linearization toolchain for creating the linearized state-space representation of the optimization problem is presented in Fig. 5. The automatic generation of the optimization problem includes dimensioning of all vectors and matrices as well as setting the time-(in)variant coefficients of all linearization matrices and reference points. Likewise, the variable bounds and the approximated cost function including optional constraint softening are specified. Using the variable map generated by the linearization module, forecasts for disturbance quantities (e.g. weather, occupancy) or external data (e.g. energy prices) can be integrated by mapping the Modelica variables on the mathematical variables. After solving the optimization problem with a Pyomo-interfaced solver, the calculated control inputs are mapped back onto the Modelica variables and applied in an FMU simulation of the nonlinear Modelica model over the prediction horizon. In the MPC loop, the simulation trajectory calculated by the FMU simulation serves as the reference trajectory for the linearization step of the next MPC iteration as shown in Fig. 6. In order to avoid excessively large dimensions of linearization matrices and cost function Jacobians and Hessians, a variable reduction step is conducted before executing the MPC. A block-lower-triangular (BLT) decomposition method developed by Magnusson and Åkesson [61] in the JModelica.org framework is employed onto the nonlinear Modelica optimization problem. Solving some of the algebraic equations of the DAE in a preprocessing step, the number of algebraic variables and DAEs can be significantly reduced which reduces computation time and linearized controller model complexity.

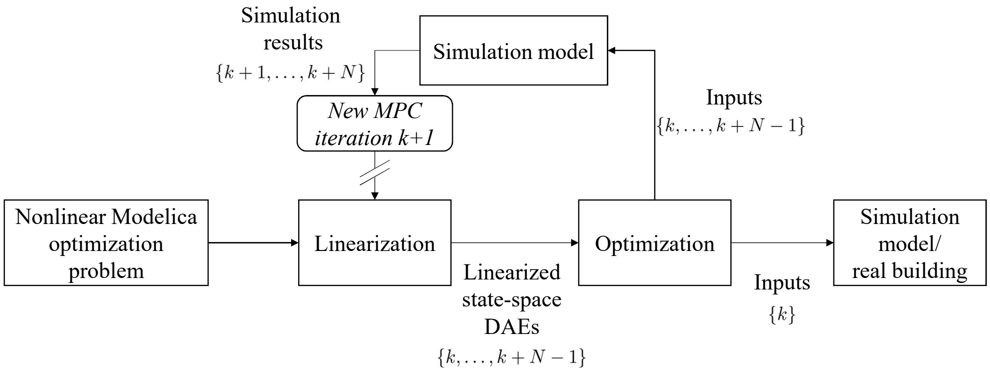


Figure 6: Linearization process during the MPC execution (k: current MPC iteration index, N: prediction horizon)

3.2. Integration of integer characteristics

Based on the linearized state-space representation of the optimization problem, the developed toolchain enables the integration of different integer characteristics. The incorporation of at least one of the described integer characteristics results in either an MILP (cost function based on Jacobians) or an MIQP (cost function based on Jacobians and Hessians).

First, discrete set constraints can be applied to a variable constraining the respective values to be chosen from a set of discrete operation states, e.g. prescribed by the hardware. The discrete set constraint can be formulated by:

$$u_{\text{discrete}}(t) = \sum_{k=1}^{\text{N}_{\text{op\_states}}} \delta_{k,\text{bin}}(t) \cdot s_k \qquad \forall \sum_{k=1}^{\text{N}_{\text{op\_states}}} \delta_{k,\text{bin}}(t) = 1 \qquad \forall t \in T \quad (5)$$

, where $u_{\text{discrete}}(t)$ corresponds to an exemplary variable constrained to adopt values from a discrete set, $\text{N}_{\text{op\_states}}$ to the number of elements in the discrete set, s_k to the values of the discrete set and $\delta_{k,\text{bin}}(t)$ to a binary variable indicating the activation status of each s_k. The sum of the binary variables $\delta_{k,\text{bin}}(t)$ is constrained to be equal to 1.

A binary constraint can be integrated by restricting a variable to binary values, e.g. in case of on/ off control behavior of system components. The binary constraint can be extended by including a minimum part load constraint, a common phenomenon known amongst others for heat pumps [62, 63]. It for example prescribes that a component can only be operated in a range of 30 % (minimum part load bound) to 100 % of the maximum power otherwise it is switched off. The minimum part load constraint can be formulated by:

$$\delta_u(t) \cdot u_{\min} \leq u(t) \leq \delta_u(t) \cdot u_{\max} \qquad \forall t \in T \qquad (6)$$

, where $u(t)$ represents the constrained variable, $u_{\min}$ the minimum part load bound, $u_{\max}$ a potential upper bound and $\delta_u(t)$ the binary on/ off state of the component.

The binary constraint can be further enhanced by integrating minimum times for a component to be operated in the on and/ or off state. The minimum on/ off times can be used to avoid oscillatory behavior which could be a potential disturbance to the building user as well as to reduce wear and tear and maintenance costs. The formulation of this constraint is detailed in Eq. 8g and 8h of the description of the optimization problem in Section 4.

In order to integrate building- and HVAC-characteristic operation modes into the optimization problem, the proposed framework makes use of *Pyomo.GDP*, a Pyomo extension for Generalized Disjunctive Programming (GDP) [64]. GDP allows for intuitive modeling of logical propositions (e.g. if-then-else relationships) and disjunctive relationships between sets of constraints. Different operation modes can be described by disjuncts forming selection alternatives for the optimization problem. Pyomo.GDP integrates an automated reformulation of the logical into linear constraints in a way that exhaustive algebraic transformations by the user (for example conversion of the Mixed Logical Dynamical (MLD) models into linear mixed-integer inequality constraints as described in [65]) are avoided. For this purpose, it employs different standardized methods such as the big-M, hull reformulation or the cutting-plane algorithm. Compared to other MILP-approaches, GDP facilitates a modular integration of both logical and algebraic constraints in a unified way and improves the interpretability of the optimization problem formulation [66].

The classical form of a GDP can be represented by [64]:

$$\min \quad f(x) \tag{7a}$$

$$\text{s.t.} \quad g(x) \leq 0 \tag{7b}$$

$$\bigvee_{i \in D_k} \begin{bmatrix} Y_{ik} \\ r_{ik}(x) \leq 0 \end{bmatrix} \qquad \forall k \in K \tag{7c}$$

$$\Omega(Y) = True \tag{7d}$$

$$x \in X \subseteq \mathbb{R}^n \tag{7e}$$

$$Y_{ik} \in \{True, False\} \tag{7f}$$

, where the minimization of an objective function (7a) is subject to global constraints (7b) and a set of logical disjunctions $k \in K$, each representing a major discrete decision. Each disjunction k consists of different selection alternatives, the disjuncts D_k (7c), which are linked by a logical OR-operator $\vee$. The i-th disjunct in the disjunction k is associated with the logic variable Y_{ik} (7f). When the Boolean variable Y_{ik} is *True*, the respective constraints $r_{ik}(x) \leq 0$ are imposed. There might be further logical propositions (7d) modeling the relationship between the logic variables, often defining an exclusive-OR (applied to the operation modes in the case study in Section 4) or in a "at most one" relationship. x is assumed to correspond to continuous variables that are closed and bounded in the set X (7e).

The limitations respectively potential improvements of the toolchain consist in the manual formulation of the different integer characteristics in Pyomo as they are not part of the Modelica (simulation) model itself. The current generalized formulation allows an automated formulation in Pyomo for which just the respective constrained variable, discrete set values, minimum part load bounds, minimum on/ off times or operation mode characteristics have to be specified. By making use of the GDP and its disjunctive sets, the implementation of the operation modes is straightforward and intuitive. A disadvantage of the GDP formulation is the necessary provision of variable bounds for each GDP-integrated variable, which does not allow a constraint softening of the respective variable bounds.

Aside from that, the proposed approach does not support specific Modelica expressions, which are not permitted by the CasADi compilation module of JModelica.org such as *integer(x)*, *pre(x)*, *floor(x)* and *sign(x)*, which are expressions that introduce discontinuities and are not supported by IPOPT (further unsupported expressions can be found under Chapter 9 in [67]). Model parts that include these expressions have to be transferred from Modelica to the modeling section in Pyomo and coupled to the linearized DAEs at this point.

4. Results and discussion

The performance of the developed toolchain is evaluated on the nonlinear Modelica use case of the coupled optimization of energy demand and supply presented in Section 2. According to Fig. 3, the optimization problem is split into a hierarchical optimization where the energy demand constitutes the upper layer and the energy supply represents the lower layer MPC. The optimization problems are coupled by the reference variable of the requested heating power $\dot{Q}_{\text{ref}}(t)$ which is supplied to the convector and CCA in the energy demand model. In every MPC iteration, a new trajectory of the requested heating power over the prediction horizon is sent from the upper to the lower layer MPC. It is assumed that the energy demand optimization problem does not include any integer decision variables which is why it can be solved as a pure NLP based on JModelica.org and IPOPT. The lower level optimization problem of the energy supply includes various integer characteristics and thus, is handled by the proposed toolchain presented in Section 3.

In the following, the different integer characteristics of the energy supply MPC are specified. The heat pump is required to be operated above a part load ratio of 30 % of the nominal compressor power which implies that below this threshold it is switched off. The on/ off control introduces a binary variable to the optimization problem. In addition, the operation of the heat pump is restricted by minimum on and off times. A minimum time of 2 h for both the on and off operation state is set to avoid oscillatory behavior and decrease tear and wear and maintenance costs. It is assumed that the pump located in the water circuit between the storage and the heat pump can only be operated in discrete operation states. In order to evaluate the implementation of operation modes, two different modes are established which refer to the control of the heat pump and the pump in the circuit between the storage and heat pump. The first operation mode describes the on operation state of the heat pump and allows for an unconstrained operation of the pump supplying the water to the condenser by manipulating $\dot{m}_{cond}$. In the second operation mode, during which the heat pump is switched off and no thermal power is provided to the heating water circuit, the pump is restricted to provide a water flow at the minimum level $\dot{m}_{\text{cond,min}}$. Thereby, unnecessary pump operation and transport of water flow (which is not taken into account in the cost function) is avoided reducing wear and tear to a minimum. The two modes are implemented via two disjunctive sets implicating that exactly one mode must be activated at any time.

In this case study, a focus is placed on the energy supply optimization problem in the lower layer. A detailed description of the energy demand optimization problem in the upper layer minimizing thermal and electrical energy consumption can be found in [56]. The nonlinear optimization problem of the energy supply is formulated as follows with $T = \{k, .., k + N - 1\}$ being the discretized prediction horizon (formulation before softening of variable bounds):

$$\min_{\substack{\dot{m}_{\text{stor,cons}}(t),\\ \dot{m}_{\text{cond}}(t),\\ \dot{Q}_{\text{cond}}(t),\delta_{\text{HP}}(t)}} \sum_{t=k}^{k+N-1} \left(\alpha \cdot P_{\text{el,HP}}(t) + \beta \cdot \left(\Delta\dot{Q}_{\text{stor}}(t) - \dot{Q}_{\text{ref}}(t)\right)^2\right) \cdot \Delta t \tag{8a}$$

$$\text{s.t. } F(t, dx, x, u, w, p) = 0 \qquad \forall t \in T \tag{8b}$$

$$\dot{m}_{\text{stor,cons,min}} \leq \dot{m}_{\text{stor,cons}}(t) \leq \dot{m}_{\text{stor,cons,max}} \qquad \forall t \in T \tag{8c}$$

$$\dot{m}_{\text{cond,min}} \leq \dot{m}_{\text{cond}}(t) \leq \dot{m}_{\text{cond,max}} \qquad \forall \dot{m}_{\text{cond}}(t) \in \mathbb{D}, \qquad \forall t \in T \tag{8d}$$

$$\dot{Q}_{\text{cond,min}} \leq \dot{Q}_{\text{cond}}(t) \qquad \forall t \in T \tag{8e}$$

$$\delta_{\text{HP}}(t) \cdot P_{\text{el,HP,min}} \leq P_{\text{el,HP}}(t) \leq \delta_{\text{HP}}(t) \cdot P_{\text{el,HP,max}} \qquad \forall t \in T \tag{8f}$$

$$\begin{aligned} &\delta_{\text{HP}}(t+k) - \delta_{\text{HP}}(t+k-1) \leq \delta_{\text{HP}}(\omega_{\text{up}}) \\ &\quad \forall \omega_{\text{up}} \in \{t+k, t+k+1, \ldots, min(t+\text{N}, t+k+\text{T}_{\text{HP}}^{\text{up}}-1)\} \qquad \forall t \in T \end{aligned} \tag{8g}$$

$$\begin{aligned} &\delta_{\text{HP}}(t+k-1) - \delta_{\text{HP}}(t+k) \leq 1 - \delta_{\text{HP}}(\omega_{\text{down}}) \\ &\quad \forall \omega_{\text{down}} \in \{t+k, t+k+1, \ldots, min(t+\text{N}, t+k+\text{T}_{\text{HP}}^{\text{down}}-1)\} \qquad \forall t \in T \end{aligned} \tag{8h}$$

$$\begin{bmatrix} Y_1(t) \\ \delta_{\text{HP}}(t) = 1 \\ \dot{m}_{\text{cond}}(t) \geq \dot{m}_{\text{cond,min}} \end{bmatrix} \veebar \begin{bmatrix} Y_2(t) \\ \delta_{\text{HP}}(t) = 0 \\ \dot{m}_{\text{cond}}(t) = \dot{m}_{\text{cond,min}} \end{bmatrix} \qquad \forall t \in T \tag{8i}$$

$$\delta_{\text{HP}}(t) \in \{0, 1\} \qquad \forall t \in T \tag{8j}$$

$$Y_i(t) \in \{True, False\} \quad \forall i \in \{1, 2\} \qquad \forall t \in T \tag{8k}$$

In these equations, $\dot{m}_{\text{stor,cons}}(t)$ corresponds to the water mass flow between storage and consumer, $\dot{m}_{\text{cond}}(t)$ to the water mass flow through the condenser, $\dot{Q}_{\text{cond}}(t)$ to the thermal power provided by the condenser and $\delta_{\text{HP}}(t)$ to the on/ off operation state of the heat pump. $P_{\text{el,HP}}(t)$ is the electrical power consumption of the heat pump, $\Delta\dot{Q}_{\text{stor}}(t)$ the heating power supplied by the storage to the energy demand and $\dot{Q}_{\text{ref}}(t)$ the reference heating power requested by the energy demand in the upper level MPC. α and β constitute weighting factors for the different terms of the cost function. The first term of the cost function 8a comprises the electrical power consumption of the heat pump and the second term penalizes the deviation of the provided from the requested heating power. As the original cost function is quadratic due to the second term and since both the Jacobians and Hessians are integrated in the linearized optimization problem, an MIQP is solved.

The system DAEs in the nonlinear form are described by Eq. 8b. Both $\dot{m}_{\text{stor,cons}}(t)$ and $\dot{m}_{\text{cond}}(t)$ are constrained by minimum and maximum bounds (Eq. 8c and 8d). $\dot{m}_{\text{cond}}(t)$ is further restricted to the discrete, integer set $\mathbb{D}$ of possible operation states at intervals of 2. $\dot{Q}_{\text{cond}}(t)$ is only bounded by a minimum value $\dot{Q}_{\text{cond,min}}$ equal to 0 (Eq. 8e) as the heat pump operation is bounded by the minimum part load constraint including the upper bound in Eq. 8f. Eq. 8g and 8h describe the minimum on/ off time constraints according to [19] with the minimum up time $\text{T}_{\text{HP}}^{\text{up}}$, the minimum down time $\text{T}_{\text{HP}}^{\text{down}}$ (both equal to 2 h) and prediction horizon N. Eq. 8i represents the operation modes in the form of the disjunctive sets specified by the GDP which are connected by an exclusive-OR forcing exactly one of the operation modes to be active. $Y_1(t)$ and $Y_2(t)$ are Boolean variables indicating the activation status of each mode. The energy supply optimization problem consists of 74 DAEs and 81 variables after the BLT reduction (20 states, 54 algebraic variables, 4 control inputs and 3 external inputs).

The performance of the proposed toolchain on the hybrid nonlinear MPC use case based on time-variant linearization around a trajectory (trajectory-LTV) is compared against control variants based on time-variant linearization around a reference point (point-LTV), LTI models and a reference control concept of an NLP formulation of the optimization problem. The NLP is performed based on JModelica.org and IPOPT with a post-processing after every MPC iteration. The post-processing is employed to approximate the different integer characteristics as they do not form part of the NLP. For all control variants, an additional constraint that prescribes the top layer storage temperature at the end of the simulation horizon to be equal to a prespecified value of 302 K is imposed to guarantee comparability between the approaches.

The different control approaches are compared in terms of the Key Performance Indicators (KPIs) tracking of the requested heating power (in the form of the Root Mean Square Error (RMSE)), electrical energy consumption, deviation of simulated and optimized trajectories for the provided heating power $\Delta\dot{Q}_{\mathrm{stor}}$ (also as an indicator for the linearization error; in the form of the RMSE) and computational time ratio. The computational time ratio quantifies the ratio of the computation time to the sampling time (period during which the building is controlled). If the quotient is smaller than 1, the control strategy is real-time capable.

The used execution platform as well as optimization tools and settings are listed in Table 1. For the algebraic reformulation of the GDP, the *bigM* transformation method is chosen.

Execution platform	OpenStack instance based on a Linux machine, Ubuntu 18.04, 8 vCPUs and 32 GB RAM
Optimization framework	*Energy demand*: JModelica.org 2.14, IPOPT 3.13.1, linear HSL solver ma27 [68] *Energy supply*: Pyomo 5.7.2, Gurobi 9.1.1, MIPGap: 0.01 %, TimeLimit: 60 s
Discretization	*For both energy demand and supply*: Collocation with 96 collocation elements with 2 collocation points and piecewise constant control inputs
MPC parameters	*For both energy demand and supply*: Prediction horizon: 24 h, sampling period: 15 min, simulation horizon: 3 days (=72h) starting at 8 a.m.

Table 1: Optimization parameters

The evaluated KPIs for the different control approaches are given in Table 2. In the following, the corresponding control performance is detailed for every control variant.

In Fig. 7 the simulation results for the MIQP with LTV models based on time-variant linearization around a trajectory (trajectory-LTV) are shown. For each MPC iteration, the linearization reference points are updated at intervals of 15 min along the simulation trajectory. Hereinafter, this control approach is referred to as the "proposed approach". The first subplot depicts the fulfillment of the requested heating power $\dot{Q}_{\mathrm{ref}}(t)$ by the provided power $\Delta\dot{Q}_{\mathrm{stor}}(t)$. The requested heating power is tracked very well by the provided one resulting in an RMSE of 18.6 W. In order to evaluate the linearization error introduced by the trajectory-LTV models, the optimized trajectories for the provided heating power are plotted as well in the first subplot. The simulated and optimized trajectories show a good agreement throughout the simulation horizon with an RMSE between the simulated and optimized trajectories of 5.2 W. This implies a small linearization error and accordingly, the time-variant linearization around a trajectory can precisely approximate the nonlinear system behavior. In the second subplot, the water mass flow between storage and consumer $\dot{m}_{\mathrm{stor,cons}}(t)$ is shown revealing a similar trend as the provided heating power. The third subplot of the mass flow through the condenser $\dot{m}_{\mathrm{cond}}(t)$ exhibits the successful implementation of the discrete operation states and the operation modes. The mass flow values are just chosen from the discrete set of integer numbers at intervals of 2. In combination with the last subplot of the heat pump operation states, it can be derived that the mass flow through the condenser is just operated above its minimum bound when the heat pump is operated otherwise it is restricted to its minimum value. In the fourth subplot, the trajectories of the storage temperatures (top and bottom layer) including the inlet temperatures from the consumer and heat pump circuits are shown. To be capable of providing the heating power demand during days two and three, which is higher compared to day one, the temperature of the storage is increased in an anticipatory manner by adding more thermal energy than is released to the consumer side. As shown in the "Heat pump power" subplot, the trajectories of the electrical heat pump power fulfill the minimum part load constraint. The heat pump is either switched off or operated above the minimum part load of 30 % of the nominal power while at the same time satisfying the maximum bound. Likewise, the minimum on and off times of 2 h for the heat pump operation are complied with as depicted in the "Heat pump state" subplot showing a smooth, non-oscillatory behavior of the heat pump operation state. The electrical energy consumption is 14 034 kJ and the computational time ratio is 0.15. Accordingly, the approach is 7 faster than real-time.

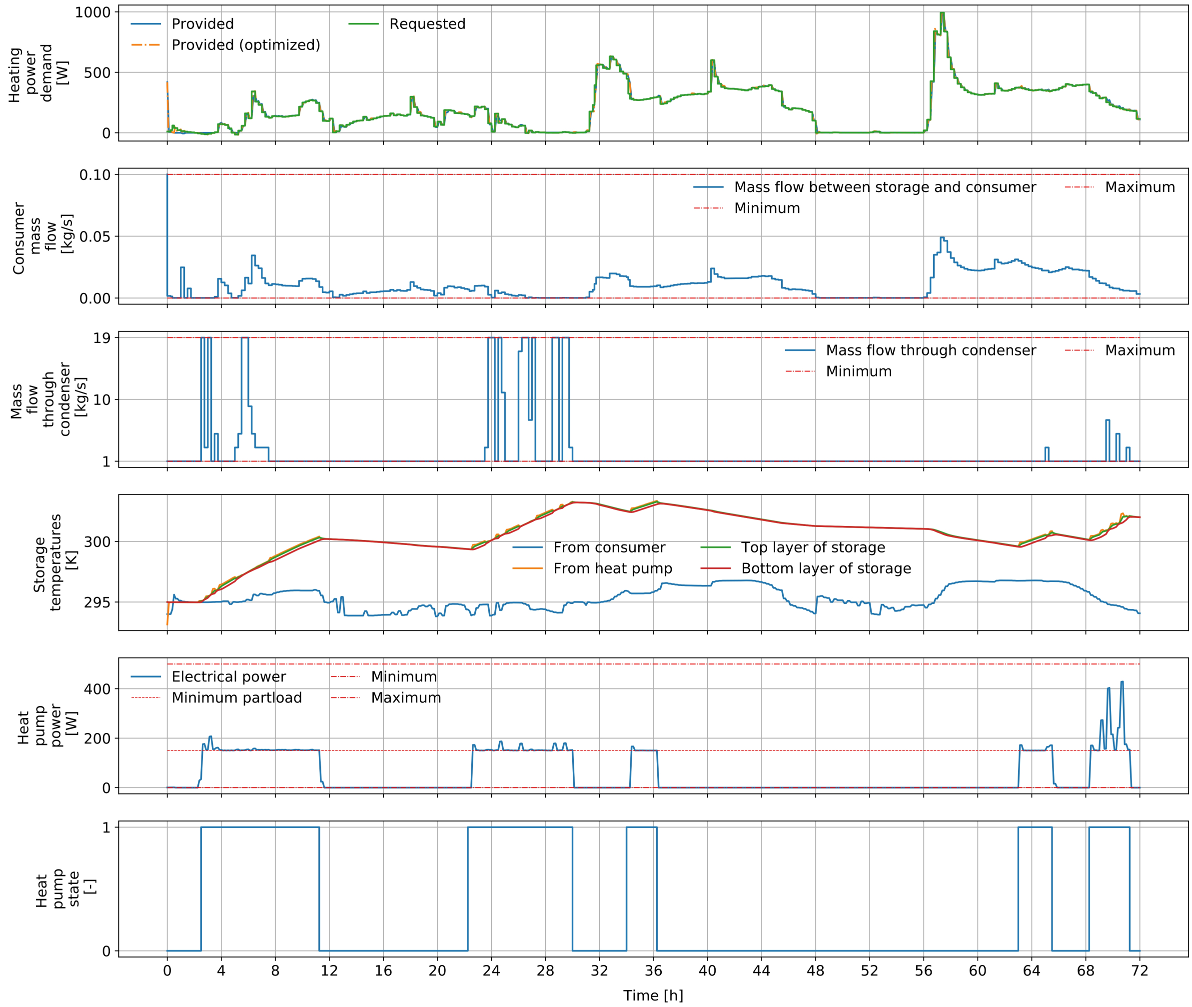


Figure 7: Simulation results of the MIQP - trajectory-LTV

In Fig. 8 the results for the MIQP including LTV models based on time-variant linearization around a reference point (point-LTV) are depicted. The same optimization settings as for the linearization around the trajectory (trajectory-LTV) are applied. The current operating point ($t = 0$) of every new MPC iteration is chosen as the linearization reference point. Compared to the approach based on the trajectory-LTV models, additional outliers at hours 4–8, 8–12 and especially 34–46 characterize the first subplot introducing deviations between the provided and requested heating power whereas for most of the prediction horizon the tracking of the requested heating power performs well. As the partly high deviation between simulated and optimized provided heating power reveals, the linearization reference points chosen for the point-LTV models are not able to reproduce the system dynamics over the full operating range. As an example, this results in a decrease of the storage temperatures below the level of the consumer return temperatures during hours 36-44 as shown in the "Storage temperatures" subplot. In addition, in the "Heat pump power" subplot it can be seen that there are violations of the upper variable bound around hours 45 and 49. The energy consumption is 11.7 % lower compared to the linearization around the trajectory with 12 395 kJ which can be attributed to the worse tracking of the heating power (especially around hours 34–46) with an RMSE of heating power tracking of 132.8 W (614 % higher compared to trajectory-LTV). The constraints for minimum part load, minimum on/ off times and the operation modes including discrete operation states are fulfilled for this control approach as well apart from a slight violation of the minimum part load at hours 70–72 and the minimum on/ off times at hour 17. The RMSE of the deviation of simulated and optimized trajectories for $\Delta\dot{Q}_{\mathrm{stor}}$ is 134.0 W and is 2477 % higher compared to the control approach based on the trajectory-LTV models. This can be attributed to a higher induced linearization error and reduced approximation accuracy using the point-LTV models. The computational time ratio is 0.14.

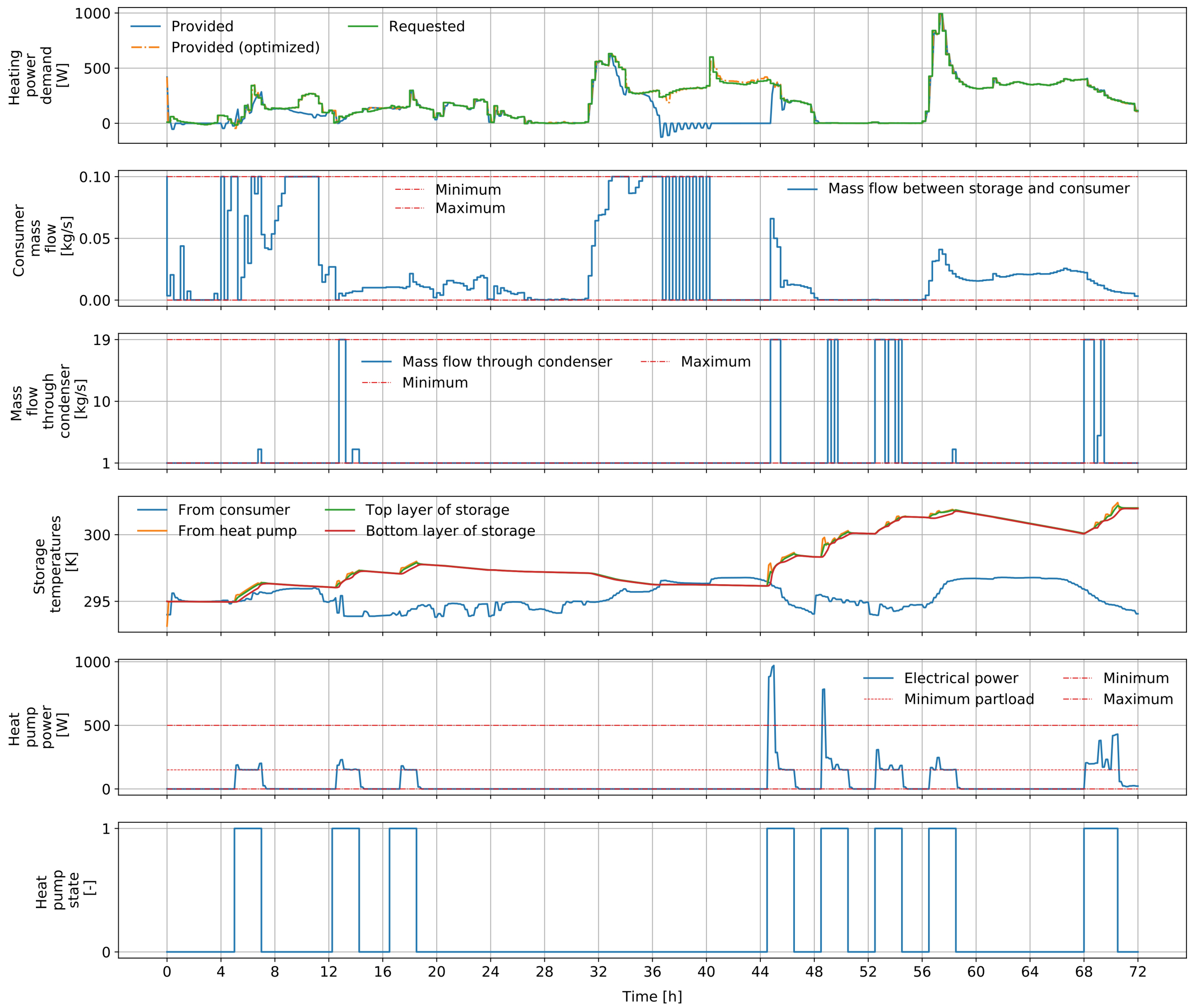


Figure 8: Simulation results of the MIQP - point-LTV

Fig. 9 presents the results for the MIQP based on time-invariant linearization around a reference point (LTI) using the same optimization settings as for the first two approaches. The linearization is performed just once after the first MPC iteration around the operating point $t = 0$ at this time. An MPC iteration is performed before the linearization to find a linearization reference point where the system dynamics are already initialized properly. Compared to the first two control approaches, the results show a considerably poorer control performance with a significantly increased linearization error. In the first subplot, the trajectory for the provided heating power shows a partly similar but significantly shifted trend compared to the requested power trajectory. This results in a high RMSE of the heating power tracking of 399 W (2046 % higher compared to the proposed approach) and an RMSE of the deviation between the simulated and optimized trajectories for $\Delta\dot{Q}_{\mathrm{stor}}$ of 398 W (7562 % higher). Additionally, the "Heat pump power" subplot reveals substantial violations of the upper variable bound. Thus, the operating point chosen as the reference point for performing the time-invariant linearization does not suffice for reproducing the system dynamics and operating conditions throughout the simulation horizon.

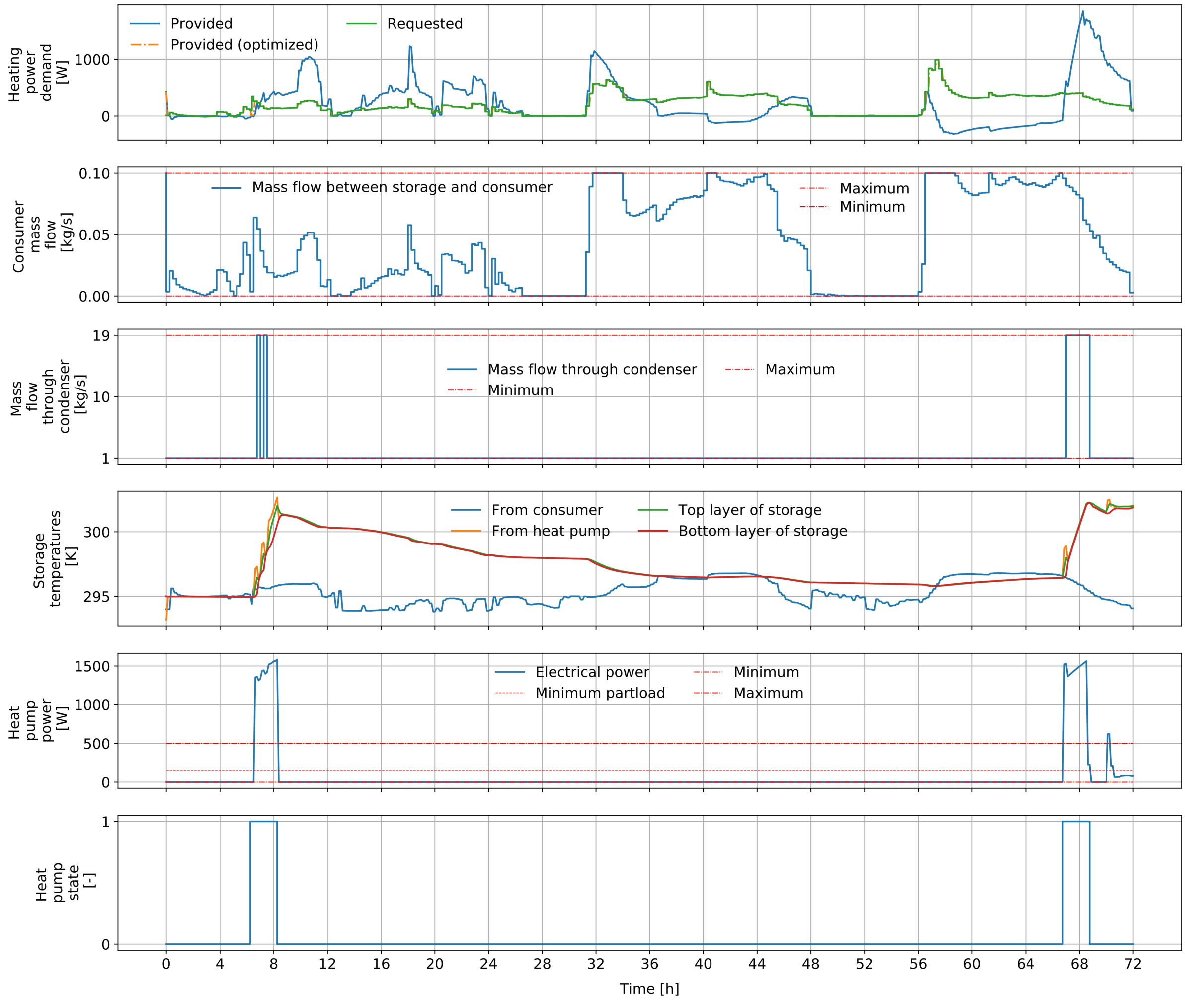


Figure 9: Simulation results of the MIQP - LTI

In Fig. 10 the control performance of the NLP including post-processing and approximation of the integer characteristics after every MPC iteration is presented. The discrete operation states from Eq. 8d as well as the minimum part load constraint from Eq. 8f are taken into account by rounding the optimization results based on the most recent evaluated value of the heat pump COP. The minimum on/ off times are forced in every MPC iteration by comparing the spent time in a heat pump state with the specified minimum time. The different operation modes are imposed on $\dot{m}_{\mathrm{cond}}(t)$ depending on the current operation state of the heat pump. Due to the integration of the post-processing the NLP leads to worse KPIs for the tracking of the requested heating power and the deviation between simulated and optimized trajectories for $\Delta\dot{Q}_{\mathrm{stor}}$ compared with the proposed approach as the actually implemented variables differ from the optimized ones. The first subplot reveals this behavior with slight deviations between the "provided" (simulated after post-processing) and "provided (optimized)" heating power. The RMSE of the heating power tracking is 37.4 W and 101 % higher in comparison to the proposed approach. The consumed electrical energy is 14 040 kJ which is almost identical to the proposed approach and the RMSE of the deviation between the simulated and optimized trajectories for $\Delta\dot{Q}_{\mathrm{stor}}$ increases by 310 % (with 21.3 W) compared to the proposed approach. The computational time ratio of the approach is 0.04. It should be noted that for this case study the NLP including post-processing achieves sufficient control performance; however, for larger and more complex systems with more binary or integer constraints, if-then-else relationships and particularly multiple operation modes including switching dynamics, the formulation of the post-processing will become cumbersome up to impossible. Additionally, the post-processing may lead to a high deviation between the optimized and actually implemented variables and potential violations of constraints.

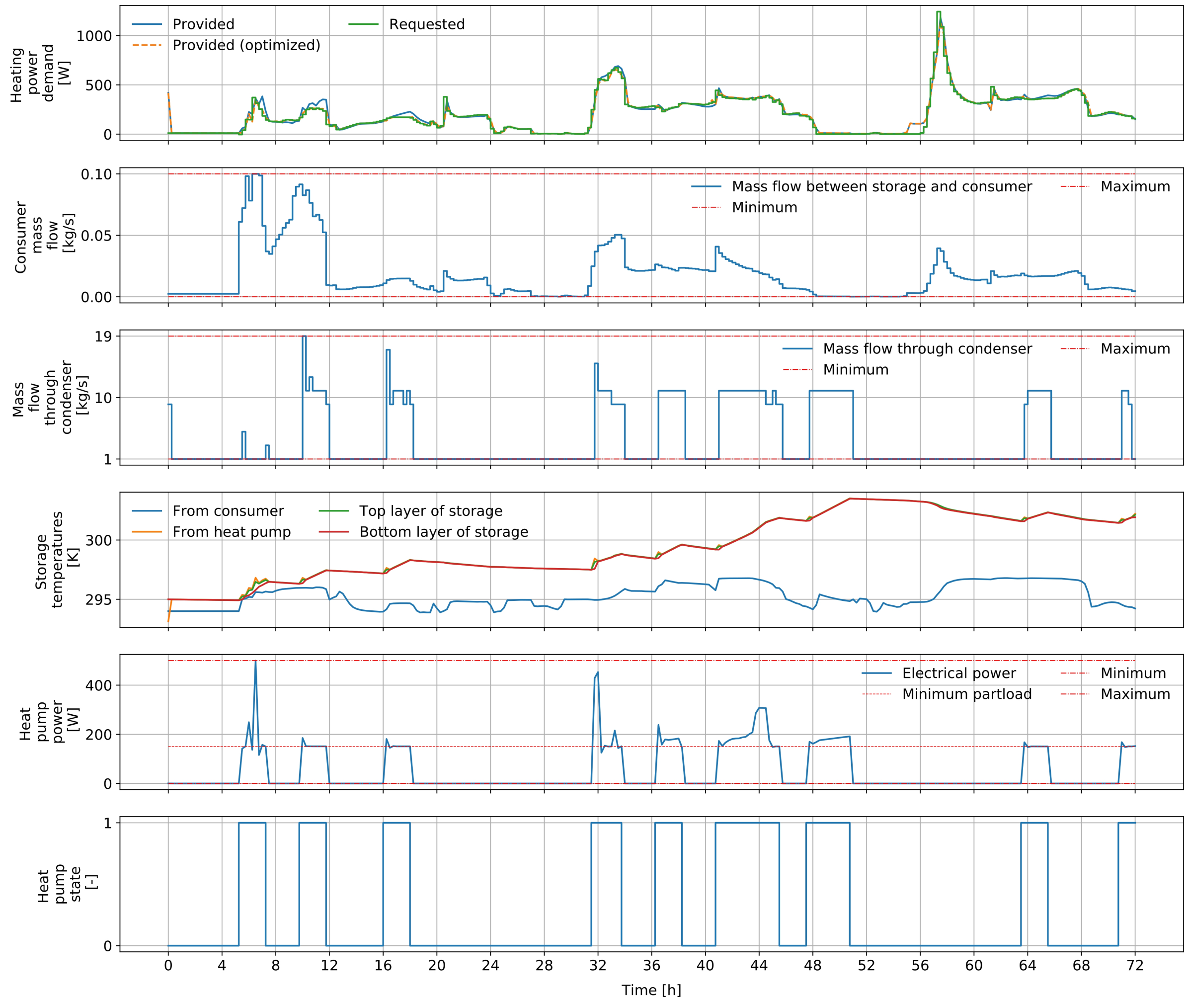


Figure 10: Simulation results of the NLP including post-processing

To conclude, a Modelica-based toolchain for nonlinear hybrid MPC based on automated time-variant linearization around a reference trajectory has been developed and its control performance is compared in a nonlinear case study against different control variants with a lower relinearization frequency and an

	RMSE heating power tracking in W	Consumed electrical energy in kJ	RMSE deviation of sim./ opt. trajectory for $\Delta \dot{Q}_{\text{stor}}$ in W	Comp. time in s / comp. time ratio
MIQP trajectory-LTV	18.6	14 034	5.2	39 849 / 0.15
MIQP point-LTV	132.8	12 395	134.0	36 578 / 0.14
MIQP LTI	399.1	19 612	398.4	30 939 / 0.12
NLP incl. post-processing	37.4	14 040	21.3	10 188 / 0.04

Table 2: Comparison of the KPIs for the different control approaches

NLP with an approximation of the integer characteristics. The control performance results demonstrate a small linearization error induced by the time-variant linearization module and a successful real-time capable implementation of all integer characteristics and constraints. With respect to the applied KPIs the proposed approach outperforms the reference control variants with a lower relinearization frequency and the NLP approximating the integer characteristics.

5. Conclusion and outlook

In this paper, a toolchain for nonlinear hybrid MPC of building energy systems based on Modelica models is presented which bridges the gap between Modelica and mixed-integer optimization. The approach combines the integration of both nonlinearities (characteristic of building energy systems and integrated HVAC) and integer decision variables which often arise due to discrete operation states, on/ off control, if-then-else relationships or operation modes. The toolchain builds upon a high-accuracy time-variant linearization approach, which transforms the nonlinear Modelica optimization problem into a linearized state-space representation in every MPC iteration. Based on the linearization output, an optimization problem is automatically formulated in the modeling framework Pyomo which can be extended by various integer characteristics and is solved by the MIQP solver Gurobi.

The functionality of the approach is demonstrated in a simulative control of a building energy system based on a nonlinear Modelica case study. The global optimization problem is split into the subproblems of energy demand and energy supply which are solved within a hierarchical optimization. The energy demand model comprises a thermal zone that is heated by a convector and CCA and is solved as an NLP based on JModelica.org and IPOPT. The energy supply model provides the heating energy to the thermal zone and consists of a coupled model of a thermal buffer storage and a heat pump. The energy supply optimization problem includes on/ off control including minimum part load, minimum on/ off times, discrete operation states and different operation modes and is solved as an MIQP based on the proposed toolchain. The simulation results demonstrate a good performance of the MIQP approach based on the time-variant linearization including a small linearization error, a successful real-time capable implementation of all integer characteristics and outperforms the reference control concepts with a lower relinearization frequency and the NLP approximating the integer characteristics.

Future work and improvements consist in integrating the nonlinear hybrid MPC in a distributed MPC framework, application to use cases with more complex operation modes including switching dynamics, model calibration and the practical implementation of the approach.

Author contributions

Maximilian Mork: Conceptualization, Methodology, Software, Investigation, Writing - Original draft, Writing - Review & Editing, Visualization. Nick Materzok: Methodology, Software, Investigation, Writing - Review & Editing, Visualization. André Xhonneux: Supervision, Writing - review & editing, Project administration, Funding acquisition, Resources. Dirk Müller: Supervision, Writing - review & editing, Project administration, Funding acquisition, Resources.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

Funding: This work was supported by the BMWi (Federal Ministry for Economic Affairs and Energy), promotional reference 03EGB0010A.

References

[1] European Union, Directive (EU) 2018/844 of the European Parliament and of the Council of 30 May 2018 amending Directive 2010/31/EU on the energy performance of buildings and Directive 2012/27/EU on energy efficiency, 2018. URL: https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32018L0844&from=EN.

[2] European Commission, Commission Recommendation (EU) 2019/1019 of 7 June 2019 on building modernisation, 2019. URL: https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32019H1019&from=EN.

[3] European Commission, The European green deal: communication from the commission to the European Parliament. the European Council, the Council, the European Economic and Social Committee and the Committee of the Regions, 2019. URL: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:52019DC0640.

[4] M. W. Ahmad, M. Mourshed, B. Yuce, Y. Rezgui, Computational intelligence techniques for HVAC systems: A review, Build. Simul. 9 (2016) 359–398. doi:10.1007/s12273-016-0285-4.

[5] International Energy Agency, Building Energy Performance Metrics- Supporting Energy Efficiency Progress in Major Economies, 2015. URL: https://www.iea.org/reports/building-energy-performance-metrics.

[6] A. Afram, F. Janabi-Sharifi, Theory and applications of HVAC control systems - A review of model predictive control (MPC), Build. Environ. 72 (2014) 343–355. doi:10.1016/j.buildenv.2013.11.016.

[7] G. Serale, M. Fiorentini, A. Capozzoli, D. Bernardini, A. Bemporad, Model Predictive Control (MPC) for enhancing building and HVAC system energy efficiency: Problem formulation, applications and opportunities, Energies 11 (2018). doi:10.3390/en11030631.

[8] M. Killian, M. Kozek, Ten questions concerning model predictive control for energy efficient buildings, Build. Environ. 105 (2016) 403–412. doi:10.1016/j.buildenv.2016.05.034.

[9] J. Cigler, D. Gyalistras, J. Siroky, V.-N. Tiet, L. Ferkl, Beyond Theory: the Challenge of Implementing Model Predictive Control in Buildings, in: 11th REHVA world Congr. 8th Int. Conf. Energy Effic. Smart Heal. Build., 2013, pp. 1008–1018.

[10] S. R. West, J. K. Ward, J. Wall, Trial results from a model predictive control and optimisation system for commercial building HVAC, Energy Build. 72 (2014) 271–279. doi:10.1016/j.enbuild.2013.12.037.

[11] S. C. Bengea, A. D. Kelman, F. Borrelli, R. Taylor, S. Narayanan, Implementation of model predictive control for an HVAC system in a mid-size commercial building, HVAC R Res. 20 (2014) 121–135. doi:10.1080/10789669.2013.834781.

[12] R. De Coninck, L. Helsen, Practical implementation and evaluation of model predictive control for an office building in Brussels, Energy Build. 111 (2016) 290–298. doi:10.1016/j.enbuild.2015.11.014.

[13] D. Sturzenegger, D. Gyalistras, M. Morari, R. S. Smith, Model Predictive Climate Control of a Swiss Office Building: Implementation, Results, and Cost-Benefit Analysis, IEEE Trans. Control Syst. Technol. 24 (2016) 1–12. doi:10.1109/TCST.2015.2415411.

[14] J. Drgoňa, J. Arroyo, I. Cupeiro Figueroa, D. Blum, K. Arendt, D. Kim, E. P. Ollé, J. Oravec, M. Wetter, D. L. Vrabie, L. Helsen, All you need to know about model predictive control for buildings, Annu. Rev. Control 50 (2020) 190–232. doi:10.1016/j.arcontrol.2020.09.001.

[15] J. Drgoňa, L. Helsen, Different Problem Classes and Solution Techniques for Model Predictive Building Control, in: Proc. REHVA Annu. Meet. Conf. Low Carbon Technol. HVAC, Brussels, Belgium, 2018.

[16] S. Koehler, F. Borrelli, Building temperature distributed control via explicit MPC and 'Trim and Respond' methods, in: 2013 Eur. Control Conf. ECC 2013, IEEE, Zurich, Switzerland, 2013, pp. 4334–4339. doi:10.23919/ecc.2013.6669781.

[17] F. Lauro, L. Longobardi, S. Panzieri, An adaptive distributed predictive control strategy for temperature regulation in a multizone office building, in: Proc. - 2014 IEEE Int. Work. Intell. Energy Syst. IWIES 2014, IEEE, San Diego, CA, USA, 2014, pp. 32–37. doi:10.1109/IWIES.2014.6957043.

[18] S. S. Walker, W. Lombardi, S. Lesecq, S. Roshany-Yamchi, Application of Distributed Model Predictive Approaches to Temperature and CO2 Concentration Control in Buildings, IFAC-PapersOnLine 50 (2017) 2589–2594. doi:10.1016/j.ifacol.2017.08.107.

[19] B. Mayer, M. Killian, M. Kozek, Management of hybrid energy supply systems in buildings using mixed-integer model predictive control, Energy Convers. Manag. 98 (2015) 470–483. doi:10.1016/j.enconman.2015.02.076.

[20] J. Cigler, S. Prívara, Z. Váňa, E. Žáčeková, L. Ferkl, Optimization of Predicted Mean Vote index within Model Predictive Control framework: Computationally tractable solution, Energy Build. 52 (2012) 39–49. doi:10.1016/j.enbuild.2012.05.022.

[21] M. Knudsen, S. Petersen, Economic model predictive control of space heating and dynamic solar shading, Energy Build. 209 (2020). doi:10.1016/j.enbuild.2019.109661.

[22] J. Široký, J. Cigler, L. Ferkl, Hybrid MPC approach to reconfiguration of building heating system, in: 2013 Eur. Control Conf. ECC 2013, IEEE, Zurich, Switzerland, 2013, pp. 2675–2680. doi:10.23919/ecc.2013.6669256.

[23] D. Picard, F. Jorissen, L. Helsen, Methodology for Obtaining Linear State Space Building Energy Simulation Models, in: Proc. 11th Int. Model. Conf. Sept. 21-23, 2015, Versailles, France, 2015, pp. 51–58. doi:10.3384/ecp1511851.

[24] R. Sangi, A. Kümpel, D. Müller, Real-life implementation of a linear model predictive control in a building energy system, J. Build. Eng. 22 (2019) 451–463. doi:10.1016/j.jobe.2019.01.002.

[25] H. Runvik, P.-O. Larsson, S. Velut, J. Funkquist, M. Bohlin, A. Nilsson, S. Modarrez Razavi, Production Planning for Distributed District Heating Networks with JModelica.org, in: Proc. 11th Int. Model. Conf. Sept. 21-23, 2015, volume 118, Versailles, France, 2015, pp. 217–223. doi:10.3384/ecp15118217.

[26] M. Pčolka, E. Žáčeková, R. Robinett, S. Čelikovský, M. Sebek, From Linear to Nonlinear Model Predictive Control of a Building, IFAC Proc. Vol. 47 (2014) 587–592. doi:10.3182/20140824-6-ZA-1003.02783.

[27] M. Pčolka, E. Žáčeková, R. Robinett, S. Čelikovský, M. Šebek, Bridging the gap between the linear and nonlinear predictive control: Adaptations for efficient building climate control, Control Eng. Pract. 53 (2016) 124–138. doi:10.1016/j.conengprac.2016.01.007.

[28] S. Rastegarpour, S. Gros, L. Ferrarini, MPC approaches for modulating air-to-water heat pumps in radiant-floor buildings, Control Eng. Pract. 95 (2020) 104209. doi:10.1016/j.conengprac.2019.104209.

[29] S. Rastegarpour, R. Scattolini, L. Ferrarini, Performance improvement of an air-to-water heat pump through linear time-varying MPC with adaptive COP predictor, J. Process Control 99 (2021) 69–78. doi:10.1016/j.jprocont.2021.01.006.

[30] F. Oldewurtel, A. Parisio, C. N. Jones, D. Gyalistras, M. Gwerder, V. Stauch, B. Lehmann, M. Morari, Use of model predictive control and weather forecasts for energy efficient building climate control, Energy Build. 45 (2012) 15–27. doi:10.1016/j.enbuild.2011.09.022.

[31] D. Sturzenegger, D. Gyalistras, V. Semeraro, M. Morari, R. S. Smith, BRCM Matlab Toolbox: Model generation for model predictive building control, in: 2014 Am. Control Conf., IEEE, 2014, pp. 1063–1069. doi:10.1109/ACC.2014.6858967.

[32] M. Dan, S. Srinivasan, S. Sundaram, A. Easwaran, L. Glielmo, A Scenario-Based Branch-and-Bound Approach for MES Scheduling in Urban Buildings, IEEE Trans. Ind. Informatics 16 (2020) 7510–7520. doi:10.1109/TII.2020.2978870. arXiv:2003.03750.

[33] Y. Lu, S. Wang, Y. Sun, C. Yan, Optimal scheduling of buildings with energy generation and thermal energy storage under dynamic electricity pricing using mixed-integer nonlinear programming, Appl. Energy 147 (2015) 49–58. doi:10.1016/j.apenergy.2015.02.060.

[34] F. Berkenkamp, M. Gwerder, Hybrid model predictive control of stratified thermal storages in buildings, Energy Build. 84 (2014) 233–240. doi:10.1016/j.enbuild.2014.07.052.

[35] J. D. Feng, F. Chuang, F. Borrelli, F. Bauman, Model predictive control of radiant slab systems with evaporative cooling sources, Energy Build. 87 (2015) 199–210. doi:10.1016/j.enbuild.2014.11.037.

[36] A. Khakimova, A. Kusatayeva, A. Shamshimova, D. Sharipova, A. Bemporad, Y. Familiant, A. Shintemirov, V. Ten, M. Rubagotti, Optimal energy management of a small-size building via hybrid model predictive control, Energy Build. 140 (2017) 1–8. doi:10.1016/j.enbuild.2017.01.045.

[37] K. Deng, Y. Sun, S. Li, Y. Lu, J. Brouwer, P. G. Mehta, M. Zhou, A. Chakraborty, Model Predictive Control of Central Chiller Plant With Thermal Energy Storage Via Dynamic Programming and Mixed-Integer Linear Programming, IEEE Trans. Autom. Sci. Eng. 12 (2015) 565–579. doi:10.1109/TASE.2014.2352280.

[38] M. Fiorentini, J. Wall, Z. Ma, J. H. Braslavsky, P. Cooper, Hybrid model predictive control of a residential HVAC system with on-site thermal energy generation and storage, Appl. Energy 187 (2017) 465–479. doi:10.1016/j.apenergy.2016.11.041.

[39] Y. Huang, W. Wang, B. Hou, A hybrid algorithm for mixed integer nonlinear programming in residential energy management, J. Clean. Prod. 226 (2019) 940–948. doi:10.1016/j.jclepro.2019.04.062.

[40] G. Schweiger, P.-O. Larsson, F. Magnusson, P. Lauenburg, S. Velut, District heating and cooling systems – Framework for Modelica-based simulation and dynamic optimization, Energy 137 (2017) 566–578. doi:10.1016/j.energy.2017.05.115.

[41] J. Zhong, Y. Tan, Y. Li, Y. Cao, Y. Peng, Z. Zeng, Y. Nakanishi, Y. Zhou, Distributed Operation for Integrated Electricity and Heat System With Hybrid Stochastic/Robust Optimization, Int. J. Electr. Power Energy Syst. 128 (2021) 106680. doi:10.1016/j.ijepes.2020.106680.

[42] Y. Ma, F. Borrelli, B. Hencey, A. Packard, S. Bortoff, Model Predictive Control of thermal energy storage in building cooling systems, in: Proc. 48h IEEE Conf. Decis. Control held jointly with 2009 28th Chinese Control Conf., IEEE, Shanghai, China, 2009. doi:10.1109/CDC.2009.5400677.

[43] C. R. Touretzky, M. Baldea, A hierarchical scheduling and control strategy for thermal energy storage systems, Energy Build. 110 (2016) 94–107. doi:10.1016/j.enbuild.2015.09.049.

[44] A. Bürger, C. Zeile, A. Altmann-Dieses, S. Sager, M. Diehl, Design, implementation and simulation of an MPC algorithm

for switched nonlinear systems under combinatorial constraints, J. Process Control 81 (2019) 15–30. doi:10.1016/j.jprocont.2019.05.016.
[45] G. Schweiger, R. Heimrath, B. Falay, K. O'Donovan, P. Nageler, R. Pertschy, G. Engel, W. Streicher, I. Leusbrock, District energy systems: Modelling paradigms and general-purpose tools, Energy 164 (2018) 1326–1340. doi:10.1016/j.energy.2018.08.193.
[46] M. Wetter, C. Van Treeck, New Generation Computational Tools for Building and Community Energy Systems Annex 60 Final Report, 2017. URL: http://www.iea-annex60.org/downloads/iea-ebc-annex60-final-report.pdf.
[47] J. Åkesson, K. E. Årzén, M. Gäfvert, T. Bergdahl, H. Tummescheit, Modeling and optimization with Optimica and JModelica.org-Languages and tools for solving large-scale dynamic optimization problems, Comput. Chem. Eng. 34 (2010) 1737–1749. doi:10.1016/j.compchemeng.2009.11.011.
[48] A. Wächter, L. T. Biegler, On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming, Math. Program. 106 (2006) 25–57. doi:10.1007/s10107-004-0559-y.
[49] C. Gehbauer, D. H. Blum, T. Wang, E. S. Lee, An assessment of the load modifying potential of model predictive controlled dynamic facades within the California context, Energy Build. 210 (2020). doi:10.1016/j.enbuild.2020.109762.
[50] F. Jorissen, Toolchain for Optimal Control and Design of Energy Systems in Buildings, Ph.D. thesis, Arenberg Doctoral School, KU Leuven., 2018.
[51] G. Schweiger, F. Jorissen, H. Runvik, L. Helsen, Equation-based modelling for dynamic optimization of district scale energy systems – a scalability study, in: 2019 Int. Conf. Smart Energy Syst. Technol., IEEE, 2019. doi:10.1109/SEST.2019.8849039.
[52] W. E. Hart, C. D. Laird, J.-P. Watson, D. L. Woodruff, G. A. Hackebeil, B. L. Nicholson, J. D. Siirola, Pyomo — Optimization Modeling in Python. Second Edition., volume 67 of *Springer Optimization and Its Applications*, Springer International Publishing, Cham, 2017. doi:10.1007/978-3-319-58821-6.
[53] L. Gurobi Optimization, Gurobi Optimizer Reference Manual, 2021.
[54] Y. Ma, F. Borrelli, B. Hencey, B. Coffey, S. Bengea, P. Haves, Model Predictive Control for the Operation of Building Cooling Systems, IEEE Trans. Control Syst. Technol. 20 (2012) 796–803. doi:10.1109/TCST.2011.2124461.
[55] D. Müller, M. Lauster, A. Constantin, M. Fuchs, P. Remmen, Aixlib - an Open-Source Modelica Library Within the IEA-EBC Annex 60 Framework, in: Proc. CESBP Cent. Eur. Symp. Build. Phys. BauSIM 2016, Dresden, Germany, 2016, pp. 3–9.
[56] M. Mork, A. Xhonneux, D. Müller, Hierarchical Model Predictive Control for complex building energy systems, Bauphysik 42 (2020) 306–314. doi:10.1002/bapi.202000031.
[57] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, M. Diehl, CasADi: a software framework for nonlinear optimization and optimal control, Math. Program. Comput. 11 (2019) 1–36. doi:10.1007/s12532-018-0139-4.
[58] C. Andersson, J. Akesson, C. Fuhrer, PyFMI: A Python Package for Simulation of Coupled Dynamic Models with the Functional Mock-up Interface, Tech. Rep. Math. Sci. 2016 (2016) 1–40.
[59] J. Åkesson, Optimica: an extension of modelica supporting dynamic optimization, in: 6th Int. Model. Conf. 2008, Bielefeld, Germany, 2008.
[60] B. Nicholson, J. D. Siirola, J.-P. Watson, V. M. Zavala, L. T. Biegler, pyomo.dae: a modeling and automatic discretization framework for optimization with differential and algebraic equations, Math. Program. Comput. 10 (2018) 187–223. doi:10.1007/s12532-017-0127-0.
[61] F. Magnusson, J. Åkesson, Symbolic elimination in dynamic optimization based on block-triangular ordering, Optim. Methods Softw. 33 (2018) 92–119. doi:10.1080/10556788.2016.1270944.
[62] F. J. Baader, M. Mork, A. Xhonneux, D. Müller, A. Bardow, M. Dahmen, Mixed-Integer Dynamic Scheduling Optimization for Demand Side Management, Comput. Aided Chem. Eng. 48 (2020) 1405–1410. doi:10.1016/B978-0-12-823377-1.50235-4.
[63] S. Kuboth, F. Heberle, A. König-Haagen, D. Brüggemann, Economic model predictive control of combined thermal and electric residential building energy systems, Appl. Energy 240 (2019) 372–385. doi:10.1016/j.apenergy.2019.01.097.
[64] Q. Chen, E. S. Johnson, D. E. Bernal, R. Valentin, S. Kale, J. Bates, J. D. Siirola, I. E. Grossmann, Pyomo.GDP: an ecosystem for logic based modeling and optimization development, Optim. Eng. (2021). doi:10.1007/s11081-021-09601-7.
[65] A. Bemporad, M. Morari, Control of systems integrating logic, dynamics, and constraints, Automatica 35 (1999) 407–427. doi:10.1016/S0005-1098(98)00178-2.
[66] A. Bhattacharya, X. Ma, D. L. Vrabie, Model Predictive Control of Discrete-Continuous Energy Systems via Generalized Disjunctive Programming, IFAC-PapersOnLine 54 (2021) 913–918. doi:10.1016/j.ifacol.2021.11.288. arXiv:2001.06181.
[67] Modelon AB, JModelica.org User Guide - Version 2.14, 2019.
[68] HSL. A collection of Fortran codes for large scale scientific computation., 2013. URL: http://www.hsl.rl.ac.uk/.